University of Bremen

Deep Learning for Computed Tomography Reconstruction

Learned Methods, Deep Image Prior and Uncertainty Estimation

Johannes Leuschner

Ph.D. thesis
Colloquium held on November 30, 2023

Primary supervisor and first reviewer
Prof. Dr. Dr. h.c. Peter Maaß

Secondary supervisor
Prof. Dr. Bangti Jin

Second reviewer
Dr. Tatiana Bubba

Abstract

X-ray computed tomography (CT) is a highly relevant imaging technique with clinical and industrial applications. At its core, CT involves an image reconstruction task from detector measurements that are acquired from multiple projection angles. Improving CT reconstruction using deep learning, which is being explored and utilized in various fields, is a subject of recent and current research.

This thesis comprises six papers, whose contributions can be summarized as two-fold. First, several deep learning approaches are compared quantitatively and qualitatively, involving the creation of a benchmark dataset as well as the realization and evaluation of challenges for learned low-dose and sparse-view CT reconstruction. Second, several extensions of the deep image prior (DIP)—an unsupervised deep learning image reconstruction framework—are investigated. This includes its application to CT using total-variation regularization, pretraining on synthetically generated data, and uncertainty estimation via a probabilistic model. These extensions benefit DIP-based CT reconstruction in several ways, such as an improved reconstruction quality, an accelerated reconstruction process, and the identification of potential errors in the reconstruction. Additionally, a Bayesian experimental design approach utilizing the uncertainty estimation is studied for the selection of scanning angles based on a pilot scan.

Complementing the papers, which are included without any modifications in the second part of this thesis, the first part introduces relevant foundations, as well as a large overview of literature on deep learning for CT reconstruction.

Zusammenfassung

X-ray Computertomographie (CT) ist ein wichtiges bildgebendes Verfahren, das sowohl klinische als auch industrielle Anwendung findet. Zentral bei der CT ist die Bildrekonstruktion anhand von Detektormessungen, die für verschiedene Projektionswinkel aufgenommen werden. Die Entwicklung verbesserter CT-Rekonstruktionsalgorithmen mittels Deep Learning (deutsch: tiefem Lernen), welches für verschiedenste Anwendungsbereiche erforscht und genutzt wird, ist Gegenstand neuester Untersuchungen.

Diese Arbeit enthält sechs Artikel, deren Beitrag wie folgt zusammengefasst werden kann. Einerseits werden einige Deep-Learning-Ansätze quantitativ und qualitativ verglichen, wofür unter anderem ein Benchmark-Datensatz erstellt wurde und kleine Wettbewerbe zur gelernten Rekonstruktion von CT-Daten mit geringer Dosis und mit wenigen Projektionswinkeln durchgeführt und ausgewertet wurden. Andererseits werden mehrere Erweiterungen für den Deep Image Prior (DIP)—ein unüberwachter Deep-Learning-Ansatz zur Bildrekonstruktion—untersucht. Dies umfasst seine Anwendung auf CT unter Verwendung von Totalvariations-Regularisierung, das Vortrainieren auf synthetisch generierten Daten sowie die Unsicherheitsschätzung mittels eines probabilistischen Modells. Diese Erweiterungen nützen der DIP-basierten CT-Rekonstruktion auf mehrere Weisen, etwa durch eine verbesserte Rekonstruktionsqualität, einen beschleunigten Rekonstruktionsprozess und die Identifikation potenzieller Fehler in der Rekonstruktion. Zusätzlich wird ein Bayesianischer Versuchsplanungsansatz untersucht, der die Unsicherheitsschätzung nutzt, um die aufzunehmenden Projektionswinkel basierend auf einem Pilot-Scan auszuwählen.

Als Ergänzung zu den Artikeln, die unverändert im zweiten Teil dieser Arbeit eingebunden sind, werden im ersten Teil zum einen relevante Grundlagen eingeführt, und zum anderen wird ein großer Überblick über die Literatur zu Deep Learning für CT-Rekonstruktion gegeben.

Contents

Chapter i

Preliminaries

i.1 Outline

This cumulative thesis is based on six selected papers and is organized in two parts. Part II directly includes the papers, and part I provides foundations and literature context.

Part I is split in two chapters, where chapter 1 introduces the inverse problem of computed tomography as well as deep learning, and chapter 2 covers the application of deep learning to computed tomography reconstruction.

Before the two main parts, we list the papers included in this thesis in section i.2. For each paper, we describe it shortly, clarify author contributions, and point out its context including forward-references to the relevant sections in part I chapter 2. In section i.3, some used notation and abbreviations are tabularized.

i.2 Papers and author contributions

Six papers [252, 23, 253, 30, 14, 31] are included in this thesis, which incorporate the main contributions of the author. From the other publications, the works [97, 96, 307, 33, 272] involved contributions of the author to a lesser extent and therefore have not been selected to be part of the cumulative thesis, but are relevant to the topic and will also be discussed in part I chapter 2. The contributions to other co-authored papers [342, 340, 20] during the time as a Ph.D. student are minor and are less relevant to this thesis. Below, we list the six included papers while clarifying author contributions. Throughout the thesis, we use purple instead of regular blue numbers when citing any of the included papers in order to highlight their integration in the context in a non-intrusive way.

LoDoPaB-CT, a benchmark dataset for low-dose CT reconstruction [252]

Citation:
J. Leuschner, M. Schmidt, D. O. Baguer, and P. Maass. "LoDoPaB-CT, a benchmark dataset for low-dose computed tomography reconstruction". In: *Scientific Data* 8.1 (2021), p. 109. ISSN: 2052-4463. DOI: 10.1038/s41597-021-00893-z

Description:
We construct a public dataset of paired CT images and simulated low-dose observations, designed for training and benchmarking learned reconstruction methods. The ground truth data is extracted from the LIDC-IDRI database [18].

Contributions:
Maximilian Schmidt and Johannes Leuschner contributed equally. Peter Maass pointed out the need for a public CT reconstruction benchmark dataset and organized an initial meeting, where we received valuable input from experienced community members. Maximilian Schmidt and Johannes Leuschner designed, curated, implemented and documented the dataset. Maximilian Schmidt contributed most to the conceptualization and writing, and also by testing. The technical dataset creation and the supporting DIV$\alpha\ell$ library, which includes simple access to the dataset, were mainly developed by Johannes Leuschner. Daniel Otero Baguer implemented the U-Net post-processing reconstruction method serving as a first reference and also contributed to the curation.

Context and connections with chapter 2:
A large-scale paired dataset is a prerequisite for many of the methods described in chapter 2. The dataset served as a prerequisite for our benchmarks in [23] and [253], and has been frequently used (as weak but easy-to-access indicators, on 6th of July 2023, Crossref counted 26 citations of the published article, Google Scholar counted 76 citations of the preprint or published article, and Zenodo counted several thousand data downloads). The included U-Net baseline is a directly trained post-processing, which is described in section 2.1.1.

Links:
https://www.nature.com/articles/s41597-021-00893-z — Open access paper
https://zenodo.org/record/3384092 — Dataset
https://github.com/jleuschn/dival — DIV$\alpha\ell$: Utilities and reference method collection
https://github.com/jleuschn/lodopab_tech_ref — Technical reference data and scripts
https://lodopab.grand-challenge.org — Challenge website with public leaderboard

CT reconstruction using DIP and learned reconstruction methods [23]

Citation:
D. O. Baguer, **J. Leuschner**, and M. Schmidt. "Computed tomography reconstruction using deep image prior and learned reconstruction methods". In: *Inverse Problems* 36.9 (2020), p. 094004. DOI: 10.1088/1361-6420/aba415

Description:
We investigate deep image prior [249] with total variation regularization [265] (DIP+TV) for CT reconstruction and benchmark it against different learned reconstruction approaches given a varying amount of training data. For the low-data regime, in which DIP+TV competes with learned methods, we propose to combine both by leveraging an initial learned reconstruction.

Contributions:
Daniel Otero Baguer came up with the ideas behind this paper. Implementation of all methods and experiments were split between all three authors. The implementation uses the DIV$\alpha\ell$ library (https://github.com/jleuschn/dival) principally developed by Johannes Leuschner, and, in turn, implementations and trained networks from this paper were integrated into DIV$\alpha\ell$ afterwards by Johannes Leuschner.

Context and connections with chapter 2:
This paper applied (and to the best of our knowledge established) the DIP+TV method [249, 265] for CT reconstruction, which is described in section 2.7.1. The benchmark with learned methods compares the reconstruction accuracy and data efficiency of several methods falling in the categories of directly trained post-processing reconstruction (section 2.1.1), end-to-end trained learned iterative reconstruction (section 2.5.1) and fully learned reconstruction (section 2.6), along with the unsupervised DIP+TV and classical methods.

Links:
https://iopscience.iop.org/article/10.1088/1361-6420/aba415 — Open access paper
https://github.com/oterobaguer/dip-ct-benchmark — Code and results
https://github.com/jleuschn/dival — DIV$\alpha\ell$: Utilities and reference method collection

Quantitative comparison of DL-based ... methods for ... CT applications [253]

Citation:
J. Leuschner, M. Schmidt, P. S. Ganguly, V. Andriiashen, S. B. Coban, A. Denker, D. Bauer, A. Hadjifaradji, K. J. Batenburg, P. Maass, and M. van Eijnatten. "Quantitative Comparison of Deep Learning-Based Image Reconstruction Methods for Low-Dose and Sparse-Angle CT Applications". In: *Journal of Imaging* 7.3 (2021). ISSN: 2313-433X. DOI: 10.3390/jimaging7030044

Description:
We compare eight deep-learning-based reconstruction methods and three classical reconstruction methods using two large-scale datasets (LoDoPaB-CT [252] and Apple CT [86]). It summarizes results from open challenges, which have been started during the Code Sprint 2020 online event on "Benchmarking Deep Learning based CT Image Reconstruction Methods".

Contributions:
Initiated by Maureen van Eijnatten, Carola Bibiane Schönlieb, Peter Maass and Kees Joost Batenburg, the Code Sprint 2020 event was planned and organized by Maureen van Eijnatten, Poulami Somanya Ganguly, Maximilian Schmidt and Johannes Leuschner. While the LoDoPaB-CT [252] dataset already existed, the Apple CT datasets [86] were created in preparation of the Code Sprint 2020 by Sophia Bethany Coban, Vladyslav Andriiashen, Poulami Somanya Ganguly, Maureen van Eijnatten and Kees Joost Batenburg, collaborating with the GREEFA company. Dominik Bauer and Amir Hadjifaradji took part in the challenges and contributed descriptions of their submitted methods. Johannes Leuschner coordinated the technical evaluation of the methods. In addition to external submissions, existing methods were included; some had already been implemented in our DIV$\alpha\ell$ library [251], others use code by Daniël Maria Pelt, Tianlin Liu, Alexander Denker and Sophia Bethany Coban. Maximilian Schmidt, Poulami Somanya Ganguly, Vladyslav Andriiashen, Sophia Bethany Coban, Alexander Denker, Maureen van Eijnatten and Johannes Leuschner undertook the comparative evaluation and presentation of results.

Context and connections with chapter 2:
This paper compares several deep learning methods described in chapter 2, more precisely directly trained post-processing reconstruction with different architectures (section 2.1.1), post-processing reconstruction using a normalizing flow (section 2.1.3), an end-to-end trained learned iterative reconstruction (section 2.5.1), a fully learned reconstruction method (section 2.6) and DIP+TV reconstruction (section 2.7.1).

Links:
https://www.mdpi.com/2313-433X/7/3/44 — Open access paper
https://lodopab.grand-challenge.org — Challenge website with public leaderboard
https://apples-ct.grand-challenge.org — Challenge website
https://zenodo.org/record/4460055 — Model parameter results
https://zenodo.org/record/4459962 — Reconstruction results for LoDoPaB-CT
https://zenodo.org/record/4459250 — Reconstruction results for Apple CT
https://github.com/jleuschn/learned_ct_reco_comparison_paper — Code and material
https://github.com/jleuschn/dival — Part of code and model parameter results

An educated warm start for DIP-based micro CT reconstruction [30]

Citation:
R. Barbano, **J. Leuschner**, M. Schmidt, A. Denker, A. Hauptmann, P. Maass, and B. Jin. "An Educated Warm Start for Deep Image Prior-Based Micro CT Reconstruction". In: *IEEE Transactions on Computational Imaging* 8 (2022), pp. 1210–1222. DOI: 10.1109/TCI.2022.3233188

Description:
We investigate a pretraining on synthetic data to initialize the U-Net weights of DIP+TV [249, 265, 23] for CT reconstruction. A speed-up and stabilization of the subsequent unsupervised deep image prior (DIP) optimization is demonstrated on real-measured 2D and 3D μCT data.

Contributions:
Riccardo Barbano, who brought up the idea, and Johannes Leuschner contributed equally by principally designing, implementing and conducting the experiments. Maximilian Schmidt and Alexander Denker helped by discussing and evaluating ideas. Andreas Hauptmann, Peter Maaß and Bangti Jin provided advice on the focus and presentation as well as support on technical and methodological aspects.

Context and connections with chapter 2:
This paper presents an extension of the DIP+TV applied to CT [23] (section 2.7.1), by pretraining the network to perform post-processing (section 2.1.1) on synthetic data.

Links:
https://ieeexplore.ieee.org/document/10003972 — Paper (non-open access)
https://arxiv.org/abs/2111.11926 — Preprint (accepted version)
https://github.com/educating-dip/educated_deep_image_prior — Code
https://zenodo.org/record/7234749 — Material and results
https://educateddip.github.io/docs.educated_deep_image_prior — Website

Uncertainty estimation for CT with a linearised DIP [14]

Citation:
J. Antoran, R. Barbano, **J. Leuschner**, J. M. Hernández-Lobato, and B. Jin. "Uncertainty Estimation for Computed Tomography with a Linearised Deep Image Prior". In: *Transactions on Machine Learning Research* (2023). ISSN: 2835-8856. URL: https://openreview.net/forum?id=FWyabz82fH

Description:
We propose a method to estimate uncertainty of deep image prior (DIP) reconstructions, based on a hierarchical prior model placed over the convolutional kernels. Utilizing a linearized network model that is expanded around the original DIP weights, we obtain a Gaussian predictive posterior, with the mean being the original DIP reconstruction. Hyperparameters of the prior model are obtained by marginal likelihood maximization, employing gradient estimation techniques to scale to real-measured 2D μCT data. While the predictive posterior covariance can be represented in closed form, sampling and patch-wise evaluation allow for scalable estimation.

Contributions:
Javier Antorán and Riccardo Barbano first devised the method, presented it as a symposium contribution [29] and contributed equally to this work. Johannes Leuschner joined the development to scale to higher resolutions. Model and methodology were primarily elaborated by Javier Antorán. Riccardo Barbano and Johannes Leuschner mainly developed the numerical implementation and conducted most of the experiments while conferring with Javier Antorán and Bangti Jin. All authors contributed to the development and presentation of the paper.

Context and connections with chapter 2:
This paper presents an uncertainty estimation framework (section 2.9) for the DIP+TV applied to CT [23] (section 2.7.1).

Links:
https://openreview.net/forum?id=FWyabz82fH — Open access paper and review
https://github.com/educating-dip/bayes_dip — Code
https://zenodo.org/record/7282279 — Material and results

Bayesian experimental design for CT with the linearised DIP [31]

Citation:
R. Barbano, **J. Leuschner**, J. Antorán, B. Jin, and J. M. Hernández-Lobato. *Bayesian Experimental Design for Computed Tomography with the Linearised Deep Image Prior*. Presented at ICML Workshop on Adaptive Experimental Design and Active Learning in the Real World (ReALML) 2022, July 22, Baltimore, MD, USA. 2022. DOI: 10.48550/ARXIV.2207.05714

Description:
We develop a method to select additional CT scanning angles based on a sparse pilot scan. A greedy angle selection strategy is employed that maximizes either the expected information gain or the expected squared error, informed by different image prior covariance models. Using the image prior covariance of a linearized deep image prior [14], improvements upon equidistant angle selection are obtained on a dataset with clear preferential directions.

Contributions:
Riccardo Barbano, Johannes Leuschner and Javier Antorán contributed equally. Javier Antorán elaborated the approach while Riccardo Barbano and Johannes Leuschner implemented the method, constructed the dataset, and designed and conducted the experiments. José Miguel Hernández-Lobato and Bangti Jin gave advice on the focus and methodology.

Context and connections with chapter 2:
This paper investigates Bayesian experimental design for optimized angle selection in CT based on a pilot scan, using an image covariance obtained from a linearized deep image prior [14] uncertainty model (section 2.9) for the DIP+TV (section 2.7.1).

Links:
https://arxiv.org/abs/2207.05714 — Preprint
https://github.com/educating-dip/bayesian_experimental_design — Code
https://zenodo.org/record/6635902 — Results

i.3 Notation and abbreviations

notation examples	meaning
$x^\top$, $X^\top$	transposition of a vector or matrix
$X_{:,0}$, $X_{3,:,:}$	selection of matrix or tensor slices, any dimension with index ":" is kept
$\vec{X}$	vectorization of a matrix, $X \in \mathbb{R}^{m,n} \Rightarrow \vec{X} = [X_{0,:}, \ldots, X_{m-1,:}]^\top \in \mathbb{R}^{m \cdot n}$
1D, 2D, 3D	one-dimensional/one dimension, two-dimensional/two dimensions, ...
w.r.t.	with respect to
i.i.d.	independent and identically distributed
$\approx$	approximately equals
$\ll$	much smaller than
$x := \ldots, \ldots =: x$	define x
$\mathcal{R}(A)$	range (image) of function A
$\oplus$	direct sum of spaces
$\perp$	orthogonal complement of a subspace
$\propto$	is proportional to
$\underset{x \in X}{\operatorname{argmin}} L(x)$	element $x \in X$ that minimizes $L(x)$ (assuming existence)
$\underset{x \in X}{\widetilde{\operatorname{argmin}}} L(x)$	element $x \in X$ found by "early-stopped" iterative minimization of $L(x)$
$\mathcal{L}(X, Y)$	space of linear maps between vector spaces X and Y
[00]	citation
[00]	citation of work that is included in the cumulative dissertation (part II)

abbreviation	meaning
AD	automatic differentiation
ADMM	alternating direction method of multipliers
ART	algebraic reconstruction technique
BCE	binary cross-entropy
CNN	convolutional neural network
CT	computed tomography
DEQ	deep equilibrium models
DIP	deep image prior
DRN	dilated residual network
FBP	filtered back-projection
FDK	Feldkamp-Davis-Kress
FFT	fast fourier transform
GAN	generative adversarial network
GCN	graph convolutional network
INR	implicit neural representations
MACE	multi-agent consensus equilibrium
MAP	maximum a posteriori
MAR	metal artifact reduction
MLP	multi-layer perceptron
MRI	magnetic resonance imaging
MSE	mean squared error
PET	positron emission tomography
PGD	projected gradient descent
PnP	Plug-and-Play
PSNR	peak signal-to-noise ratio
RED	regularization by denoising
ROI	region of interest
SGD	stochastic gradient descent
SSIM	structural similarity index measure
TV	total variation

Part I

Foundations and literature overview

Chapter 1

Inverse problems, computed tomography and deep learning

1.1 Inverse problems

Mathematical models usually aim to describe the effects that are observed given a certain cause. One is often interested in solving the *inverse problem*, which asks the opposite question: Given some observations, what has been the cause for it?

One class of inverse problems is concerned with physical systems modeled by partial differential equations, which together with a set of *parameters* determine the behaviour of the system. Here, the aim is to identify unknown parameter values based on measured quantities. The parameters can be of different kinds. They can, for example, specify the spatial distribution of a quantity at an initial time point, represent a boundary value, or represent a coefficient involved in the equations. Tomographic reconstruction constitutes another class of inverse problems. Here, the spatial distribution of a quantity is asked to be reconstructed from a set of measured line integral values of this quantity.

Solving an inverse problem can be challenging for various reasons. In practical applications, the observations are most often *noisy* measurements. If the solution depends on the observations in a sensitive way, then even minor uncertainty in the observations translates to major uncertainty in the solution. Another potential issue is *non-uniqueness*, meaning that multiple solutions can explain the observations equally well. For example, tomographic reconstruction of high-resolution images from an undersampled set of measurements (e.g. sparse-view or limited-view computed tomography [353]) involves this difficulty, meaning that the true solution cannot be recovered without ambiguity (unless incorporating additional information). In addition to noise perturbing the observations, the forward model might also be inaccurate, e.g., simplifying the true physical behaviour, which might be unknown or computationally intractable. Finally, solving inverse problems often requires difficult and costly numerical optimization.

Before considering practical solution approaches, we first recite a few mathematical considerations. For this purpose, the following, very abstract definition serves as a starting point.

Definition 1 (Exact inverse problem) *Assume the relation between a space of causes X and a space of observable effects Y is given exactly by a mapping $A : X \to Y$ ("forward model"). The*

task of finding the set $A^{-1}_{\text{set}}(y) := \{x \in X \mid A\,x = y\}$ *for given* $y \in Y$ *is called* ***exact inverse problem****.*

Note that this formulation is idealized for at least two reasons: (i) observations are usually perturbed, e.g. by measurement noise, and (ii) perfect knowledge of the model A is assumed.

For an exact inverse problem, existence and uniqueness of a solution are determined by the cardinality of the solution set $A^{-1}_{\text{set}}(y)$: One can distuingish the cases $A^{-1}_{\text{set}}(y) = \emptyset$ (no solution), $|A^{-1}_{\text{set}}(y)| = 1$ (unique solution) and $|A^{-1}_{\text{set}}(y)| > 1$ (multiple solutions).

The properties of the inversion function $A^{-1}_{\text{set}}(y)$ are commonly used to classify an inverse problem as either *well-posed* or *ill-posed*. Different definitions have been used in the literature, including those of Hadamard [159] and Nashed [303], which are stated in the following.

1.1.1 Ill-posedness

In practice one usually must expect y to be perturbed (e.g. a noisy measurement), so it seems natural to consider whether the solution (set) depends stably on y. Intuitively speaking, $A^{-1}_{\text{set}}(y)$ should be similar for the non-perturbed and the perturbed value of y.

Hadamard's definition of well-posedness requires the problem to be uniquely solvable for all $y \in Y$. This allows to describe the inversion with the mapping $A^{-1} : Y \to X$ such that $A^{-1}_{\text{set}}(y) = \{A^{-1}(y)\}$. Requiring A^{-1} to be continuous in a general topological sense completes the definition:

Definition 2 (Ill-posedness in the sense of Hadamard) *The exact inverse problem for a model* $A : X \to Y$ *between topological spaces* X *and* Y *is called* well-posed *in the sense of Hadamard [159] if the following conditions hold:*

1. *For all* $y \in Y$*, there exists* $x \in X$ *solving* $A\,x = y$. *(surjectivity of model* $A : X \to Y$*)*
2. *For all* $y \in Y$*, the solution of* $A\,x = y$ *is unique.* *(injectivity of model* $A : X \to Y$*)*
3. *The inverse mapping* $A^{-1} : Y \to X$ *is continuous.*

Otherwise (if any of these conditions is violated), the problem is called ill-posed *in the sense of Hadamard.*

An important subclass of inverse problems are those posed by a linear model $A \in \mathcal{L}(X, Y)$ between Hilbert spaces X and Y, which we will consider in the following. For finite-dimensional spaces X and Y, the generalized (Moore–Penrose) inverse [293, 320], denoted by A^+, addresses the problem of unique solvability by:

- generalizing the inverse problem to the minimization of observation error in the Y-norm $x^* \in L := \operatorname{argmin}_{x \in X} \|A\,x - y\|_Y$ (allowing for approximate solutions if $A\,x \neq y\ \forall x \in X$);
- choosing the minimum norm solution $x^* = \operatorname{argmin}_{x \in L} \|x\|_X$ (resolving possible ambiguity).

The notion of generalized inverses has been generalized to linear operators between general Hilbert spaces [384, 43, 47, 48, 117], which is somewhat more involved, e.g. requiring the domain of A to be decomposable as the direct sum of the null-space and the carrier of A, and only allowing the generalized inverse to be defined for $y \in \mathcal{R}(A) \oplus \mathcal{R}(A)^\perp \subseteq Y$ [43, 117].

Being able to uniquely solve the generalized inverse problem, there remains the question whether the generalized inverse A^+ is stable (analogously to condition 3 in definition 2). The

following definition of well-posedness by Nashed identifies a generalized linear inverse problem to be well-posed if and only if A^+ is continuous (or, equivalently, bounded), via the equivalent condition of the range of A being closed [184]:

Definition 3 (Ill-posedness in the sense of Nashed) *Consider the generalized inverse problem for a linear model $A : X \to Y$ between Hilbert spaces X and Y. The problem is called* well-posed *in the sense of Nashed [303] if the range of A is closed; otherwise the problem is called* ill-posed *in the sense of Nashed.*

If A has an inverse A^{-1}, it coincides with A^+, and thus ill-posedness in the sense of Nashed implies ill-posedness in the sense of Hadamard due to condition 3 in definition 2.

An inverse problem with finite-dimensional range of the linear model A is always well-posed in the sense of Nashed; thus problems that are discretized in finite dimensions—like it is usually done in order to solve an inverse problem numerically—are well-posed, regardless of potential ill-posedness of the original analytical problem. However, the solution can still depend sensitively on the (noisy) observations if A has small singular values (i.e., the problem is ill-conditioned), and the term "ill-posed" is also commonly used for such problems [163]. In general, the singular value decay of A is informative about the degree of ill-posedness: A problem is commonly called mildly ill-posed if the decay is polynomial, and severely ill-posed if the decay is exponential [115, 21, 184].

In the remainder of section 1.1, we stick to the assumption of X and Y being Hilbert spaces, which also means that they have norms $\|\cdot\|_X$ and $\|\cdot\|_Y$ induced by their scalar products.

1.1.2 Regularization

The aforementioned instabilities arising in inverse problems cause a need for regularizing techniques. Even if the model does not allow for exact and robust inversion (e.g., due to noise that would be amplified excessively, or because of a non-injective model A), one can still aim for a robust inversion method that approximates the true solution. Regularization can be accomplished in different ways.

Instead of only minimizing the observation error, *variational regularization* adds a regularization term to the objective function. Classical examples for regularization terms are the ℓ_2 norm (known as Tikhonov regularization [228]), which promotes small solution values, and total variation (TV regularization [335, 331]), which promotes solutions with small gradients. With iterative methods, such as the Landweber iteration [162], *early stopping* is another effective way to regularize the solution, and can in fact be related to variational regularization [237, 148, 422]. Further regularization strategies include approximate analytic inversion and discretization as regularization—see [45] and [21, sec. 2.4] for more detailed overviews.

Studying the regularized solution of inverse problems can be approached from different view-points. We will briefly depict both the classical deterministic and the statistical approach.

Deterministic approach

In classical regularization theory, the observations $y^\delta \in Y$ are assumed to contain noise that is bounded by some constant δ,

$$y^\delta = A\,x + \epsilon, \quad \|\epsilon\|_Y \leq \delta. \tag{1.1}$$

For this setting, a regularization method is formed by a family of continuous operators parameterized by a regularization parameter, along with a parameter choice function depending on

the noise level δ and optionally also on y^δ, that yields approximate solutions converging to the minimum norm solution as $\delta \to 0$ [115].

Example 1 *A Tikhonov regularization can be formalized as the operator family* $\{R_\alpha\}_{\alpha>0}$*,*

$$R_\alpha : Y \to X,\ y^\delta \mapsto \operatorname*{argmin}_{x \in X} \|A\,x - y^\delta\|_Y^2 + \alpha\|x\|_X^2, \tag{1.2}$$

with a suitable choice function for the parameter α*, e.g. any continuous monotonously decreasing function* $\delta \mapsto \alpha(\delta)$ *fulfilling* $\alpha(\delta) \to 0$ *and* $\delta^2/\alpha(\delta) \to 0$ *for* $\delta \to 0$ [116].

A parameter choice function that does not depend on y^δ, but only on δ, is called an *a priori* choice, while a choice that takes y^δ into account is called *a posteriori*. A well-known a posteriori choice was proposed by Morozov [294]: Since the true solution x has a residual norm $\|A\,x - y^\delta\|_Y$ of up to δ (see eq. (1.1)), it appears reasonable to choose the regularization parameter α such that the regularized solution $R_\alpha(y^\delta)$ has a residual norm of similar magnitude, $\|A\,R_\alpha(y^\delta) - y^\delta\| \approx \delta$.

Classical theory also studies convergence rates and the stability of regularization methods. For example, a variant of Morozov's discrepancy principle yielding a Tikhonov regularization with order-optimal convergence is proposed in [114], and [299] studies the best stability that can be achieved depending on the regularity that is assumed for the solution. Constraints on the set of possible solutions, such as regularity, are called source conditions, and provide the basis for proofs of convergence rates [58, 122, 21]. Convergence results for total variation (TV) regularization of linear inverse problems are shown e.g. in [196].

Statistical approach

So far, all variables and the model have been considered to be deterministic, with the cause x and noise ϵ being unknown. In the statistical approach, one instead models the inverse problem with random variables. More precisely, x is assumed to follow a *prior* distribution with density $p(x)$ modeling information about x that is available before observing y^δ, and the dependence of y^δ on x is modeled in terms of the conditional probability distribution with density $p(y^\delta \,|\, x)$, which is also called *likelihood*. See [210] for a more detailed introduction to statistical inversion.

Solving the inverse problem in the statistical sense extends to recovering the full posterior distribution with density $p(x \,|\, y^\delta)$, i.e., estimating probabilities for all possible solutions x given an observation y^δ. In contrast, the deterministic approach reports a single estimate for x. From an estimated posterior distribution, one can obtain point estimates, such as its maximum, called the maximum a posteriori (MAP) estimate, but also other statistical quantities, such as its mean, known as the conditional mean, as well as interval estimates. To estimate some quantities, e.g. the conditional mean and standard deviation, it is sufficient to sample from the estimated posterior distribution, so an explicit formulation of the estimated density is not necessarily required. For simplicity, we only consider finite-dimensional Hilbert spaces $X = \mathbb{R}^n$ and $Y = \mathbb{R}^m$ in the following.

Bayes' rule [37, 195] provides us with the relation

$$p(x \,|\, y^\delta) = \frac{p(x)\,p(y^\delta \,|\, x)}{p(y^\delta)},$$

stating that the posterior density for given y^δ is proportional to the product of prior density and likelihood, normalized by the marginal density $p(y^\delta)$, which must be non-zero and can be disregarded for many purposes as a constant that does not depend on x.

Comparing with the deterministic approach, the likelihood function $p(y^\delta \,|\, x)$ models both the mapping from X to Y and the noise, while the prior density $p(x)$ plays a regularizing role. In

fact, many classical regularizations have a statistical interpretation; e.g. the Tikhonov functional that is minimized in eq. (1.2) can be motivated as follows:

Example 2 *Consider finite-dimensional Hilbert spaces $X = \mathbb{R}^n$ and $Y = \mathbb{R}^m$. Without knowledge of y^δ, we assume the cause x to follow a normal distribution $x \sim \mathcal{N}(0, \sigma_x^2 I_n)$. Consider a mapping $A : X \to Y$ and additive normal-distributed noise $\epsilon \sim \mathcal{N}(0, \sigma_y^2 I_m)$ for the observation model, leading to the conditional probability distribution $y^\delta \,|\, x \sim \mathcal{N}(A\,x, \sigma_y^2 I_m)$. For the prior density and likelihood function we obtain*

$$p(x) \propto e^{-\|x\|_X^2/(2\sigma_x^2)}$$
$$p(y^\delta \,|\, x) \propto e^{-\|y^\delta - A\,x\|_Y^2/(2\sigma_y^2)}$$

and hence for the posterior density:

$$p(x \,|\, y^\delta) \propto p(x)\, p(y^\delta \,|\, x) = e^{-\|x\|_X^2/(2\sigma_x^2) - \|y^\delta - A\,x\|_Y^2/(2\sigma_y^2)}.$$

The resulting MAP estimate, which is found by minimizing the posterior negative log-likelihood $-\log(p(x \,|\, y^\delta))$, coincides with the Tikhonov-regularized solution $R_\alpha(y^\delta)$ with $\alpha = \sigma_y^2/\sigma_x^2$ (see eq. (1.2)).

Many other minimization functionals for variational regularization can be motivated in the same way, i.e. by summing the negative log-likelihoods of the prior $p(x)$ and the observation model $p(y^\delta \,|\, x)$ [21].

The discrepancy principle of Morozov, which we previously discussed in the context of choosing the regularization parameter in a deterministic approach, can also be transferred to statistical inverse problems, e.g. to define stopping criteria for iterative solvers [50]. More vaguely speaking, the discrepancy principle says to solve the problem only up to the given accuracy of observations, where the accuracy is given by noise bounds in the deterministic approach or instead by properties of random distributions (such as the noise level σ_y in example 2) in the statistical approach.

1.2 Computed tomography

Obtaining images visualizing the interior of a subject or object in a non-invasive way is important in many applications. Especially for medical diagnostic, imaging is a crucial tool. Various imaging modalities exist, with the most prominent ones being based on X-ray attenuation, magnetic resonance, nuclear emission, or ultrasound. With the present section, we give a brief introduction to X-ray computed tomography (CT), focusing on reconstruction techniques and challenges. Besides medical applications, X-ray CT is useful for many other applications in industry, engineering and science [401]. Our introduction to X-ray CT is, among others, based on the book [60] (see also the more compact chapter [61]).

In the 1970s, the first medical X-ray CT scanners became commercially available, making it the first tomographic modality applied to human bodies. While a single X-ray radiograph projects the attenuation information of a volume onto a 2D plane, thus leading to superposition of the structures at different depths, computed tomography combines acquisitions from multiple angles to reconstruct images with significantly higher contrast. A source emitting X-ray quanta rotates around the subject, and the radiation after passing through the subject is detected on the opposite side in order to quantify the attenuation. Since the third generation of CT scanners, a sufficiently large fan-beam of X-rays emitted by the source can be captured by the detector simultaneously,

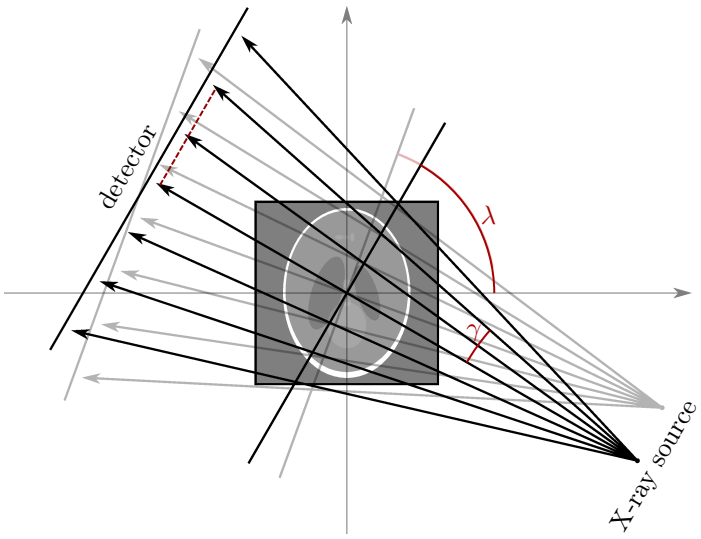

Figure 1.1: Illustration of a 2D fan-beam CT geometry.

allowing for a continuous rotation of the source while acquiring projections. Figure 1.1 illustrates a 2D fan-beam geometry: For different angles λ, the attenuation at multiple detector pixels is acquired, corresponding to different X-ray paths (parameterized by γ).

Given the measurements of a CT scan, a tomographic reconstruction problem needs to solved in order to obtain the CT image. Clearly, CT reconstruction is an inverse problem, where the model describes the scanning process, the observation is formed by detector measurements, and the unknown quantity to be recovered is an image that indicates the X-ray attenuation at each location in the subject. The whole scanning process is non-deterministic, mostly due to the emission, transmission and detection of X-ray quanta being statistical in nature [60]. Ideally, a model should incorporate the various stochastic effects (including ones that are more challenging to simulate, such as Compton scattering [276]). However, for simplicity and tractability, one typically formulates a model consisting of a deterministic forward model and a noise model, chained sequentially.

In a simplified model of CT scanning, the deterministic forward model performs integration over lines corresponding to the paths of X-ray quanta traveling from the source to the detector. First, we assume a monochromatic X-ray source, in which all quanta have the same energy level. For each line L, Beer-Lambert's law of attenuation relates the detected intensity I to the intensity without attenuation I_0 (i.e., if vacuum would be scanned) via

$$I = I_0 \exp\left(-\int_{p\in L} x(p)\,\mathrm{d}p\right),$$

where $x(p)$ denotes the linear attenuation coefficient at location p. By isolating the integral over x on the right-hand side, the linear equation

$$-\log\left(\frac{I}{I_0}\right) = \int_{p\in L} x(p)\,\mathrm{d}p \tag{1.3}$$

is obtained, which is more convenient in practice. The log-transformed measurements $-\log(I/I_0)$ are called post-log data, and they are often computed in a pre-processing step in order to linearize the model.

In a CT scan, intensities are acquired for a set of lines through the interior of the scanner. E.g., for a 2D image space, the fan-beam geometry depicted in fig. 1.1 presents one such set of lines, parameterized by λ and γ:

$$G_{\text{fan}} := \{L(\lambda,\gamma) \mid \lambda \in [0,2\pi), \gamma \in [-\gamma_{\max}, \gamma_{\max}]\}, \quad L(\lambda,\gamma) := \{p_{\text{src}}(\lambda) + t\, d(\lambda,\gamma) \mid t > 0\},$$
$$p_{\text{src}}(\lambda) := R_{\text{src}} \begin{pmatrix} \cos\lambda \\ \sin\lambda \end{pmatrix}, \quad d(\lambda,\gamma) := \sin\gamma \begin{pmatrix} -\sin\lambda \\ \cos\lambda \end{pmatrix} - \cos\gamma \begin{pmatrix} \cos\lambda \\ \sin\lambda \end{pmatrix},$$

where $R_{\text{src}} > 0$ is the radius at which the source rotates around the origin and $\gamma_{\max}$ denotes the maximum fan angle (cf. [98]). Intensities are measured for each line $L(\lambda,\gamma)$ in the geometry. One typically assumes that the subject is contained in a region that is fully covered by the fan, and that the attenuation coefficient $x(p)$ for points p outside this region is zero. Thus, the integration over L in eq. (1.3) can be restricted to the line segment inside this region. The idealized, linear and noise-free forward model of CT is then given by the operator that maps x to the post-log intensities (eq. (1.3)) for all lines in the scanning geometry. For example,

$$A_{\text{fan}} : X \to Y, \quad [A\,x](\lambda,\gamma) := \int_{p \in L(\lambda,\gamma)} x(p)\,\mathrm{d}p$$

describes the forward operator for the fan-beam geometry. In an analytical setting, the projection space Y and the image space X are function spaces over subsets of $[0,2\pi)\times\mathbb{R}$ and $\mathbb{R}^2$, respectively, with some regularity assumptions being made in order to obtain a well-defined, invertible reconstruction problem [304]. In practice, the measurements are obtained for a finite number of source angles λ_i, $i = 0,\dots,k-1$ and a finite number of detector pixels γ_j, $j = 0,\dots,l-1$, and one aims to obtain a discrete reconstruction with $r \times r$ pixels; thus, the elements from X and Y are represented by vectors in $\mathbb{R}^{r\cdot r}_{\geq 0}$ and $\mathbb{R}^{k\cdot l}_{\geq 0}$, respectively, and their linear relation A is represented by a matrix in $\mathbb{R}^{k\cdot l \times r\cdot r}_{\geq 0}$, whose entries specify the intersection of each line with each pixel. We examplarily used a 2D fan-beam geometry to illustrate the definition of a forward model, but other geometries can be defined similarly; e.g., a standard 3D cone-beam geometry definition only differs in using a 2D detector with an additional fan angle $\gamma_\perp$ perpendicular to the rotation plane of the source and considering a 3D reconstruction space.

Mathematical analysis often considers the parallel-beam geometry, which consists of parallel lines from multiple source positions for each rotation angle; in this case, A is the Radon transform, whose inversion is presented in [325]. Image space and projection space are related in a more elementary way for the Radon (parallel-beam) transform, but extensions exist for the transforms of other geometries; see [304] for functional analysis background on CT transforms and reconstruction.

1.2.1 Basic theory

In order to recite a few classical results, let us define the 2D parallel-beam geometry

$$G_{\text{Radon}} := \{L(\alpha,s) \mid \alpha \in [0,\pi), s \in \mathbb{R}\}, \quad L(\alpha,s) := \left\{ s \begin{pmatrix} \cos\alpha \\ \sin\alpha \end{pmatrix} + t \begin{pmatrix} -\sin\alpha \\ \cos\alpha \end{pmatrix} \,\middle|\, t \in \mathbb{R} \right\}$$

and the Radon transform integrating over each of the lines in G_{Radon},

$$A_{\text{Radon}} : X \to Y, \quad [A_{\text{Radon}}\,x](\alpha,s) := \int_{p \in L(\alpha,s)} x(p)\,\mathrm{d}p,$$

where X denotes the absolutely integrable functions on $\mathbb{R}^2$ (commonly, X and Y are chosen to be Schwartz spaces on $\mathbb{R}^2$ and $[0,\pi)\times\mathbb{R}$, respectively). The most central result is the projection-slice theorem, also called Fourier-slice theorem, which states that the 1D Fourier-transformed

parallel-beam projection values from an angle equal the 2D Fourier-transformed image values along the line with this angle:

Theorem 1 (Projection-slice theorem) *Let F_{1D} and F_{2D} denote the Fourier transform for functions on $\mathbb{R}$ and $\mathbb{R}^2$, respectively. Assuming that all involved integrals exist, the equality*

$$\big[F_{\mathrm{1D}}\big([A_{\mathrm{Radon}}\, x](\alpha, \cdot)\big)\big]\,(r) = [F_{\mathrm{2D}}\, x](r\cos\alpha, r\sin\alpha)$$

holds for all angles $\alpha \in [0, \pi)$ and frequencies $r \in \mathbb{R}$.

A derivation of this result can be found in [211], and an elegant proof for n-dimensional Radon and X-ray transforms between Schwartz spaces is given in [304].

The projection-slice theorem already implies a direct analytical inversion formula for the reconstruction of x from $y = A_{\mathrm{Radon}}\, x$, which can be described as three steps:

1. Apply the 1D Fourier transform w.r.t. the detector coordinate s, i.e. $y(\alpha, s) \mapsto [F_{\mathrm{1D}}(y(\alpha, \cdot))](r) =: \hat{y}(\alpha, r)$.
2. Apply the coordinate transform from polar to Cartesian coordinates $(u, v) = (r\cos\alpha, r\sin\alpha)$, which by theorem 1 yields $\hat{y}(\alpha, r) = [F_{\mathrm{2D}}\, x](u, v) =: \hat{x}(u, v)$.
3. Apply the inverse 2D Fourier transform, which recovers $F_{\mathrm{2D}}^{-1}\, \hat{x} = x$.

In practice, Fourier-based direct reconstruction would bear difficulties due to the polar-to-Cartesian coordinate transform with the regular discrete sampling of angles and detector pixels. Since the measurements y are usually sampled on a rectangular grid of angles α and detector pixels s, they only determine the values of $F_{\mathrm{2D}}\, x = \hat{x}$ on a polar grid, whose points have increasing distance for higher frequencies. In order to apply a standard inverse 2D fast fourier transform (FFT), these values need to be interpolated on a rectangular grid of frequencies u, v (called "regridding").

Instead, a different direct inversion formula, called filtered back-projection (FBP), forms the basis for a class of commonly used reconstruction formulas. The FBP, which may also be derived using the projection-slice theorem, combines a filtering step and a simple back-projection step that distributes each value of the projection space uniformly over the line along which it has been integrated according to the forward model. Mathematically, simple back-projection for A_{Radon} coincides (up to a constant scaling factor) with the adjoint of A_{Radon},

$$A_{\mathrm{Radon}}^* : Y \to X, \quad [A_{\mathrm{Radon}}^*\, y](p_0, p_1) = \frac{1}{\pi} \int_{\alpha=0}^{\pi} y(\alpha, p_0\cos\alpha + p_1 \sin\alpha)\, \mathrm{d}\alpha$$

(see e.g. [306] for a proof that this is the adjoint). The filtering step is most easily expressed in the 1D Fourier space of the projections as the multiplication with the absolute frequency $|r|$:

Theorem 2 (Filtered back-projection) *Given a function x over $\mathbb{R}^2$ for which all involved integrals are well-defined, its Radon transform $y = A_{\mathrm{Radon}}\, x$ relates to the original function x by*

$$x = \frac{1}{2} A_{\mathrm{Radon}}^*\, h_{\mathrm{ramp}}\, y,$$

where $h_{\mathrm{ramp}} = F_{\mathrm{1D}}^{-1} \circ \big[\hat{y}(\alpha, \cdot) \mapsto |\cdot|\, \hat{y}(\alpha, \cdot)\big] \circ F_{\mathrm{1D}}$ is the ramp filter, which point-wise multiplies $\hat{y}(\alpha, r)$ with the absolute frequency $|r|$.

A proof of this theorem is found e.g. in [119, Theorem 6.2].

Alternatively to the projection-domain filtering, the simple back-projection $A_{\mathrm{Radon}}^* y$ can be deconvolved in the image domain, which is known as the filtered layergram inversion formula [60].

1.2.2 Reconstruction

The inverse problem of CT reconstruction in the analytical setting is ill-posed in the sense of Nashed, since the range of A is non-closed, since A is a compact linear operator with infinite rank on a bounded domain [209], meaning that the generalized inverse A^+ is not continuous. The singular values of the Radon transform tend to 0, which makes the inversion unstable. However, the singular value decay is rather slow, and the inversion of 2D and 3D Radon and X-ray transforms is classified as modestly ill-posed in [304]. For a practical discretized reconstruction—as already noted in the final paragraph of section 1.1.1—Nashed's notion of ill-posedness does not apply, but the problem is ill-conditioned, so the inversion is sensitive to noise. Reconstruction from undersampled projections is especially difficult, since the linear system is underdetermined.

CT reconstruction is possible with many different approaches. The previously stated FBP (theorem 2) points out a direct, efficient way for parallel-beam geometries. However, parallel-beam geometries are seldomly used in practice. FBP type and other analytical inversion formulas also exist for fan-beam and cone-beam geometries, but are more involved; see e.g. [98]. A popular approximate method for cone-beam reconstruction similar to FBP is the Feldkamp-Davis-Kress algorithm [120]. Data from commonly used geometries can also be rebinned to other geometries that allow for easier reconstruction; for example, the 3D cone-beam data can be rebinned to parallel-fan-beam data, which uses a 2D fan-beam geometry for each slice. In the FBP (and similar inversion formulas), the ramp filter emphasizes high frequency components, it presents an unstable inversion and amplifies noise. For stabilization, the ramp filter is commonly combined with a low-pass filter [39, 209], e.g. a Hann or cosine window filter. This effectively regularizes the inversion, which then is only approximate but more stable.

Alternative to direct (approximate) inversion formulas, iterative methods form a powerful group of approaches for CT reconstruction. Iterative methods are designed to solve optimization problems, e.g., minimizing a variational regularization objective,

$$x^* \in \operatorname*{argmin}_x D(A\,x, y^\delta) + \alpha R(x), \tag{1.4}$$

where D measures the data discrepancy and R is a regularization term, weighted by a parameter $\alpha \geq 0$. Iterative reconstruction offers great modeling freedom w.r.t. the forward model A, noise statistics guiding the choice of D, and regularizing prior knowledge. Different optimization schemes can be employed. For example, a constrained formulation of the total variation (TV) reconstruction problem is solved using projection onto convex sets and steepest descent with adaptive step-size (ASD-POCS) in [352], and Chambolle-Pock algorithms are applied to a TV-regularized objective in the form of eq. (1.4) with $D(A\,x, y^\delta) = \|A\,x - y^\delta\|_1$ in [377]. Simple, yet effective optimization of eq. (1.4) is also possible with a variant of gradient descent, such as the Adam optimizer [221]. As already mentioned in section 1.1.2, besides explicit regularization, iterative methods can induce regularization when stopped early, aiming to find a point in the optimization at which image details are sufficiently reconstructed while part of the noise has not yet been fitted. Often, a reconstruction by e.g. FBP is used to initialize the iterative process, which leads to faster convergence, and when using early stopping can also act like a prior.

We now give a brief overview of iterative reconstruction approaches, following the review in [42], which is organized by the different involved modeling aspects. The discretized forward model A is the minimum of modeling used by all iterative approaches. A classical family of iterative approaches is based on the algebraic reconstruction technique (ART), which applies Kaczmarz's method to solve the (in practice only approximate) linear system of equations $A\,x = y^\delta$ [304]. While original ART [150] updates the result only by the information of a single measurement (i.e. one row of the linear system of equations) in each iteration, the simultaneous ART (SART) [11] uses all measurements from one source angle at a time, and the simultaneous iterative

reconstruction technique (SIRT) [141] uses all measurements from all angles in each update step. Updates of ART, SART and SIRT are additive, but there are also multiplicative variants of ART (MART) [150]. ART-based approaches model the discretized acquisition geometry more precisely than FBP-based inversion formulas. The sequential use of groups of measurement data, e.g. in SART, is generalized in a concept called ordered subsets (OS). In OS-SIRT [416], projections from groups of multiple angles are used, thus falling in between SART and SIRT. Ordered subsets are also used in combination with other iterative methods discussed in the following. Statistical reconstruction also includes a noise model, which is mainly determined by the statistical process of photon emission, attenuation and detection, which can be modeled jointly with one Poisson distribution [60]. For ultra-low-dose scans, electronic noise also becomes relevant, which is usually modeled as Gaussian noise. If the measurements are log-transformed in order to obtain a linear model (eq. (1.3)), this modulates the noise statistics, and the resulting post-log noise is often approximated as Gaussian [269]. When using ultra-low dose, photon starvation occurs, which cannot be accurately modeled in post-log domain; some statistical methods for ultra-low-dose CT therefore use a non-linear pre-log model [126]. The optimization for statistical reconstruction can be realized e.g. with maximum-likelihood expectation maximization (ML-EM) [234, 277], convex maximum a posteriori (MAP) optimization [236, 235, 40], or based on iterative coordinate descent (ICD) [338, 380, 46]. Additional modeling aspects include further refinement of the geometry [380, 79], taking into account the non-infinitesimal extent of the source and/or the detector pixels, additional physical effects, such as the non-linear effects of polyenergetic X-rays [112] and scattering. Improving advanced aspects of the model can improve reconstruction quality and avoid artifacts (see section 1.2.3) in the first place, but comes with increased computational cost and may require additional knowledge. However, the flexibility to use sophisticated, more accurate models is the main strength of iterative reconstruction, which therefore is often referred to as model-based iterative reconstruction (MBIR). Regularization with prior information about the image can support the reconstruction. Besides the already mentioned TV regularization, which encourages image gradient sparsity, other regularizations include an edge-preserving variant of TV [381], bilateral filters and non-local means [417].

1.2.3 Reconstruction artifacts

Reconstructions contain structured errors, called artifacts, due to many factors, ranging from the practical acquisition to inaccurate modeling. We shortly describe some of the artifacts, which are discussed in more detail e.g. in [60, 341]. Most of the artifacts are inevitable with FBP-based reconstruction, whereas suitable (model-based) iterative reconstruction can alleviate some of the following problems.

As detector pixels and the source focal spot have a non-infinitesimal size, only an average intensity for a collection of rays can be measured at each detector pixel. This leads to blurring, and also streaking artifacts at paths for which the detector pixel averages very different intensities (i.e. paths tangential to edges), because the attenuation, if assumed to be uniform across the detector pixel, is underestimated due to the logarithmic dependence of attenuation and intensity. These artifacts are called *partial volume artifacts*, and the underestimation at edges is also called *edge effect*. Another cause of artifacts is the physical effect of *beam hardening*: In practice, the X-ray quanta have different energies and are attenuated differently depending on the energy and attenuating material, leading to a non-linear acquisition. Since lower energy X-rays experience stronger attenuation in general, the spectrum of transmitted X-rays is shifted to higher energies, giving rise to the name beam hardening. In the reconstruction from the intensity measurements of a standard detector that integrates over the X-rays of all energies, this leads to streaking and cupping artifacts, where cupping artifacts can be more easily compensated for by calibration,

while streaking artifacts, which occur at rays through high-density areas, are typically more persistent. Due to the effect of Compton scattering (which actually contributes to the attenuation mechanism of CT), some X-ray quanta change their energy, do not travel along a straight line and may hit the detector at a different location. As the standard model is based on direct radiation only, the scattered radiation adds noise to the signal. For detector areas that receive very little radiation due to high attenuation of the direct X-rays, the scattering noise is strong compared to the signal. This can lead to *scattering artifacts* in the form of streaks through areas of high attenuation. If metal is present in the imaged volume, it leads to severe *metal artifacts* due to the strong attenuation (or blocking) of rays that pass through the metal pieces, caused by beam hardening, scattering and the edge effect, among others. *Motion artifacts* occur when the subject being imaged performs movements, such as breathing or heart beats. Reducing scanning times helps to reduce these artifacts. While there are technical limitations, current scanners perform rotations in less than a second. With very fast scanning, *detector afterglow* may become relevant, i.e., the activation of detector pixels that lasts for a short time frame can affect subsequent acquisitions, and thus lead to artifacts unless a correction is employed. *Sampling artifacts*, also called aliasing artifacts, may occur when the measurements are undersampled, i.e., the condition of Shannon is not satisfied [180].

1.2.4 Reconstruction purpose and challenges

CT reconstructions usually serve a down-stream task, e.g. a diagnosis or a segmentation. Hence, the ultimate goal of improving reconstruction quality is to facilitate reliable extraction of the features relevant for this task, by a human inspector and/or a machine.

Appropriate CT imaging includes several challenges, driven by practical demands and potentials. In medical imaging, the radiation dose needs to be kept as low as possible, since it carries a risk of causing of cancer [54]. A low dose is most directly achieved by using a low radiation intensity, which leads to an increased noise-to-signal ratio. Alternatively, a fast scanning that only acquires projections from few (equidistant) source angles, called sparse-view CT, also leads to dose reduction at the cost of undersampling the measurements; short scanning times are also desirable in both medical imaging and industrial applications [444, 59, 453], e.g. to reduce motion artifacts and for economical reasons. In some applications, projections can be acquired only from a limited angular range [123], called limited-view CT, which is particularly difficult due to the contiguous missing projection part and constitutes a severely ill-posed inverse problem due to its ill-conditioning [93, 304].

Challenging, practically founded conditions thus motivate the development of reconstruction algorithms that are capable of sufficiently accurate and artifact-reduced reconstruction for the given purpose.

1.3 Deep learning

Using artificial deep neural networks as models for machine learning, which is then called *deep learning*, has become the predominant approach in many complex applications, such as speech and image recognition, natural language processing and autonomous driving to name a few.

An artificial neural network can be abstractly represented by a parameterized mapping $y = f_\theta(x)$, $\theta \in \mathbb{R}^p$, that computes outputs $y \in Y = \mathbb{R}^{d_{\text{out}}}$ from inputs $x \in X \subseteq \mathbb{R}^{d_{\text{in}}}$. The parameters θ are learned in order to solve a certain task by an optimization procedure. Typically, the learning relies on large datasets on which the network is trained, but there exists a variety of forms:

- Supervised learning employs a dataset of pairs $(x_i, y_i^*)_{i=0,1,\dots,N-1}$ of inputs and desired outputs. The outputs y_i^* are also called "labels" (such as the class in classification or segmentation tasks [164], but the term is also used with regression tasks [393]).
- Semi-supervised learning utilizes both a small dataset of pairs $(x_i, y_i^*)_{i=0,1,\dots,M-1}$ and a large dataset of unlabelled data $(x_i)_{i=0,1,\dots,N-1}$, $M \ll N$ (see [421] for a survey).
- Unsupervised learning only has access to unlabeled data, such as a dataset of inputs $(x_i)_{i=0,1,\dots,N-1}$ (e.g. clustering [286] or Noise2Inverse [179]), a dataset of desirable samples $(y_i^*)_{i=0,1,\dots,N-1}$ (e.g. generative models [383]), two independent (i.e. non-paired) datasets of inputs and desirable outputs (e.g. CycleGAN [450]), or in some cases just a single data instance x_0 (e.g. deep image prior [249]).
- Reinforcement learning lets a virtual agent learn by trial and error, so the data is generated during the learning process by interacting with the environment (see [22] for a survey).

In this thesis we will consider supervised and unsupervised deep learning in the context of CT reconstruction.

1.3.1 History of neural networks

Artificial neural networks, or short *neural networks*, have been studied since the 1940's. The neurophysiologist and cybernetician Warren S. McCulloch and the logician Walter H. Pitts developed a calculus inspired by the principles of biological neural networks [282]. Donald O. Hebb proposed the Hebbian learning mechanism, which is a form of unsupervised learning based on the way neural connections are strengthened in human brains [173]. The basic perceptron, very roughly modeling biological neurons, was invented by Frank Rosenblatt [334]. Alekseĭ Ivakhnenko created the group method of data handling (GMDH), constituting the first working multi-layer neural networks (although named differently at that time) [200, 199, 339]. Despite some early successful applications, such as MADALINE [399] suppressing echoes on a telephone line, neural network research slowed down. The study of perceptrons halted in 1969 when Marvin Minsky and Seymour Papert showed limitations of basic forms [288], like the inability to learn the XOR boolean operation with a single-layer perceptron; moreover the computational hardware resources available at that time were insufficient for learning large neural networks. Instead, research temporarily focused more on symbolic artificial intelligence [367].

In the 1980's, neural networks research was revived as *connectionism* became popular again, mainly in the form of parallel distributed processing [281]. Core principles consisted in connecting many simple units—like artificial neurons—to a large network capable of learning complex tasks, and in the *distributed representation* within the network [281, 148]. One significant (re-)discovery from this time was the gation algorithm allowing for efficient learning with deep neural networks [263, 336]. Early successful applications include protein structure prediction [323], handwritten ZIP code and alphabet recognition [240, 441] as well as medical image analysis [440, 439]. Convolutional neural network (CNN) architectures were already used for visual tasks at this time, starting with the neocognitron [127, 148], which in turn is based on research by David Hubel and Torsten Wiesel on the visual cortex of mammals [194]. Limited computational resources combined with some unsolved problems, like vanishing gradients [182], delayed the further development of deep learning. Nevertheless, both methodological advances, such as the long short-term memory (LSTM) architecture [183] and the efficient pretraining of deep belief networks [181], and increasing availability of computational power and large datasets finally enabled the highly successful application in various fields. Several challenge competitions were won using deep neural networks in the early 2010's, among others in computer vision, leveraging the accelerated training

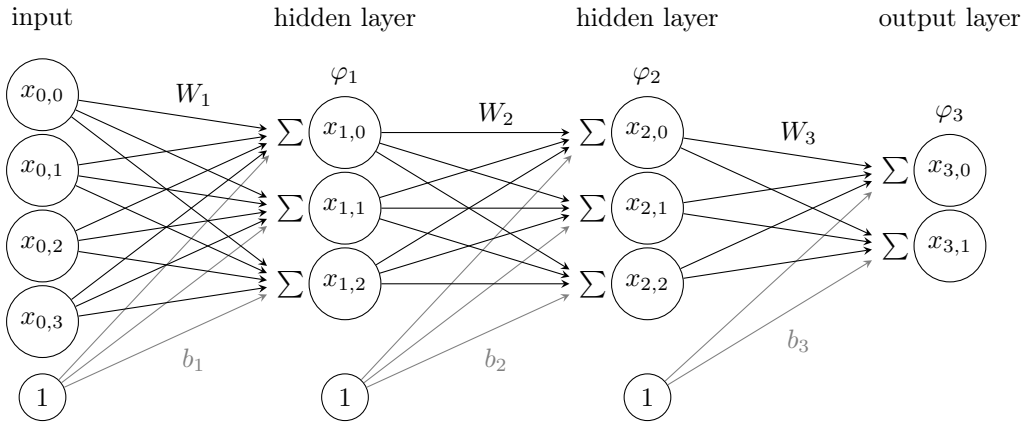

Figure 1.2: A multi-layer perceptron (MLP), which is a fully connected feedforward neural network. W_i and b_i are the weights and biases of the i-th layer, and φ_i is the activation function (e.g. ReLU : $x \mapsto \max(0, x)$) applied in each neuron $x_{i,j}$, $j = 0, 1, \ldots$ of the layer.

with graphics processing units (GPUs) [309, 158]. Today, deep learning is a very active area of research with many applications [348]. Although artificial neural networks are inspired by neuroscience, they are now independently utilized as computational models, without the aim to model biological neural networks [148].

1.3.2 Basic principles of neural network

Artificial neural networks are typically structured in uniform *layers* of neurons. The activations of all neurons in a layer form an activation vector $x_k \in \mathbb{R}^{d_{\text{out},k}}$ (where k identifies the layer). Figure 1.2 illustrates a multi-layer perceptron (MLP), which is a fully connected *feedforward network*. In a feedforward network, the information is processed by a chain of L layers $y = \big(f^{[L-1]}_{\theta_{L-1}} \circ f^{[L-2]}_{\theta_{L-2}} \circ \cdots \circ f^{[0]}_{\theta_0}\big)(x)$; if on the contrary there are feedback connections—i.e., if the graph of the layers processing the information contains cycles—the network is called a *recurrent neural network* (RNN).

Layers can be arbitrary functions, which need to be (sub-)differentiable in order to train the network via gradient descent. In practice, it is common to use convex functions with few non-differentiable points, for which one can simply choose a subgradient. A prominent example is the activation function ReLU : $x \mapsto \max(0, x)$, which is both cheap to compute and found to perform well [145].

A prototypical layer applies an affine transformation on its input vector $s_k \in \mathbb{R}^{d_{\text{in},k}}$, followed by a non-linear activation function φ_k:

$$f^{[k]}_{\theta_k}(s_k) = \varphi_k\big(W_k\, s_k + b_k\big), \quad W_k(\theta_k) \in \mathbb{R}^{d_{\text{out},k} \times d_{\text{in},k}},\ b_k(\theta_k) \in \mathbb{R}^{d_{\text{out},k}}. \tag{1.5}$$

The matrix W_k determines the *weights* for each input-output combination and the vector b_k is an additive *bias*. Both W_k and b_k are parameterized by the layers' parameters θ_k (sometimes also called "weights"). In a fully connected (dense) layer, each entry of W_k is a separate parameter, e.g. $[\vec{W}_k^\top, b_k^\top] = \theta_k^\top$. Other layer types, for example convolutional layers (see section 1.3.4), do not connect every input with every output and/or share parameters across multiple weight entries.

The parameters θ of a network are usually optimized by some kind of minibatch stochastic gradient descent (SGD) [305] of a loss $\ell(y)$ placed on the output of the network, using reverse mode

automatic differentiation (AD) to compute the gradients w.r.t. the parameters. This application of reverse mode AD is also called backpropagation, and is very efficient compared to forward mode AD because the loss is a scalar. To compute the gradient w.r.t. the parameters θ_0 of the first layer of a feedforward neural network, one needs to evaluate the product of the Jacobian matrices of the loss and all layers,

$$\nabla_{\theta_0} \ell|_{f_\theta(x)} = \nabla_y \ell \, J^{[L-1]} \, J^{[L-2]} \dots J^{[0]}, \tag{1.6}$$

where $\nabla_y \ell = \nabla_y \ell|_{f_\theta(x)} \in \mathbb{R}^{1 \times d_{\text{out}}}$, $J^{[k]} = \nabla_{\theta_k} f_{\theta_k}^{[k]}(s_k)\big|_{\theta_k} \in \mathbb{R}^{d_{\text{out},k} \times d_{\theta_k}}$ and s_k are the layer inputs saved from a so-called forward pass that computes $f_\theta(x)$; reverse mode AD evaluates the right-hand side of eq. (1.6) from the left, yielding first the gradients $\nabla_{\theta_{L-1}} \ell|_{f_\theta(x)} = \nabla_y \ell \, J^{[L-1]}$, $\nabla_{\theta_{L-2}} \ell|_{f_\theta(x)} = \nabla_y \ell \, J^{[L-1]} \, J^{[L-2]}$, etc., and finally $\nabla_{\theta_0} \ell|_{f_\theta(x)}$. This approach is computationally much cheaper than forward mode AD, which would first evaluate $J^{[0]}$, then $J^{[1]} \, J^{[0]}$, etc., which are all matrices, until the final multiplication with the vector $\nabla_y \ell$. A downside of reverse mode AD is the requirement to save the intermediate layer inputs s_k during the forward pass, which can impose large memory requirements; this can be partially relaxed by only saving some layer inputs and recomputing the others as needed, but usually such extra costs are avoided if possible.

There exists a large variety of neural network architectures, including (among many others) RNNs with LSTMs or gated recurrent units (GRUs), CNNs, transformers, and implicit neural representations. In this thesis, we only use CNNs; a brief general introduction will follow in section 1.3.4. Before, we take a look at supervised learning with neural networks.

1.3.3 Supervised deep learning

The most classical supervised approach defines a per-sample loss function $L : Y \times Y \to \mathbb{R}$ and trains the network f_θ by minimizing the empirical risk over the training dataset $(x_i, y_i^*)_{i=0,1,\dots,N-1}$,

$$R_{\text{emp}}(\theta) = \sum_{i=0}^{N-1} L(f_\theta(x_i), y_i^*). \tag{1.7}$$

By training on many representative data points, the ultimate aim is to *generalize* to previously unseen data of the same kind. Here, "data of the same kind" means that it stems from the same data-generating distribution [148].

Assuming a deterministic dependence of y on x, one can formalize the ideal solution to the task as a hypothetical function $y^* = f^*(x)$ returning the ideal output $y \in Y$ for any input $x \in X$. We would like the network f_θ to approximate f^*. A perfectly generalizing network approximates f^* on its full domain X, despite the training being performed on a limited number of examples.

Obviously, one requirement is that the network has sufficient *expressivity* (or *capacity*) to approximate f^*. The existence of such networks is covered by universal approximation theorems [186, 185, 91], which prove the capability of neural network families to approximate arbitrary functions in a function space, e.g. continuous functions on compact sets. Most results consider either arbitrarily wide or arbitrarily deep networks.

In addition to a sufficiently expressive network architecture, a learning algorithm is needed to find suitable network parameters θ. The by far most common approach is to employ a variant of minibatch stochastic gradient descent, e.g. Adam [221], to minimize the empirical risk eq. (1.7) on the training data. Compared to standard gradient descent, which would optimize the empirical risk over the whole training dataset, minibatch stochastic gradient descent both makes the training feasible by reducing the data per gradient step to a small minibatch of samples and is found to have a regularizing effect [356, 148]. Network weights are usually initialized randomly in

order to let the neurons behave differently, thus "breaking the symmetry" that would be present with fixed-value initialization; differently from weights, biases may be initialized zero (cf. [148]). Several techniques can be employed to support the learning, including data augmentation [350], explicit regularization like weight decay (corresponding to an ℓ_2-penalty on the network weights), or dropout [363], which randomly replaces part of the activation values in some layers with zero at each training step in order to prevent overfitting.

The generalizing properties of a network depend on many factors, including the data, architecture and optimization technique. If the training dataset is not representative or too small, the relevant structures cannot be extracted from it. Choosing an appropriate architecture can direct the learning towards certain kinds of information processing, e.g. when using convolutional neural networks for visual features (see section 1.3.4). Optimization of a neural network refers to finding a "good" local optimum in the typically non-convex loss landscape, which requires a good balance of exploration and stability. The learning dynamics are of course an interplay of all of these factors.

A network is said to *overfit* if a small training loss (1.7) is achieved, but generalization to new data of the same kind fails. This can happen if the network simply memorized the training dataset without learning relevant structures for the actual task. To estimate the performance on new data, one usually splits off a validation set from the training set and evaluates the empirical risk on this unseen data at certain intervals during the training. If this validation loss increases over time while the training loss is decreasing, this is an indication for overfitting. A common form of early stopping thus consists in selecting the network parameters that achieved the minimum validation loss.

The no free lunch theorems [402] state, vaguely speaking, that without prior knowledge no learning algorithm can be better than random guessing when averaging over all possible tasks. However, learning is usually concerned with solving specific tasks, for which performance can obviously differ between methods. Moreover, it is still possible to develop rather general learning techniques that perform well on a number tasks which might be more relevant to us than other more hypothetical tasks (see also [148]).

1.3.4 Convolutional neural network architectures

The most popular architectures for image processing tasks are based on convolutions, and thus called convolutional neural networks (CNNs) [241]. Other applications of CNNs include time series, speech and natural language processing, however recurrent neural networks and transformers are more typical in these fields. Recently, transformers are also emerging in computer vision [219], constituting an alternative to classical CNN architectures.

CNN building blocks

Convolutional blocks are typically composed of few layers, with one of them being a learned convolution operation, followed or preceded by a non-linear activation function (cf. section 1.3.2) and a normalization (e.g. batch normalization [197] or group normalization [411]). They can also include down- or up-sampling operations. Depending on whether series, images or image volumes are processed, 1D, 2D or 3D convolutions are used. In addition to the 1, 2 or 3 signal dimensions, a channel dimension is used to represent multiple features. For example, 2D image data with red, green, and blue light components would have three channels, represented with shape $\mathbb{R}^{3,H,W}$. The activations of hidden layers typically have much more channels, allowing to represent and combine many features. In the following description we choose the 2D image setting for clarity, with 1D or 3D convolutions working analogously.

The convolution operation applies different convolution filters for each input-output channel combination and sums over the input channels:

$$\begin{aligned}
&\mathrm{conv2d}_{K,b} : \mathbb{R}^{C_{\mathrm{in}},H,W} \to \mathbb{R}^{C_{\mathrm{out}},H,W}, \\
&\mathrm{conv2d}_{K,b}(s)_{c_{\mathrm{out}},:,:} := \left(\sum_{c_{\mathrm{in}}=0}^{C_{\mathrm{in}}-1} K_{c_{\mathrm{out}},c_{\mathrm{in}},:,:} \star s_{c_{\mathrm{in}},:,:} \right) + b_{c_{\mathrm{out}}} \quad \forall\, c_{\mathrm{out}} = 0, \ldots, C_{\mathrm{out}} - 1,
\end{aligned}$$

where $K \in \mathbb{R}^{C_{\mathrm{out}},C_{\mathrm{in}},F,F}$ contains the filters of size $F \times F$ and $b \in \mathbb{R}^{C_{\mathrm{out}}}$ contains bias values. The way the channels are mixed can be seen as a matrix-vector multiplication, i.e., the "matrix" of filters K is multiplied with the "vector" of signals s, where convolution is applied instead of multiplication for each matrix-vector element pair. The filters K and bias values b form the learnable parameters. As $\mathrm{conv2d}_{K,b}$ is an affine transformation, it can be viewed as one specific way to parameterize W_k and b_k in eq. (1.5) by K and b. Compared to a fully connected layer, the number of parameters is reduced greatly due to the convolutions operating on $F \times F$ windows only. Usually, the convolution "$\star$" is actually implemented as a cross-correlation (which is equivalent to convolution with flipped filters),

$$\begin{aligned}
&\star : \mathbb{R}^{F,F} \times \mathbb{R}^{H,W} \to \mathbb{R}^{H,W}, \\
&(f \star g)_{j_0,j_1} = \sum_{f_0=0}^{F-1} \sum_{f_1=0}^{F-1} f_{f_0,f_1} g_{j_0+f_0,j_1+f_1} \quad \forall\, j_0 = 0, \ldots, H-1,\ j_1 = 0, \ldots, W-1,
\end{aligned}$$

where $g_{j_0+f_0,j_1+f_1}$ with invalid indices $j_0 + f_0 \notin \{0, \ldots, H-1\}$ or $j_1 + f_1 \notin \{0, \ldots, W-1\}$ evaluates to a padding value (e.g. zero).

Convolutional architectures most commonly include down-sampling and/or up-sampling operations. Down-sampling can be realized by *strided convolution*, which applies the convolutional kernels only at an equidistant locations—for example, a strided convolution with stride 2 skips every second location, reducing each signal dimension by a factor of two. Strided convolution is equivalent to standard convolution followed by sub-sampling. Another form of down-sampling are *pooling* layers, which applies a reduction to the elements inside small windows. The most common reduction type is taking the maximum value (max-pooling), which adds non-linearity to the network in contrast to taking the mean (average-pooling) and induces invariance to small shifts (as long as the maximum element stays within the pooling region) [148]. Usually, the size and stride of pooling windows are chosen to be equal (called local pooling), such that the signal is essentially divided in non-overlapping blocks; however overlapping windows have also been used [229]. Pooling operates on the signal dimensions only and is applied to each channel individually. While the computational complexity of pooling is more similar to sub-sampling by using a stride in the preceding convolution, one can also contrast it with using an additional strided convolution in place of the pooling. Of course an additional convolution introduces more learned parameters and has increased computational cost, but it allows to learn different, channel-mixing down-sampling patterns, and is found to perform competitively not only due to the increased number of parameters [361]. As both max-pooling and strided convolution effectively sub-sample the input, they potentially cause high-frequency artifacts in the gradients [308], or aliasing artifacts, which could be remedied by using a smoother pooling variant [177]. Nevertheless, max-pooling and strided convolution are used successfully and most commonly.

Similar to down-sampling, up-sampling can be realized via a convolution variant which is known under the names *transposed convolution*, fractionally strided convolution, up-convolution, or (inappropriately) deconvolution. Transposed convolution is equivalent to first inserting zero values between the original signal values, such that the original values appear on a grid with

some stride (i.e. the up-sampling factor per signal dimension), followed by a standard convolution operation. Other forms of up-sampling are given by various kinds of interpolation, e.g. nearest-neighbor or (bi-/tri-)linear interpolation. As interpolation contains no learned parameters, one might want to apply a standard learned convolution after it. The authors of [308] propose this approach of interpolation followed by convolution, and favor it over transposed convolution, which they observe to produce checkerboard artifacts in some cases.

Besides convolutions, non-linear activations, and down- and up-sampling, it is often helpful or even necessary to include normalization layers, such as batch normalization [197]. Batch normalization usually facilitates more stable and faster training, as well as better generalization [337, 271]. While in the original paper the authors suggested to insert batch normalization layers before activation functions, other orders are now also used successfully and sometimes lead to faster convergence [166]. As batch normalization relies on statistics computed across a minibatch, it is likely to cause problems when applied with very small minibatch sizes due to unstable estimates. In such cases, group normalization is an alternative that is independent of the minibatch size [411]. Two special cases of group normalization are commonly known as layer normalization (with one group containing all channels) and instance normalization (with single-element groups for all channels).

Dropout [363] can be employed to reduce overfitting when training CNNs. While both dropout and batch normalization have regularizing effects, dropout is more explicit in this regard, and can be adjusted via the dropout rate parameter. However, combining dropout with batch normalization is noticed to perform suboptimal in some cases (but not in others), which has lead to recommendations in the literature to either be aware of and to address potential problems arising from the combined use, or to use batch normalization only when uncertain [257, 134].

CNN architectures

Figure 1.3 shows three CNN architectures developed for different tasks. We first consider the task of supervisedly learned image recognition, in which the network extracts information from an image. The network processing usually involves several down-sampling steps, reducing the image resolution while increasing the numbers of channels to represent many features, before finally computing the output, e.g. via fully connected layers or average-pooling followed by an output activation like softmax. The first example shown in fig. 1.3, VGG-16 [354], is a standard network for classification on ImageNet [95] with 1000 classes at a size of $(224\,\mathrm{px})^2$, trained on more than a million images. It improved upon the AlexNet architecture [229], which previously won the ImageNet Large Scale Visual Recognition Challenge (ILSVRC-2012), mainly by increasing depth, and thus the number of non-linearities, while using smaller (3×3) kernels and thereby reducing the number of parameters. Several configurations of the VGG architecture were presented, ranging from 11 to 19 layers with 133 to 144 millions of parameters. To address the problem of vanishing or exploding gradients arising from the backpropagation through many layers, the ResNet [170] architecture was proposed, which, by introducing shortcut connections, allows for the training of much deeper networks. In a ResNet, many residual blocks (cf. fig. 1.4) are chained, each of which learns a deviation from a shortcut signal that skips over the convolutions. ResNet configurations with up to 152 layers were presented for ImageNet, which despite the increased depth have fewer parameters and lower computational complexity than VGG networks [170]. Other popular CNNs for image recognition tasks include GoogLeNet [374], Inception-ResNet-V2 [375] and Darknet-19 [327]. Image recognition networks can learn to extract quite general features in the first part of the network, which enables their re-use, e.g. by adapting and fine-tuning a network that was pretrained to perform classification on ImageNet for the use in a different image recognition task, or as a so-called *backbone* in an object detection system [44, 349]. Many

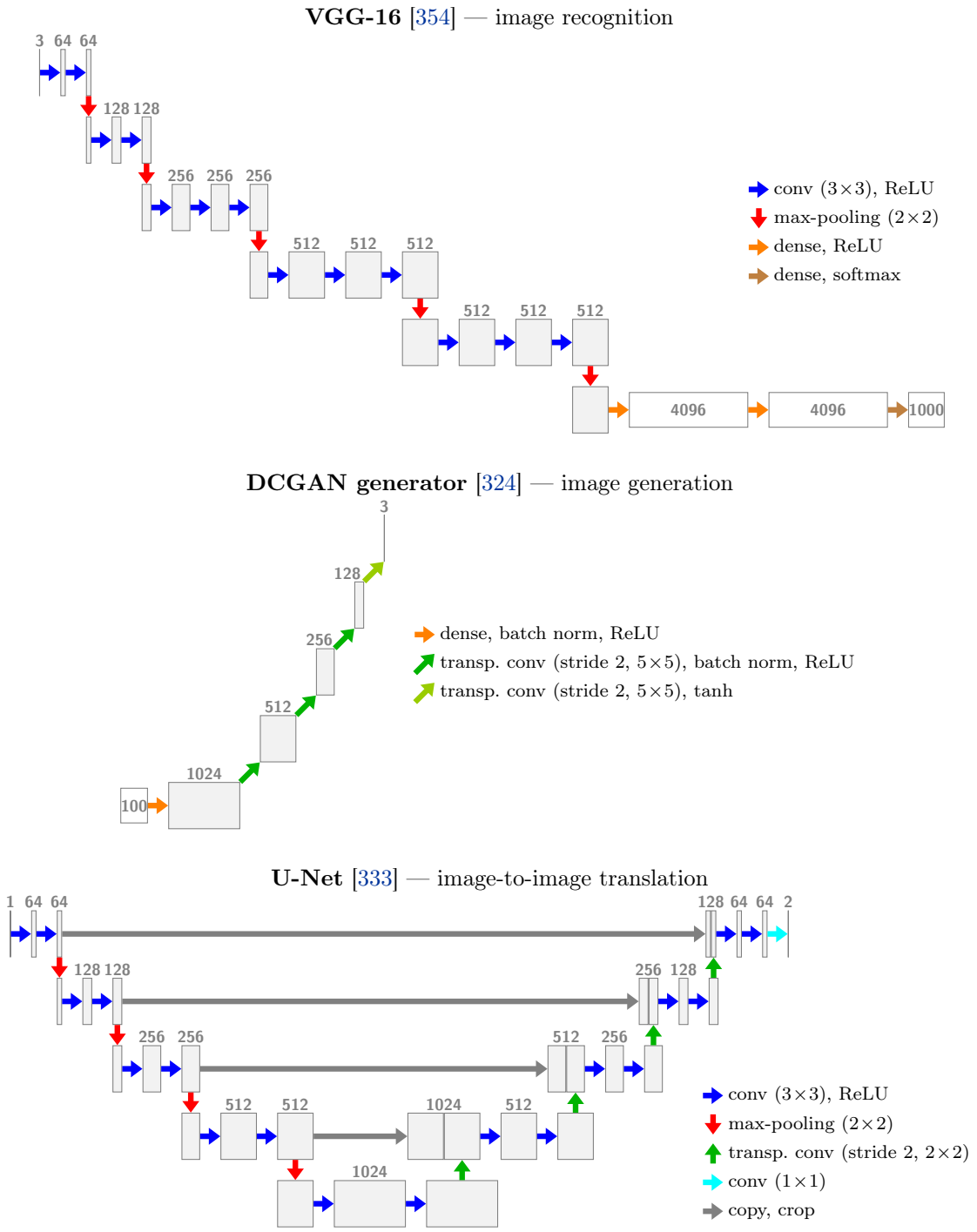

Figure 1.3: CNN architecture examples for different tasks. Gray boxes depict image-type activations, with the width indicating the number of channels (also specified above) and with the vertical position indicating the image resolution (top ≙ fine, bottom ≙ coarse). Non-image activations are visualized as white boxes containing the number of neurons inside.

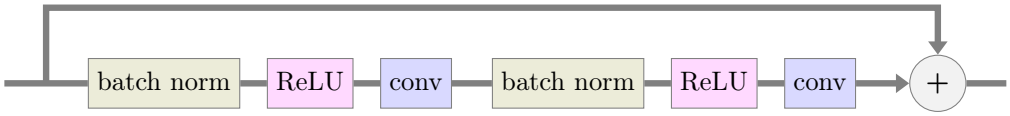

Figure 1.4: A residual block, bypassing information via an additive shortcut connection. Using residual blocks facilitates the training of very deep so-called residual networks (ResNets) [170]. The shown block has the structure proposed in [171]; other variants use a different order of operations, e.g., placing ReLU after summation, or apply a 1×1 convolution in the shortcut connection, optionally using stride a stride to perform down-sampling [170, 171].

pretrained networks (including but not limited to image recognition tasks) are publicly available from online repositories such as PyTorch Hub (`https://pytorch.org/hub/`) or TensorFlow Hub (`https://www.tensorflow.org/hub`).

Image generation, which is in some sense converse to image recognition, aims at generating images of a certain kind. The network learns to generate images from a distribution, being only provided with a training dataset of samples. Generative networks can be conditional on some user-specified variable, e.g. a class label or an input image, which can be fed into the network at arbitrary locations. Typically, the network transforms a simple random input while up-sampling several times. The second example in fig. 1.3 shows the generator part of DCGAN [324], which is a generative adversarial network (GAN) [149, 154]. GANs are a popular family of generative systems, made up of a generator and a discriminator network, where the latter has the role to predict whether an image is real or "faked" by the generator, thus encouraging the generator to produce images that are indistinguishable from real images. Other generative approaches include variational autoencoders (VAEs) [222, 322], energy-based models (EBM) [242] including diffusion models [357, 359, 87], autoregressive models [310] and normalizing flows [330, 313]; see [53, 111] for reviews and [383] for a book on deep generative approaches.

As a third task for CNNs we consider image-to-image translation (including image segmentation) that involves both aspects of feature extraction and of image generation. Architectures for such tasks often have an encoder-decoder structure, like the U-Net [333] (see fig. 1.3), which was proposed for medical image segmentation. The encoder extracts features and takes the role of capturing context, involving down-sampling of the signal, while the decoder synthesizes the segmentation map by processing, up-sampling and combination of the encoder signals at different scales, which are forwarded via shortcut connections. U-Net variants are popular in many medical imaging tasks, including segmentation [351], post-processing reconstruction [151] and synthesis [395]. Other popular architectures are based on variational autoencoders [222] or GANs, and sometimes also leverage standard recognition networks; for surveys, see [287] on segmentation methods, and [312] on image-to-image translation methods in general.

Network architecture definitions generally include some freedom of choice, regarded to as hyperparameters. For example, typical hyperparameters of CNNs are the number of scales and the number of channels for each of the convolutional layers. The choice of hyperparameters can have significant influence on a networks performance and may require some tuning for the specific task. Among others, it controls the expressivity of network parts and the risk of overfitting. Limited computational resources may also guide the hyperparameter choice. While it is common to perform manual hyperparameter searches, e.g. by training multiple supposedly interesting configurations, automated approaches for this task exist and are regarded to as neural architecture search (NAS; see [329] for a survey).

Chapter 2

Deep-learning-based reconstruction for computed tomography

As computed tomography is an important tool for both medical and industrial applications, it comes at no surprise that the success of deep learning has motived research on how to leverage its capabilities to improve CT reconstruction. E.g. for medical diagnosis, a high reconstruction quality is required, while the potentially harmful radiation dose should be as low as possible. Additional application-determined obstacles may hinder the reconstruction, such as a limited angular range from which projections can be measured or the presence of metal in medical scans. Artifacts in the reconstruction should be avoided as far as they are detrimental for the purpose of the scan. Hence, there is a great demand for better algorithms that solve challenging reconstruction tasks. In this chapter, we will give an overview of the great variety of deep learning approaches for X-ray CT reconstruction, which provides context for all our papers included in this cumulative dissertation, especially [252, 23, 253, 30]. Subsequently, we briefly discuss uncertainty estimation, which can help to assess the reliability of reconstructed features, and Bayesian experimental design for optimized acquisition, providing context for [14] and [31], respectively.

We consider the inverse problem of recovering the X-ray attenuation image $x \in \mathbb{R}^n$, $x \geq 0$, from measurements $y^\delta \in \mathbb{R}^m$, which are assumed to stem from the model

$$y^\delta = A\,x + \epsilon, \tag{2.1}$$

where $A \in \mathbb{R}^{m \times n}$ denotes the forward projection operator (mapping between the discrete spaces) and $\epsilon \in \mathbb{R}^m$ denotes the perturbations (e.g. noise) which can depend on x.

Most image reconstruction methods are in principle quite generally applicable, e.g. to different imaging modalities or image restoration tasks, for which the model is commonly formulated like in eq. (2.1), but differing in the forward operator A, the noise type and the image distributions. However, it is worth pointing out that tomographic imaging involves an operator A that transforms the image globally, while most natural image restoration tasks, such as deblurring or inpainting, deal with local transformations. Additional practical differences between applications lie in the data availability, especially for training supervised models, as well as the accurate knowledge of the operator A and the model for perturbations ϵ. Both data availability and operator knowledge are excellent for X-ray CT, especially compared to less popular imaging modalities. While vendor-specific acquisition models used to hinder the direct utilization of real medical projection data, progress is being made in this direction, and reconstruction from public real medical projection data is becoming possible (e.g. [387] reconstructing from the LDCT-and-Projection-data [291]). However, obtaining ground truth images, especially paired ones corresponding directly to degraded

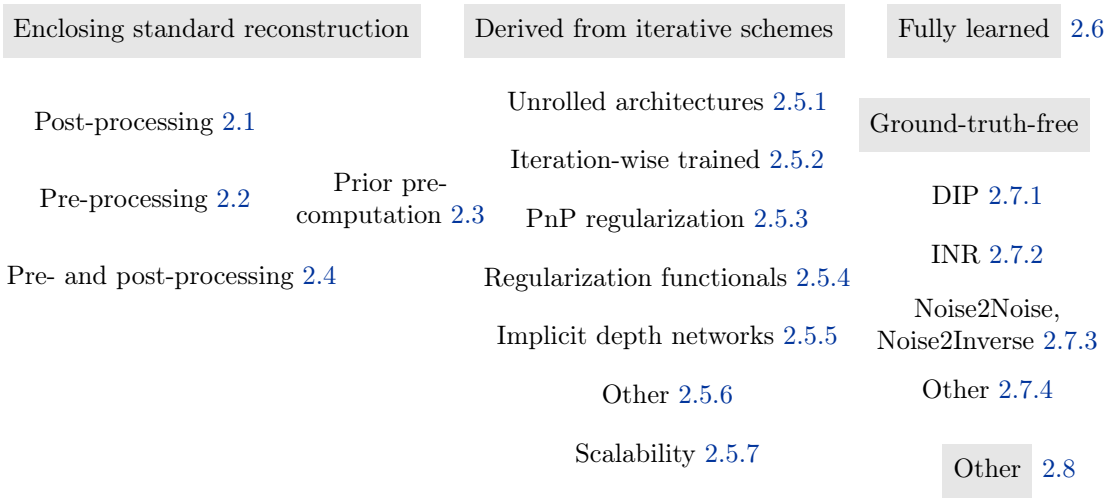

Figure 2.1: Outline of sections 2.1–2.8, which summarize deep learning approaches for CT

measurements or images, is an inherently difficult or even impossible task. Therefore, the creation of paired training datasets involves simulation of at least the degradation process (which is done in [291]), and it is also common to use fully simulated measurements with high-quality images serving as ground truth. Besides CT, magnetic resonance imaging (MRI) is another modality with very active deep learning research and outstanding public resources.

In our overview of reconstruction approaches, we solely focus on works evaluating their method on X-ray CT reconstruction tasks, only occasionally mentioning approaches from related fields. Despite the serious attempt to give an exhaustive overview, the list of discussed works in the present chapter is most likely incomplete due to the enormous publication activity in this field.

Sections 2.1–2.8 are organized by grouping the methods into categories of different learning strategies. A graphical outline is shown in fig. 2.1. The first groups of methods combine an existing reconstruction method with learned post-processing of the image (section 2.1), learned pre-processing of the measurements (section 2.2), learned prior pre-computation (section 2.3), or learned pre- and post-processing (section 2.4). The used training methodology includes standard supervised learning as well as generative modeling techniques, such as adversarial training and normalizing flows; note that ground-truth-free approaches falling in the categories of learned pre- and/or post-processing are covered later in section 2.7. Another group is called learned iterative reconstruction (section 2.5), which is derived from iterative reconstruction, e.g. by unrolling and modifying a finite number of iterations of a classical scheme into a network architecture, by applying a previously trained network in each iteration, or by iteration-wise training. Learned iterative reconstruction also includes methods that learn a regularization, either in the form of a learned "denoising" step in a plug-and-play approach, or explicitly as a learned regularization term to be included in the objective for iterative reconstruction. Some approaches follow the ambitious aim of learning the full inversion directly, which we refer to as fully learned reconstruction (section 2.6). Other approaches stand out by not requiring ground truth data (section 2.7), including e.g. the deep image prior (DIP), implicit neural representations (INRs), and Noise2Noise- and Noise2Inverse-based approaches. Methods that are not covered in the preceding sections are discussed in section 2.8. Estimation of uncertainty in the reconstructions is briefly discussed in section 2.9. Finally, we mention the current adoption into practice in section 2.10, and close with a review and outlook in section 2.11.

Note that the variable naming typically used in the machine learning and the inverse problems community differ in a potentially confusing way: In section 1.3, we used the naming style of machine learning that associates x with network inputs and y^* with desired outputs (labels); instead, when employing a network to predict a solution to the inverse problem eq. (2.1), it needs to map the input y^δ to an output x^*. We will henceforth use the variable naming of inverse problems. Throughout the chapter, when stating training loss functions, we will use the notation of expected values over dataset samples, e.g. $\mathbb{E}_{(y_i^\delta, x_i^*)}[\,\cdot\,]$, that should be understood as empirical estimates on a minibatch in each training step (cf. section 1.3.3).

2.1 Learned post-processing reconstruction

One way to introduce deep learning to image reconstruction consists in learned post-processing of preliminary reconstructions obtained by other methods. An analytical inversion formula is typically used to obtain the preliminary reconstructions, such as filtered back-projection (FBP) or Feldkamp-Davis-Kress (FDK) reconstruction, since they are efficient to evaluate. Subsequently, the task of the network amounts to the removal of artifacts, structured noise and in some cases blurring, which occur in classical reconstructions of e.g. sparse-view and low-dose CT. Thus, learning solely takes place in the image domain, for which deep learning is extensively studied in other applications as well, facilitating re-use of methodological advances to some degree.

Clearly, a non-injective preliminary reconstruction method could potentially discard important information that the network might not be able to recover. Including regularization in the first step can provide the network with input reconstructions of higher quality, which can be beneficial, depending on both how well the regularization suits the data and what can effectively be learned by the network from training data.

Learned post-processing methods can be roughly categorized by their training strategy. We will first consider directly trained methods, before turning to paired and unpaired adversarial training methods and to normalizing flows.

2.1.1 Directly trained post-processing reconstruction

Direct training uses a dataset of pairs $(\tilde{x}_i, x_i^*)_{i=0,1,\dots,N-1}$ of preliminary reconstructions $\tilde{x}_i$ and ground truth images x_i^* by minimizing the empirical risk

$$R_{\text{emp}}(\theta) = \mathbb{E}_{(\tilde{x}_i, x_i^*)}\big[L(f_\theta(\tilde{x}_i), x_i^*)\big]. \tag{2.2}$$

Typically, the loss L is implemented using the ℓ_2-distance (mean squared error, MSE), the ℓ_1-distance (mean absolute error, MAE), the structural similarity index measure (SSIM), or a combination of these. Some approaches also include a term comparing $h(f_\theta(\tilde{x}_i))$ with $h(x_i^*)$, where h is feature extraction or segmentation function, e.g. implemented by a previously trained network.

Early works falling in the category of learned post-processing reconstruction include [72, 73, 74, 215, 176] for low-dose CT and [446, 161, 205, 444, 415] for sparse-view CT. Many approaches utilize residual learning via shortcut (or skip) connections in the network architecture; some even directly add the input $\tilde{x}_i$ to the network output such that the network path effectively learns to solve the potentially simpler task of predicting residuals $x_i^* - \tilde{x}_i$ instead of clean images x_i^* (see [161] for a persistent homology analysis comparing the residual and original manifold).

Variants of the U-Net [333], which originally was developed for medical segmentation, are popular architecture choices for post-processing reconstruction [205, 452, 300]. It has also been

proposed to employ a U-Net in frequency domain (which relates to a synthetic parallel-beam sinogram as discussed in section 1.2.1), combined with an image-domain U-Net [85]. A different architecture called mixed-scale dense (MS-D) networks is proposed in [317], in which all layers are densely connected with skip connections, i.e., each layer receives the outputs of all preceding layers as its input. All layers operate at the same image size; instead of down- and up-sampling operations known from encoder-decoder type networks, dilated convolutions are used to perform processing at multiple scales. MS-D networks use a single output channel in each layer, resulting in a comparatively small number of parameters. A framelet-based multi-scale architecture is proposed in [425], linking classical image processing and network-based image processing. Other recently used architectures utilize dilated residual network (DRN) [438] parts and attention modules, the building blocks of transformers (cf. [155]): In [379], a DRN is applied for CT post-processing; in [255], a two-stage network architecture is used, including a DRN, channel attention modules, and a self-calibration module between the network stages; in [279], fused attention modules are inserted in a DRN. These works also utilize a perceptual loss, either based on activations of a pretrained VGG-16 network [379, 279] or using the encoder part of a separately trained autoencoder [255], combined with classical ℓ_2, ℓ_1 and/or SSIM loss terms. Multi-head attention modules are used in [445] in a network that first processes low- and high-frequency image components individually, before merging both network paths. A graph convolutional network (GCN) [223] using an encoder-decoder structure is proposed in [75]. Both DRNs and GCNs are techniques used to enlarge the receptive field, which helps to process non-local features.

While the preliminary reconstruction is most often obtained by an analytical inversion formula, which is computationally cheap and also preserves a direct relationship to the measurement data such that the structure of artifacts might be easier to learn, other reconstruction methods can be used as well. An example is SARTConvNet [392] (inspired by FBPConvNet [205]), which uses SART reconstructions for limited angle CT in order to provide the network with higher quality preliminary reconstructions.

For temporally resolved (4D) cone-beam CT, it is proposed in [448] to not only input the sparse-view phase images to a post-processing CNN, but also a motion-blurred reconstruction from projections of all phases, which is regarded to as a prior image guiding the CNN.

2.1.2 Adversarially trained post-processing reconstruction

Using a training strategy involving adversarial losses is usually motivated by the aim to produce detail-rich images, overcoming a tendency towards over-smooth reconstructions observed with other methods, such as directly trained methods, especially if an ℓ_2 loss is used, or methods using manual priors like TV. Adversarially trained post-processing methods differ in the required training data. Approaches derived from a conditional generative adversarial network (GAN) setting [289, 198] require a paired dataset of input and ground truth images $(\tilde{x}_i, x_i^*)_{i=0,1,\dots,N-1}$, while other methods only need unpaired datasets $(\tilde{x}_i)_{i=0,1,\dots,N-1}$ and $(x_j^*)_{j=0,1,\dots,M-1}$. We first discuss adversarially trained post-processing approaches that use a conditional setting, before later turning to unpaired approaches.

While conventional GAN approaches [149] aim to generate new images from a distribution after learning from a dataset of example images, conditional GANs (cGANs) [289] can be used for image-to-image translation tasks [198]. In cGAN-based image-to-image processing, an input image, called the conditioning, is provided to the generator and optionally also to the discriminator, while the random input provided to conventional GAN generators may be omitted.

Post-processing reconstruction approaches based on cGANs use a supervised training strategy on paired data $(\tilde{x}_i, x_i^*)_{i=0,1,\dots,N-1}$, employing a two-fold loss consisting of both an adversarial discriminator-based part and a ground-truth-based reconstruction loss term. Arguably, this

procedure shares more similarity with directly trained post-processing than with GANs, since both approaches use a reconstruction loss term computed on supervised pairs $(\tilde{x}_i, x_i^*)$, and, in the absence of a random input, the generator acts as a deterministic function transforming a degraded (e.g. low-dose) input image to a reconstruction. While for this reason randomization is artificially reintroduced via test-time dropout in [198], the image-to-image tasks considered in [198] are different from CT post-processing, for which test-time dropout seems to be uncommon in this context. However, the low-dose CT input images do contain (structured) noise.

There are several GAN variants computing the adversarial losses in different ways. Traditionally, GANs use binary cross-entropy (BCE) on a sigmoid-activated discriminator output, i.e., $L_{\text{bce}}(b, c) = -c \log(b) - (1 - c) \log(1 - b)$, where b is the sigmoid-activated discriminator output and $c = 0$ or $c = 1$ encodes "fake" or "real", respectively. Alternatively, adversarial losses can use least squares (LSGANs [278]) or a loss derived from the Wasserstein distance (WGANs [17]). The reconstruction loss term, e.g. realized with ℓ_2-distance, ℓ_1-distance, SSIM and/or a perceptual loss, is added to the adversarial generator loss, using some constant weighting factor $\lambda > 0$. This results e.g. (shown here for a BCE-based GAN) in the following generator and discriminator losses for post-processing reconstruction:

$$L_G(\theta_G) = \mathbb{E}_{(\tilde{x}_i, x_i^*)} \big[L_{\text{bce}}(D_{\theta_D}(G_{\theta_G}(\tilde{x}_i)), 1) + \lambda L(G_{\theta_G}(\tilde{x}_i), x_i^*) \big], \tag{2.3}$$

$$L_D(\theta_D) = \mathbb{E}_{\tilde{x}_i} \big[L_{\text{bce}}(D_{\theta_D}(G_{\theta_G}(\tilde{x}_i)), 0) \big] + \mathbb{E}_{x_i^*} \big[L_{\text{bce}}(D_{\theta_D}(x_i^*), 1) \big]. \tag{2.4}$$

Here, the adversarial L_{bce} parts encourage the generator to produce realistic images, while the image-domain loss L guides it to match the ground truth, like in eq. (2.2). Generator and discriminator may be trained alternatingly or simultaneously.

Most adversarially trained CT post-processing approaches target the task of low-dose image denoising. In [403], a traditional-GAN-based approach mixed with ℓ_2 image-domain loss is introduced. The works [420, 428] propose WGAN-based approaches, where [420] applies a VGG-based perceptual reconstruction loss, whereas [428] proposes to replace the VGG-based loss by a combined ℓ_1 and SSIM loss in order to avoid potential content distortion, while also extending the method to use a 3D CNN. Another WGAN-based approach is presented in [27], employing a U-Net with custom multi-scale dilated blocks in the skip connections as a generator, and using a reconstruction loss that in addition to the ℓ_2-distance also penalizes differences of image gradients between reconstruction and ground truth. The MAP-NN proposed in [344] also employs a WGAN loss, combined with an ℓ_2 and a Sobel-filter-based edge reconstruction loss, while using a sequential network architecture containing five residual U-Net-like blocks with a single output channel, which gradually process the input image and thus allow to report multiple reconstructions with different "denoising" strengths; the results are compared with those from commercial iterative reconstruction algorithms in a reader study, reporting favourable or comparable performance at a shorter computation time. In [193], an LSGAN-based approach is developed, also using both image and image gradient domain reconstruction losses, but employing in each domain a separate special discriminator with U-Net architecture of which not only the decoder's but also the encoder's output is used in the adversarial loss in order to obtain local discriminative feedback for the generator, regularized by the CutMix technique. A different application is targeted in [230], namely the super-resolution of 3D CT images to from thick to thin slices on different body parts, using a GAN with enhanced discriminator conditioning including both the thick-slice image and meta information about the slice, combined with an ℓ_1 reconstruction loss.

We note that among the previously discussed works, discriminator conditioning is only used in [230], where its usefulness is demonstrated for the therein considered virtual thin slice super-resolution task on different body parts. In contrast, the other aforementioned adversarially trained post-processing methods for low-dose denoising [403, 420, 27, 193] do not pass the degraded

input to the discriminator, with [403] reporting an observed bias towards the discriminator's performance when using discriminator conditioning. This deviates from the typical cGAN setting, in which both generator and discriminator are provided with the same conditioning [289, 198].

In [231], a special network structure is used for the discriminator, comprising a shared encoder before branching into (i) a decoder trained via a reconstruction loss, (ii) a global discriminator classifying the entire image, and (iii) a per-pixel segmentation discriminator. The discriminator loss consists of the reconstruction loss, adversarial classification and segmentation losses, and consistency losses that enforce the classification and segmentation predictions to stay similar when passing the discriminator's decoder output instead of the standard input to the discriminator. The generator is trained using the corresponding adversarial losses as well as supervised image and image gradient losses.

A CycleGAN-based approach that uses paired training data, which enables the inclusion of synthetic consistency losses in addition to the non-paired adversarial and cycle consistency losses of standard CycleGAN [450], is proposed in [165] for post-processing of cone-beam reconstructions.

We now turn to training strategies that do not require paired training data, but only separate datasets of low-dose images $(\tilde{x}_i)_{i=0,1,\dots,N-1}$ and normal-dose images $(x_j^*)_{j=0,1,\dots,M-1}$. In order to enforce correspondence of the generated output with the respective degraded input image, unpaired approaches mostly resort to a reconstruction loss based on the input image, as paired ground truth data is unavailable. One such method is presented in [376], which combines a CycleGAN [450] with a reconstruction loss term that compares the generated reconstructions with BM3D-denoised [92] versions of the low-dose input images. The works [315] and [427] use an ℓ_2 reconstruction loss comparing the generator output directly with the low-dose input; [315] uses a GAN with Kullback-Leibler (KL) divergence, and [427] uses a WGAN and additionally employs a VGG-based perceptual loss comparing the features of generator outputs with those of *unpaired* normal-dose images, aiming to learn general high-level semantic features of normal-dose CT images. A CycleGAN-based approach for multiphase CT is proposed in [216], using training images from a low-dose and a normal-dose phase, which only loosely match each other (i.e. they may be considered in between paired and unpaired data); the training loss contains, in addition to the adversarial and cyclic losses, identity losses that encourage each of the two generators to keep its output close to its input when applied to images from its target distribution.

2.1.3 Normalizing flows for post-processing reconstruction

Normalizing flows (NFs) [330] constitute another generative modeling framework. Here, an invertible network f_θ defines a mapping between the data space (e.g. CT images) and a latent probability space that is equipped with a simple density $p_Z(\,\cdot\,)$ from which it is also easy to draw samples (e.g. a normal distribution). This model implies a density in data space $p_X(\,\cdot\,)$ that is given by the change-of-variables formula $p_X(x) = p_Z(f_\theta(x))\,|J_{f_\theta}(x)|$, where $|J_{f_\theta}(x)|$ denotes the Jacobian determinant of f_θ w.r.t. its input evaluated at x. The network f_θ is trained to maximize the density $p_X(x^*)$ for the data samples x^* of a training dataset. From the trained NF, new samples can be generated by drawing samples z from the latent distribution and applying the network inverse f_θ^{-1}. One benefit of NFs over GANs and VAEs is the ability to efficiently evaluate the likelihood $p_X(x)$ of new samples x according to the flow model. NFs require special architectures. First, the network needs to be invertible, e.g. implemented as a sequence of invertible building blocks. Each invertible building block should allow for efficient computation of the inverse operation (for sampling) as well as the Jacobian determinant (for training). At the same time, the network composed of these blocks should have sufficient expressivity. Several block types have been proposed, including various kinds of coupling blocks [102] and Lipschitz-constrained

i-ResNet blocks [41]; see [53] for an overview. Due to the invertibility, the latent space must have the same dimension as the data space (at least when numbers are quantized as with computers). To reduce the high computational and memory demands of maintaining the full dimension in all intermediate activations, [103] proposes a multi-scale architecture that successively factors out parts of the information via shortcuts to the latent output. Since the network blocks are invertible, another strategy to significantly reduce memory consumption while slightly increasing computational cost during training consists in recomputing the activations during the backward pass via the inverse operations, as proposed in [118].

Like other generative models, NFs can be conditional [400, 16]. In a conditional NF, the invertible network receives a conditioning input $\tilde{x}$ that may be used in all layers and is trained to maximize the conditional density $p_X(x^* \,|\, \tilde{x})$ on a paired dataset of condition inputs and ground truth images $(\tilde{x}_i, x_i^*)_{i=0,1,\dots,N-1}$. For all possible conditions $\tilde{x}$, the conditional network $f_\theta(x; \tilde{x})$ needs to be an invertible function between its input x and its output; however, the condition $\tilde{x}$ may be processed by non-invertible operations. Conditional NFs can be applied to estimate the posterior density of an inverse problem, which is the conditional density of reconstructions for a given observation (see the statistical approach in section 1.1.2).

In [97, 96], a conditional NF is used for low-dose CT reconstruction. The therein proposed method does not directly use y^δ, but instead utilizes FBP reconstructions for the conditioning input $\tilde{x}$, thus falling in the category of post-processing reconstruction. The network is trained to maximize the conditional density given the FBP $p_X(x^* \,|\, \tilde{x})$ via the change-of-variables formula

$$p_X(x^* \,|\, \tilde{x}) = p_Z(f_\theta(x^*; \tilde{x})) \left| J_{f_\theta(\cdot;\tilde{x})}(x^*) \right| ,$$

where $\left| J_{f_\theta(\cdot;\tilde{x})}(x^*) \right|$ denotes the Jacobian determinant of $f_\theta(x; \tilde{x})$ w.r.t. its input x evaluated at x^*. Using a paired dataset of FBPs and ground truth images $(\tilde{x}_i, x_i^*)_{i=0,1,\dots,N-1}$, the negative log-likelihood is minimized:

$$L_{\mathrm{NF}}(\theta) = \mathbb{E}_{(\tilde{x}_i, x_i^*)} \big[-\log p_X(x_i^* \,|\, \tilde{x}_i) \big] = \mathbb{E}_{(\tilde{x}_i, x_i^*)} \big[-\log p_Z(f_\theta(x_i^*; \tilde{x}_i)) - \log \left| J_{f_\theta(\cdot;\tilde{x}_i)}(x_i^*) \right| \big].$$

After training, the likelihood of reconstruction candidates x can be evaluated, and reconstruction samples can be drawn via $z' \sim p_Z$, $x' = f_\theta^{-1}(z')$. Note that the conditional NF yields a full distribution of reconstructions, which may be used to estimate the model's uncertainty. From samples, statistics such as the pixel-wise mean and standard deviation can be easily computed. Practically, the pixel-wise mean over e.g. 100 or 1000 reconstruction samples is reported as the final reconstruction in [97, 96]. Other estimators could be used in principle, however initial experiments in [96] to minimize the negative log-likelihood according to the flow (MAP estimation), as well as minimizing a classical data discrepancy term regularized by the negative log-likelihood have lead to worse reconstruction quality in terms of peak signal-to-noise ratio (PSNR) and SSIM compared to the pixel-wise mean. The pixel-wise standard deviation may be interpreted as an indicator of how uncertain the model is about each of the pixel values.

For p_Z, [97] uses a normal distribution density (where $-\log p_Z(\,\cdot\,)$ equals $\frac{1}{2}\|\cdot\|^2$ up to a constant), and [96] experiments with choosing either a normal distribution or a radial distribution (where $-\log p_Z(\,\cdot\,)$ equals $\frac{1}{2}\|\cdot\|^2 + (n-1)\ln(\|\cdot\|)$ up to a constant, with n denoting the number of image pixels), for which, in contrast to the high-dimensional normal distribution, the typical set matches the high-density region near the mean (zero). It is observed that the radial distribution, while not leading to consistent improvements, requires less samples to obtain a good reconstruction, and the samples show much smaller deviations. We note that here the latent density p_Z is, like in [103, 16], not conditioned on $\tilde{x}$; however, this would be possible and has been used in [400] for different applications.

Based on NICE [102] and RealNVP [103], a multi-scale architecture using affine coupling blocks is used in [97], while employing i-RevNet down-sampling [201] and Haar down-sampling

[16]. The conditioning is integrated into the affine coupling blocks via the contained CNN parts that are not required to be invertible. A non-invertible conditioning CNN is used to process the FBP before feeding it to the conditional coupling blocks at each scale, trained jointly with the invertible flow network. In [96], both a multi-scale architecture and an iUNet architecture [118] are considered. Learned invertible down-sampling, which has been proposed in [118], has been used for both architectures. Affine coupling blocks are used for the multi-scale architecture and additive coupling blocks are used for the iUNet. Three different conditioning networks are tested for the multi-scale architecture, among which ResNet conditioning performs best. For the iUNet architecture, a U-Net conditioning is used, connected to the iUNet at the respective scales.

Conditional normalizing flows have also been studied for the application of nano-CT [272], using a simulated dataset, which features random object shifts for each scanning angle to model inexactness of the forward operator but does not contain noise. Here, the initial reconstructions are obtained by FBP, by RESESOP-Kaczmarz [51], or by a Dremel approach [108], where operator is inexactness is taken into account by the latter two methods. Both multi-scale and iUNet architectures are tested, as well as variants of both that are trained residually, which is found to be important. A direct U-Net post-processing is found to perform better than the flow-based approaches, but the flow-based approaches allow for further evaluation of the estimated posterior, e.g. for uncercainty estimation.

2.2 Learned pre-processing reconstruction

Complementary to learned post-processing reconstruction, in learned pre-processing reconstruction the network is applied *before* a classical reconstruction method. Thus, the network usually operates in the domain of CT measurements (sinograms). However, the network (or part of it) may operate in the image domain by using reconstruction and forward projection layers.

Many learned pre-processing methods target sinogram interpolation for sparse-view CT. Direct reconstruction from sparse-view measurements suffers from streaking artifacts. This motivates to interpolate the sinogram to a higher angular resolution prior to reconstruction. Besides classical approaches, such as directional sinogram interpolation [436] or self-similarity exploiting interpolation [217], deep learning methods have been developed for this task, predicting the missing projections at intermediate angles given the sparse-view projections. Metal artifact reduction (MAR) is another major application for learned pre-processing. Metal objects corrupt the measurements due to beam hardening and scattering, among others. The trace in the sinogram corresponding to the X-rays passing through a metal object is effectively destroyed, and other sinogram parts are slightly corrupted as well (see section 1.2; a survey of classical learning-free MAR techniques is found in [144]). Learned pre-processing methods for MAR thus need to solve a sinogram restoration task. They are commonly trained to predict a metal-free sinogram, which in particular includes the prediction of the sinogram values inside the metal trace. Some methods restrict the learned corrections to the metal trace, hence preserving the other sinogram parts, which usually do not suffer from significant corruptions due to the metal. Identifying the metal parts requires a segmentation (e.g. based on thresholding) in practice; during training on simulated data the true mask can be used directly. The metal parts can be inserted back afterwards to obtain the final reconstruction. Other applications for learned pre-processing include limited-view CT, interior CT and cycloidal CT.

In this section, we only consider approaches with supervised and optionally also adversarial training; few ground-truth-free pre-processing approaches are included in section 2.7.2.

2.2.1 Directly trained pre-processing reconstruction

A typical direct training for pre-processing minimizes the empirical risk

$$R_{\text{emp}}(\theta) = \mathbb{E}_{(y_i^\delta, y_i^*)}\big[L(f_\theta(y_i^\delta), y_i^*)\big] \tag{2.5}$$

for some loss function L on a paired dataset $(y_i^\delta, y_i^*)_{i=0,1,\dots,N-1}$ of degraded and reference measurements. The dimension of degraded measurements y_i^δ and reference measurements y_i^* may be different, e.g., for sparse-view sinogram interpolation y_i^δ contains projections from fewer angles than y_i^*. Both image-domain and/or sinogram-domain losses may be used, e.g., a reference reconstruction can be obtained from y_i^* using a classical method and compared to an image-domain network output. When training with data simulated from images and an image-domain loss should be included, it is of course preferable to alter eq. (2.5) such that the ground truth image x_i^* is directly passed to L in addition to or instead of y_i^*.

Sparse-view sinogram interpolation has been investigated in [244, 247, 261] using residual CNNs. U-Net architectures have been used for this task in [106, 38, 107, 245], with [107, 245] operating on sinogram patches instead of full sinograms. In [106], the method is also applied to sinogram inpainting in limited-view CT.

For the mild limited-view problem of short scans in a cone-beam geometry that use an angular range of π instead of the required 2π, [413] proposes a filtering approach: Learns a projection filtering layer (implemented in Fourier domain, equivalent to a single full-size convolution kernel) for the Feldkamp-Davis-Kress (FDK) reconstruction formula, trained end-to-end via an image-domain reconstruction loss. This approach presents a learned counterpart of classical filtering, e.g. based on Parker weights [316, 398].

MAR is targeted in [143] using a shallow 3-layer CNN receiving sinograms that have been initially processed by normalized metal artifact reduction (NMAR) [285], trained with a deep supervision loss comparing each layer's output to the metal-free sinogram in the ℓ_2 distance; at test time, the CNN reconstruction is averaged pixel-wise with the initial NMAR reconstruction, and additional averaging is applied around the metal object in the image domain. In [138], a 10-layer CNN is trained to directly predict a metal-free sinogram version from the metal-containing sinogram, using an ℓ_2 loss. A modified U-Net is used in [451] to predict the values inside metal traces from an input sinogram in which the traces have been erased based on a threshold segmentation; it is trained using a combined loss consisting of squared error terms comparing the predicted sinogram with the ground truth sinogram in terms of (i) their values in the metal trace area only, (ii) their gradients along the spatial detector dimension, aiming to avoid discontinuities at the border of the metal trace, and (iii) their sums along the spatial detector dimension in order to fulfill the Helgason–Ludwig consistency conditions (HLCC, cf. [250]). Even though learned pre-processing produces a refined sinogram, a network operating in the image domain may be utilized to implement this task. In [429], both image- and sinogram-domain networks are employed to perform the pre-processing: Sinogram values inside the metal traces are initially filled in by linear interpolation, and reconstructions from both the metal-containing and the linearly interpolated sinogram are passed to a U-Net that predicts a prior image, which is forward projected and subtracted from the linearly interpolated sinogram to form the input of a pyramid U-Net, which has access to the metal trace mask in all layers and predicts a residual sinogram whose values are added to the linearly interpolated values inside the metal trace; both networks are trained simultaneously using an ℓ_1 reconstruction loss on the prior U-Net output, an ℓ_1 loss on the predicted sinogram inside the metal trace only, an additional ℓ_1 loss on the full predicted sinogram for more efficient learning, and an FBP loss on the predicted sinogram in order to prevent secondary artifacts in the final FBP reconstruction. Another pre-processing approach for MAR, which only uses an image-domain network and in fact uses a post-processing

training procedure (with a loss derived from eq. (2.2) instead of eq. (2.5)), is taken in [442]: An image-domain CNN receives corresponding patches from three preliminary reconstructions, (i) with metal artifacts from the original sinograms, (ii) with beam-hardening correction, and (iii) from the sinogram with linearly interpolated values inside the metal trace, from which it is trained to produce a metal-free image patch; subsequently, a k-means-based tissue processing is applied to create a prior image with constant value for the water area, and its forward projection is used to replace the sinogram values inside the metal trace, followed by FBP and insertion of the metal parts.

For low-dose sinogram denoising, [273] uses a shallow feature extractor, blocks of convolutional layers that are densely connected by skip connections within each block, and global residual learning including an attention module before subtracting the noisy sinogram. In [283], a supervised sub-network trained on only a small dataset of low-dose/high-dose sinogram pairs is combined with an unsupervised sub-network utilizing a large dataset of low-dose sinograms.

In interior CT, truncated sinograms are acquired, which target a region of interest (ROI), but are insufficient for exactly reconstructing the ROI, because the external region contributes to the measured projections although not being fully scanned. A U-Net is trained in [218] to predict extended sinograms from such truncated sinograms before applying FBP reconstruction. Another approach, also applied to interior CT as well as to limited-view CT, is proposed in [190, 192] based on a constrained formulation of TV reconstruction (see e.g. [352]); in addition to the standard discrepancy contraint $\|A\,x - y^\delta\|^2 < \mathrm{tol}_{\mathrm{noise}}$ on the measured projections, the constraint $\|A_{\mathrm{virt}}\,x - A_{\mathrm{virt}}\,x_{\mathrm{prior}}\|^2 < \mathrm{tol}_{\mathrm{prior}}$ is placed on the virtual projections that are not measured but are relevant for the reconstruction, using an image x_{prior} predicted by a U-Net. From a learned pre-processing perspective, the virtual projection constraint would be interpreted as a second discrepancy constraint with an individual tolerance. However, since it only applies to non-measured data, it can also be viewed as a regularization using pre-computed learned prior information; such approaches will be discussed in section 2.3, including the work [191], which is a continuation of [192]. For cycloidal CT, which yields incomplete cycloidal projections due to simultaneous rotation and translation, a mixed-scale dense (MS-D) network [317] (cf. section 2.1.1) is employed in [318] to enhance the interpolated cycloidal projections; training is performed using full projections that are aquired at few angles, interleaved with the cycloidal acquisition.

2.2.2 Adversarially trained pre-processing reconstruction

Like for post-processing, conditional GANs (cGANs) have been used as a sinogram pre-processing as well. A supervised loss comparing the generator output with ground truth is included in addition to the adversarial loss, analogous to eq. (2.4).

In [139], a modified U-Net is used as a cGAN-generator for sparse-view sinogram interpolation, trained with an adversarial loss and an ℓ_2 loss comparing the generated sinogram with the ground truth sinogram.

For limited-view CT, a generator with encoder-decoder CNN structure is used in [24], trained with an adversarial loss and an ℓ_2 loss comparing the completed part of the generated sinogram with the ground truth. Similarly, a modified U-Net is used in [260], but here an adversarial loss is combined with both sinogram- and image-domain ℓ_1 losses, comparing not only the generated sinogram with the ground truth sinogram but also the FBP reconstructions from each with each other, respectively. A different, indirect approach via image space is used in [12]: The incomplete sinogram is transformed by a 1D CNN to a latent vector, from which a generator directly predicts an intermediate reconstruction, trained using both an ℓ_2 reconstruction loss and a discriminator-based adversarial loss. At test time, the forward projection of the generated intermediate reconstruction is computed for the missing angles to complete sinogram, followed by

classical reconstruction.

A cGAN-based approach with a U-Net-based generator is used in [140] for MAR, completing the values inside metal traces, which are erased from the input sinogram, trained using an adversarial loss and an ℓ_2 loss in the sinogram domain.

While [410] mainly focuses on magnetic particle imaging (MPI), the authors also apply the proposed projection generation approach to sparse-view 3D CT. The generator receives projections from two adjacent angles as its input and predicts the projection at the intermediate angle. This can be used to double the number of projection views, and may be repeated to successively increase the number of views. Hence, it presents a form of sparse-view sinogram interpolation, but differs from most approaches by performing angle-wise prediction from two adjacent projections, which in the 3D setting is a 2D processing task, whereas most other approaches are applied to 2D settings and process a full 2D sinogram, which consists of the 1D projections from all angles, at once. The projection generator in [410] is trained with an adversarial loss combined with ℓ_2 and SSIM losses.

2.3 Learned prior pre-computation

Learning may also be used to compute prior information (typically a prior image x_{prior}) for subsequent use in a classical (typically iterative) reconstruction method. This approach shares similarity with learned pre-processing (section 2.2) in the sense that the network is applied as a first step before the classical reconstruction. However, the network output is here used to form a regularization functional. Some similarity also exists with Plug-and-Play regularization (section 2.5.3) and learned regularization functionals (section 2.5.4), which we will cover later, and which differ from learned prior pre-computation in that they evaluate a network on each of the iterates during an iterative scheme. With learned prior pre-computation, instead, the regularization functional only uses the output of a network that is evaluated before starting the iterative scheme.

In [226], a simple Tikhonov regularization functional $\|x - x_{\text{prior}}\|^2$ is used where the image x_{prior} is computed by learned post-processing reconstruction; experiments are shown for 3D low-dose CT using a patch-wise operating network. By prescribing the regularization term form, the classical regularization theory is inherited directly. The field-of-view extension approach in [190, 192], which we already described in the last paragraph of section 2.2.1, also classifies as a learned prior pre-computation approach with the regularizing constraint $\|A_{\text{virt}}\, x - A_{\text{virt}}\, x_{\text{prior}}\|^2 < \text{tol}_{\text{prior}}$ on the virtual projections, where x_{prior} is provided by a U-Net. Continued studies in this direction are presented in [191], exploring different networks for the prior image generation (FBPConvNet, Pix2pixGAN), as well as different ways of integrating the prior information in a minimization objective, either in the projection domain or directly in the image domain.

In [407], a so-called DRONE architecture is proposed for sparse-view CT, utilizing multiple sub-networks, which ultimately yield an interpolated prior sinogram y_{prior} and a prior image x_{prior}. These are then used in minimization objective terms $\|A_{\text{full}}\, x - y_{\text{prior}}\|^2$ and $\text{TV}(x - x_{\text{prior}})$, respectively, where A_{full} is the full (non-sparse) forward projection model and TV denotes a total-variation functional.

2.4 Learned pre- and post-processing reconstruction

The previous sections discussed methods that apply either a post-processing network after obtaining an initial reconstruction or a pre-processing network on projection data before reconstruction.

When denoting the classical reconstruction as an operator $A^\dagger : \mathbb{R}^m \to \mathbb{R}^n$, post-processing reconstruction reads $f_\theta \circ A^\dagger$ and pre-processing reconstruction reads $A^\dagger \circ f_\theta$. Now, we turn to learned methods comprising both pre- and post-processing, i.e., which at reconstruction time perform three sequential steps: (i) learned pre-processing, (ii) classical reconstruction, and (iii) learned post-processing. This typically can be written as $f_\theta = f_{\text{post},\theta} \circ A^\dagger \circ f_{\text{pre},\theta}$. Some pre- and post-processing methods directly forward a preliminary non-learned reconstruction $\tilde{x}$ as an additional input to the post-processing network part, i.e. $f_\theta(y^\delta, \tilde{x}) = [f_{\text{post},\theta}(\,\cdot\,, \tilde{x}) \circ A^\dagger \circ f_{\text{pre},\theta}](y^\delta)$. The most common choice for $A^\dagger$ is an FBP operator. In some cases a simple back-projection is used in place of $A^\dagger$, leaving more of the filtering task to the network parts (cf. the analytical inversion discussed in section 1.2.1).

We note that other methods exist that interleave learned processing steps with classical reconstruction steps in a more intertwined manner, e.g. in learned iterative reconstruction (section 2.5) or other approaches (section 2.8).

While the three evaluation steps of learned pre- and post-processing reconstruction are sequential, many of these approaches simultaneously train the pre- and post-processing parts end-to-end in a single network f_θ performing all steps, e.g. using a loss of the form

$$R_{\text{emp}}(\theta) = \mathbb{E}_{(y_i^\delta, x_i^*)} \big[L(f_\theta(y_i^\delta), x_i^*) \big].$$

Adding the option to use a preliminary reconstruction $\tilde{x}$ and a projection-domain loss comparing with ground truth projection data y^* (weighted by a constant factor $\lambda > 0$), we obtain the more general loss

$$\mathbb{E}_{(y_i^\delta, y_i^*, \tilde{x}_i, x_i^*)} \big[L(f_\theta(y_i^\delta, \tilde{x}_i), x_i^*) + \lambda L_{\text{proj}}(f_{\text{pre},\theta}(y_i^\delta), y_i^*) \big].$$

Some approaches utilize a separate pretraining of the projection-domain network.

Multiple works target sparse-view reconstruction with a pre-processing network that refines interpolated sparse-view projection data, combined with an image-domain post-processing network. In [430], the pre-processing part $f_{\text{pre},\theta}$ applies bicubic interpolation to the sparse-view projections followed by a residual U-Net, and FBPs from both the output of $f_{\text{pre},\theta}$ and the sparse-view projections form the input of an image-domain residual U-Net; first, the projection-domain network is pretrained separately using projection ground truth data, and subsequently the complete network f_θ is trained end-to-end with alternating optimization steps for $f_{\text{pre},\theta}$ and $f_{\text{post},\theta}$. A U-Net-like architecture with Haar down-sampling as the first and Haar up-sampling as the last layer is used in [243] for both the projection- and the image-domain network, where $f_{\text{pre},\theta}$ processes linearly interpolated sparse-view projection data; the complete network f_θ is trained end-to-end. In [447], a slice-by-slice approach for the reconstruction from 3D sparse-view helical projection data is proposed: The 3D helical sparse-view data is first linearly interpolated and then mapped to slice-wise 2D fan-beam data using a U-Net that receives the section of the helical data that is relevant for the respective slice as multiple 2D channels, followed by FBP and a smaller U-Net for image-domain refinement; residual connections are used to wrap both networks, which in the projection domain is realized by adding the 2D fan-beam forward projections of a classical 3D reconstruction from the sparse-view helical data. The helical-to-fan-beam-projection network is pretrained separately before joint end-to-end training. Separate training of pre- and post-processing networks is used in [188], which facilitates the patch-wise (and thus memory-efficient) training of the two residual 3D U-Nets, which refine the linearly interpolated sparse-view projections and the reconstructed image, respectively. [25] proposes end-to-end training with two multi-level wavelet CNNs (MWCNNs), which are U-Net-like architectures with Haar down- and up-sampling between the different scales; training losses are placed both on the output of the first MWCNN refining linearly interpolated sparse-view projections and on the output reconstruction of the image-domain MWCNN.

Some approaches also target sparse-view CT, but do not interpolate the sparse-view sinogram with the pre-processing network. A closed-loop dual-domain training approach is proposed in [156], where a sinogram-to-image mapping consisting of a pre-processing U-Net, FBP and a post-processing U-Net is trained not only with a standard supervised loss, but also via a simultaneously learned image-to-sinogram mapping consisting of a U-Net and forward projection on which a sinogram-domain loss is placed. In [388], a Swin-based transformer is used in the sinogram domain, followed by FBP, and a residual image-domain network, which additionally receives a direct FBP reconstruction; supervised training losses are placed on the sinogram-domain transformer output, the FBP reconstruction from the sinogram-domain transformer output, and the final image-domain network output.

Metal artifact reduction is targeted in [262], where sinogram values inside the metal traces are first linearly interpolated, then a sinogram-enhancement network is used to replace the values inside metal traces, and FBPs from both the linearly interpolated sinogram and the network-enhanced sinogram are passed to an image-enhancement network. End-to-end training is performed using a three-fold loss, consisting of two losses that compare the sinogram- and image-domain network outputs with the respective ground truth and a Radon consistency loss on the sinogram-domain output that compares its FBP with the ground truth image in order to avoid the potential introduction of secondary artifacts due to the sinogram-enhancement.

For low-dose CT, [136] uses a network consisting of a shallow 1D CNN in the projection domain, FBP inversion and a residual encoder-decoder CNN [74] in the image domain. It is trained end-to-end using a combined ℓ_2 and VGG-based loss. In [204], an end-to-end trained network is formed by a projection-domain U-Net with long 1D convolution kernels operating in the detector pixel dimension, followed by simple back-projection and an image-domain U-Net. Here, the projection-domain 1D U-Net plays the role of a learned filtering step that replaces the classical ramp filter of FBP or FBP-based post-processing. The method is applied to different sparse-view settings.

2.5 Learned iterative reconstruction

A family of learned reconstruction methods is inspired by iterative reconstruction schemes, while modifying it to contain learned components. Several such approaches exist, differing in the iterative reconstruction algorithm that is adapted, the components that are learned, the training strategy and the provable convergence guarantees.

2.5.1 End-to-end trained (unrolled) iterative reconstruction

When fixing the number of iterations, they can simply be unrolled as network layers, yielding a network architecture that can be trained end-to-end. The network learns from a dataset of pairs $(y_i^\delta, x_i^*)_{i=0,1,\dots,N-1}$ of measurements y_i and ground truth images x_i^* by minimizing the empirical risk

$$R_{\text{emp}}(\theta) = \mathbb{E}_{(y_i^\delta, x_i^*)}\big[L(f_\theta(y_i^\delta), x_i^*)\big], \tag{2.6}$$

where the network f_θ may reuse the measurements y_i in each layer to compute (learned) iterative update steps.

Two such end-to-end trained iterative networks are partially learned gradient descent [3] and learned primal-dual [2], which are inspired by gradient descent and the primal dual hybrid gradient (PDHG) method [68], respectively. The original algorithms only serve as loose templates for

these learned algorithms, with several modifications being applied. Most importantly, gradient or proximal operators are replaced by learned convolution blocks, while optionally providing original gradient information as inputs to the convolution blocks. Additionally, memory variables are introduced to pass on information to subsequent iterations, and learned primal-dual furthermore generalizes the update steps to be freely learned, instead of enforcing the form they take in PDHG, and learns individual parameters for each iteration. This results in powerful architectures allowing for flexible processing, which in the case of learned primal-dual also takes places in the dual domain, i.e. the CT measurement (projection) domain. Learned primal-dual performed very well and also learned efficiently from a moderate number of data pairs in our evaluations [23, 253]. Similarly, JSR-Net [435] unrolls an algorithm based on the alternating direction method of multipliers (ADMM), approximating inverse and thresholding operators with CNNs operating in both image and projection domain. Another ADMM-based unrolled architecture is presented in [169], with each iteration being structured into four blocks performing sinogram restoration, image reconstruction, residual denoising using a CNN, and a multiplier update, where the denoising CNN and several other parameters of the modified scheme are learned.

The learned experts' assessment-based reconstruction network (LEARN) [71] unrolls gradient descent and uses 3Layer-CNNs for learned residual update terms that are added to the standard non-learned gradient descent updates, motivated by viewing the CNNs as implementations in the place of fields-of-experts regularization steps. Compared to partially learned gradient descent, which uses the standard gradients only as inputs to convolution blocks, this approach stays closer to the original algorithm by directly adding the gradient. Individual CNN parameters and step sizes are learned for each iteration of the LEARN architecture, increasing the flexibility. An extended variant of LEARN called LEARN++ [443] integrates additional networks in the projection domain, which perform inpainting operations on sparse-view data. Similar to LEARN, SCRED-Net [267] also learns a residual update term added to standard gradient descent updates, but in order to achieve scalability only considers a stochastic subset of projections at each step (similar to some ART-based methods, which however are usually non-stochastic, see section 1.2.2) and shares the parameters across all iterations. AirNet [70] unrolls fused analytical and iterative reconstruction (AIR) [131], which is based on the proximal gradient method for a regularized objective while replacing the adjoint $A^\top$ in the gradient steps w.r.t. the data discrepancy term with an analytical reconstruction operator $A^\dagger$, but instead of performing proximal steps for the regularizer term applies CNNs to perform residual updates, with the CNNs being densely connected in the sense that each CNN also receives the inputs of the CNNs at the previous iterations. An extension of AirNet for temporally resolved CT data from multiple respiratory phases has been proposed in [69]. The recently presented ADMM-SVNet [394] also introduces a learned component via a regularization: It unrolls an ADMM-based scheme to optimize a sparsity regularized objective, using two U-Nets in place of the sparse transformation and its adjoint. ADMM-SVNet also learns hyperparameters of ADMM. In [227], a U-Nets cascade is used that alternates network parts with data consistency layers that minimize the error in the projection domain regularized by the forward projected output of the preceding network part. Like the previously discussed schemes, LEARN, LEARN++, SCRED-Net, AirNet, ADMM-SVNet and U-Nets cascade are trained end-to-end via an objective of the form (2.6).

An unrolled CNN architecture called ΨDONet is proposed in [56], which resembles iterations of the iterative shrinkage-thresholding algorithm (ISTA) while representing the normal operator A^*A in the wavelet domain by decomposition into subbands, downsampling, convolution and upsampling operations, under the assumption that A^*A is a convolutional operator (which is the case e.g. for the considered limited-view parallel-beam CT). Based on this representation, the network learns an additive modification of the central part of the convolutional kernel of A^*A. The paper includes convergence results.

2.5.2 Iteration-wise trained iterative reconstruction

In the SUPER learning framework [259] supervisedly learned reconstruction steps are alternated with classical iterative reconstruction steps. Training is carried out sequentially for each learned reconstruction step, attempting to predict ground truth images from the output of the preceding iterative reconstruction step. Each classical iterative step performs a predefined number of iterations and is applied "offline" on the outputs of the preceding previously trained network, before training the subsequent network. Prior information can be included in the classical unsupervised steps.

Momentum-Net [82] generalizes a block proximal extrapolated gradient method using a majorizer (BPEG-M) [81, 80], which is in turn based on the block proximal gradient framework [418]. At each iteration, an individual image refining CNN is employed, which is used to ℓ^2-regularize a model-based objective for the subsequent reconstruction step. Each reconstruction step uses a majorization of the regularized model-based objective, which is then minimized by single proximal gradient step (per iteration). Additionally, momentum-based extrapolation is used for acceleration. Like in SUPER learning, training is carried out iteration-wise, aiming to predict ground truth images from the result of the previous iteration. The usage of individual CNN parameters for each iteration enables efficient processing, but complicates the study of convergence as the number of iterations tends to infinity. Still, the authors prove convergence under an assumption of asymptotical non-expansiveness for the sequence of subsequent CNN operator pairs, which is validated empirically, along with assumptions similar to those for BPEG-M.

2.5.3 Plug-and-Play regularization

Plug-and-Play (PnP) regularization is based on the concept of employing an existing algorithm performing a denoising-like operation in each iteration of an iterative scheme for regularizing purposes, without differentiating the denoising algorithm w.r.t. to its input. Thus the PnP regularization framework can use denoisers that do not allow for efficient back-propagation, and it is very flexible in combining different data discrepancy objectives and denoiser-induced "priors". Before turning to individual publications in the context of CT, we first outline the introduction of PnP frameworks in general.

A family of iterative schemes minimizes a regularized objective using proximal splitting, where each iteration is split into three steps: (i) a discrepancy-based update, (ii) a regularizer-based update, and (iii) a dual variable update step. The regularizer-based update only depends on the regularization term, which encodes prior information (e.g. as the negative log-likelihood of a prior distribution model, cf. section 1.1.2) and can often be interpreted as a denoising step. This modular structure provides the foundation for [386] to propose a general PnP framework, in which the regularizer-based update step is flexibly replaced with a denoising-like operation. It presents an algorithmic way to implicitly induce prior-like information, giving rise to the term PnP priors, even though no prior model is explicitly formulated. A different framework called regularization by denoising (RED) [332] also incorporates a denoising-like operation (denoted by $f(x)$) flexibly, but explicitly formulates the regularization term as $R(x) = \frac{1}{2}x^\top(x - f(x))$. In RED, the denoising engine $f(x)$ is assumed to fulfill homogeneity and passivity conditions, and the gradient of $R(x)$ is then approximated as $\nabla_x R(x) = x - f(x)$, which avoids to differentiate the denoiser f. Therefore the RED framework falls in the category of Plug-and-Play regularization, while still offering an (approximate) explicit regularization term. The convergence of RED with weaker assumptions on the denoiser is shown in [328] when using the proximal gradient algorithm. Another PnP-type framework [62] solves for a multi-agent consensus equilibrium (MACE), where the individual agents can each implement a data discrepancy or regularization component flexibly.

The consensus equilibrium is solved for using Mann iteration, for which convergence analysis is included in [62].

An ADMM-based PnP approach with a residual deep CNN denoiser is applied to low-dose CT in [424]. Another PnP-style approach based on projected gradient descent (PGD) is proposed in [157] for sparse-view CT, implementing the projector as a CNN, which takes the role of projecting iterates towards the set of desired solutions and is trained for this purpose beforehand; in order to deal with the CNN projector not necessarily being a true projection onto a convex set, the authors propose a relaxed PGD (RPGD) scheme that ensures convergence when the number of iterations tends to infinity. [346] follows a similar idea, but, varying from the standard PnP framework, it applies conjugate gradient (CG) steps, alternated with CNN steps (relaxed via a fixed hyperparameter) using a encoder-decoder architecture with a bottleneck. A PnP approach based on half-quadratic splitting (HQS) is proposed for low-dose CT in [132], using a pre-log statistical forward model and a three-layer CNN denoiser. Also based on HQS, [203] presents a PnP approach for sparse-view CT using a U-Net-based denoiser receiving a noise level map as additional input. A tuning-free ADMM-based PnP approach is applied to sparse-view CT in [396], using a residual U-Net denoiser receiving an additional noise level map input combined with actor-critic reinforcement learning for automated parameter selection determining whether to terminate, the denoising strength, and a penalty parameter encouraging termination if not improving substantially. In the family of RED approaches, [369] presents a block coordinate variant of RED (BC-RED), which per iteration updates one randomly selected image patch, using a patch-wise denoising network; it is applied to sparse-view CT. In [370], this approach is extended by parallelizing block updates (by allowing to operate on an old iterate) and additionally alternating over measurement blocks stochastically (i.e., stochastic gradients); experiments are performed on low-dose CT. Sufficient conditions for convergence are given in both [369] and [370]. A RED approach combining reconstruction and angle calibration is presented in [414] for sparse-view CT, using Nesterov accelerated gradient descent with a residual CNN denoiser. In [275], a partial Mann fixed-point iteration scheme solving for a MACE [62, 362] is applied to temporally resolved (4D) cone-beam CT reconstruction from sparse-view or limited-view data; one agent implements data discrepancy, complemented by three regularizing agents operating in the orthogonal xy, yz and xz planes, which utilize a denoising CNN receiving five slices from consecutive time points (called "2.5D"). An algorithm based on approximate message passing (AMP) is proposed in [321] for sparse-view CT: In order to apply standard AMP, which performs well for compressed sensing with i.i.d. random Gaussian operators, to a CT setting, the authors suggest to use a preconditioning and a Poisson noise model; a BM3D denoiser is used in [321], but the method allows for general denoisers, including learned ones.

2.5.4 Learned regularization functionals

Besides the previously discussed Plug-and-Play regularization approaches, one can also learn an explicit regularization functional $R_\theta(x)$. For example, [83] presents a learned regularization for sparse-view CT, although not using a neural network but a learned sparsifying transform matrix that maps image patches to sparse codes; once the sparsifying transform is learned, it is used in an ℓ_1-based regularization term of the reconstruction objective, which is then minimized by alternating image steps implemented via ADMM and sparse coding steps via hard-shrinkage. Deep learning offers the opportunity to learn more complex regularizing functionals, but complicates the analysis unless a special form or conditions are assumed for the functional. In this section, we only consider non-trivial learned regularization functionals in the sense that learned operations are applied on the argument x of the functional; other approaches that only utilize network-provided prior information as part of a classical regularization functional have been covered in section 2.2.

In [270], the authors propose to train an adversarial critic (discriminator) network, which is then used as a regularization functional in standard gradient descent. The critic is trained like for WGANs [17], distuingishing preliminary reconstructions from ground truth images in terms of the Wasserstein distance while (softly) enforcing a Lipschitz constant ≤ 1. Unpaired training data is sufficient, and the paper includes theoretical analysis as well as experiments on sparse view CT. This framework of adversarial regularizers is specialized in [297] to use an input-convex network [10], allowing the authors to prove convergence of the resulting regularization and the existence of a sub-gradient descent scheme for solving it. In terms of practical performance, the convexity constraint on the network architecture is reported to be beneficial in some experiments (limited-view CT, deblurring) but restrictive in others (sparse-view CT). Provably convergent learned regularization is also studied in [254], called network Tikhonov (NETT), under some assumptions including coercivity of the network; an encoder-decoder training scheme is employed, learning to extract either the artifacts or a zero image from a corrupted or a clean image, respectively, and the norm of the encoder output is then used as the regularization term in the reconstruction objective, which is solved with alternating gradient descent. While NETT originally was applied to photoacoustic tomography (PAT), sparse-view CT experiments are presented in [49], which proposes the non-stationary iterated network Tikhonov (iNETT), based on NETT and non-stationary iterated Tikhonov [206], avoiding tuning of the regularization parameter. To prove convergence of iNETT, the network is required to be uniformly input-convex, for which the authors introduce a U-Net-based architecture. A different approach called total deep variation (TDV) [225] learns a network-based regularization term by solving a discretized optimal control problem with fixed depth on training data, where the network parameters and the stopping time present the optimizable parameters; at reconstruction time, the learned regularizer term and a data discrepancy term form an objective that is minimized by gradient descent with Lipschitz backtracking. The authors point out that the regularizer may be reused across different problems, and indeed use a regularizer trained via the optimal control problem of denoising for sparse-view CT reconstruction. In [152], a shallow learned convex regularizer is proposed, targeting reliability and interpretability. A multi-gradient-step denoiser fulfilling convexity, existence and Lipschitz constraints is trained to solve a denoising problem, using learned convolutions and learnable linear spline activations, which corresponds to a regularization term that is the sum of convex ridges with learned profile functions that are splines of second degree. The constraints enable theoretical guarantees, and like in total deep variation and Plug-and-Play regularization, this denoiser can be trained without knowledge of the forward model, and indeed is constructed as a universal denoiser before applying it to the target application, which in this case is sparse-view CT.

2.5.5 Implicit depth models

Network models with implicit depth can be defined via a fixed-point equation; a prominent example are deep equilibrium models (DEQs) [26, 142]. A single network block is trained such that its repeated evaluation leads to a fixed-point solution forming the reconstruction by minimizing an implicit loss comparing the fixed point with the ground truth image. To compute the gradients of this loss during training, only the activations for the single network block must be stored, but typically a Jacobian-based linear system must be solved (approximately) in each step, which is computationally demanding. The network block can be interpreted as a layer of a network whose depth is defined by the number of evaluations, which can be chosen arbitrarily at test time to reach convergence. In the two subsequently summarized works, implicit depth models are applied to sparse-view CT reconstruction.

Feasibility-based fixed-point networks (F-FPNs) with implicit depth, alternating the application of a neural network consisting of residual CNN blocks with diagonally relaxed orthogonal

projections (DROP) [66], are proposed in [172], using Jacobian-free backpropagation (JFB) [128] to train the fixed-point network. Fixed-point networks generally allow for training with a memory consumption that is independent of the number of iterations, and JFB training additionally avoids the computational cost of solving a Jacobian-based linear system.

In [266], an online variant of deep equilibrium RED is proposed, approximating the gradient with respect to the data discrepancy using random minibatches of the measurement in each step (like in stochastic gradient descent) in order to make the complexity independent from the measurement dimension. The considered deep equilibrium RED is a gradient-descent-based DEQ that takes the form of regularization by denoising (RED) using steepest descent.

2.5.6 Other learned iterative reconstruction approaches

In [296], unrolled iterative reconstruction (section 2.5.1) and learned adversarial regularizers [270] (see section 2.5.4) are combined. An unrolled iterative network with an architecture like learned primal-dual [2] and a CNN regularizer are trained jointly in an alternating manner. Thereby the CNN regularizer acts as a critic (discriminator), which is trained via a Wasserstein distance loss to distuingish ground truth images from outputs of the unrolled iterative network, whereas the unrolled iterative network is trained to counteract the critic and to minimize a discrepancy loss, i.e. the training loss for the unrolled iterative network forms a variational regularization objective with the CNN regularizer. After training, the unrolled iterative network is evaluated to predict the reconstruction, optionally followed by a refining image-space gradient descent using the variational regularization objective with fixed regularizer network parameters.

Mainly motivated as a technique to stabilize learned reconstruction against perturbations including adversarial attacks, an analytic compressed iterative deep (ACID) scheme is proposed and studied in [408, 409], which combines an existing reconstruction network, a compressed sensing module that suppresses non-sparse components and data residual computation.

Based on ACID, the PRIOR approach is developed in [189] for temporally resolved (4D) cone-beam CT, which integrates a motion-blurred prior image obtained from all projections as an additional network input, like the post-processing approach [448] mentioned at the end of section 2.1.1.

2.5.7 Remarks on and approaches targeting scalability

In practice, learned iterative reconstruction methods tend to be computationally expensive. One reason is the involvement of the forward model, which in the application of CT requires the evaluation of forward- and back-projections in each iteration. For end-to-end trained networks, the number of iterations is a limiting factor, since during the training with many unrolled iterations the intermediate activations occupy a large amount of memory, and the traversal of all layers is time-consuming. This cost is avoided with other approaches by different means: SUPER and Momentum-Net apply greedy iteration-wise training, PnP and explicit regularization approaches train a single network in advance and implicit depth models use fixed-point network training (with some using Jacobian-free backpropagation). Hence these approaches are potentially easier to scale to higher resolutions and 3D CT images. A greedy iteration-wise approach specifically targeting scalability is presented in [404], which sequentially trains the CNNs that are inserted between the update steps of proximal gradient descent, with each CNN processing image patches instead of full images. While the greedy strategy may hinder the learning to find global optima to some extent, it greatly reduces memory requirements during training, both by training only one network part at a time, and, most importantly, by enabling the purely image-based patch-wise training for each iteration, which is decoupled from the forward model. A different scalable

approach that facilitates end-to-end training is developed in [167], called multi-scale iterative reconstruction. It performs iterations at different resolution levels, moving from a coarse to the desired resolution. Especially in 3D, lowering the resolution in the earlier iterations significantly reduces memory usage and computation time. This upsampling iterative network part is followed by a (residual) U-Net for additional processing, with both network parts being connected at different resolutions via skip connections. Compared to typical 2D approaches, both [404, 167] use relatively few iterations while employing larger network structures. In [404], a U-Net is used in each iteration, with the authors noting that using deep networks per iteration was important to compensate for the suboptimal local optima found by the greedy training strategy. In [167], prepending iterations with lower resolution is only possible or beneficial up to a certain level of coarseness, and the subsequent U-Net connects and extends the network, yielding a rather large multi-scale structure. Both approaches thus somehow combine the iterative network idea with more complex architectural components compared to the shallow networks usually found in end-to-end trained iterative 2D architectures. Besides computational and memory complexity of training, the reconstruction process with learned iterative schemes is typically more time-consuming than learned pre- and/or post-processing reconstruction due to the computation of multiple iterations involving both network and forward-/back-projections. This also applies to unrolled end-to-end trained methods, although those usually involve a relatively small number of iterations, while other learned iterative methods can have (theoretically) arbitrary numbers of iterations depending on a stopping criterion, such as implicit depth models (section 2.5.5), Plug-and-Play regularization (section 2.5.3) and learned regularization functional approaches (section 2.5.4).

2.6 Fully learned reconstruction

Some approaches follow a radical strategy of learning the entire inversion process directly from data while using only little, rather abstract knowledge of the forward model. Such full learning is clearly more ambitious and usually expected to require more training data (see also the benchmark results in [23] matching this expectation).

In [423] and [378], the network input is constructed as a stack of single-angle back-projection images for each angle of a sparse-view geometry, i.e. each input channel is a stripe image constructed from a 1D projection. While the single-angle back-projections are computed according to the forward model, their combination is up to the network, so a substantial part of the inversion is learned. A 20-layer CNN is trained in [423] to reconstruct from this input in a patch-wise manner (using 8×8 patches from 64×64 images). Instead, a WGAN-based training including an ℓ_1 reconstruction loss is performed in [378], using full 128×128 images; we note that from the description in [378] it not clear whether the simulated sparse-view projections contain noise.

A twelve-layer architecture called iCT-Net is proposed in [258], where only abstract model information is incorporated via the non-learned second-to-last layer that applies individual rotations to each channel. The authors compare segments of the architecture to steps of a filtered back-projection pipeline. The first five layers learn a signal correction, implemented via convolutions operating along the detector pixel dimension followed by dense mixing of channels corresponding to the angle dimension. The next four layers learn a filtering via various convolutions and mixing. Finally, there remains a learned counterpart to the backprojection, which is realized as a fully connected layer transforming the features to image space, followed by the non-learned rotation layer, and a final layer combining the rotated images to a reconstruction. The training is conducted in two stages, first using simulated data for a pretraining of individual network parts and also for a preliminary end-to-end training, before performing end-to-end training on

real-measured data. Experiments are shown for sparse-view, short-scan and interior CT, as well as combinations of these. Based on the iCT-Net architecture, the iCTU-Net is proposed in [36] for MAR, and also took part in our low-dose and sparse-view CT benchmark in [253].

The so-called iRadonMAP [168] rather closely resembles filtered back-projection, using a fully connected filtering layer operating in the detector pixel dimension (shared across all angles) and a learned sparse sinusodial back-projection layer, in which the locations of non-zero connections between sinogram pixels and image pixels are predefined, but the weights are learned. The output of the back-projection layer is subsequently refined by a residual CNN part. All training is performed end-to-end, first pretraining with (noise-free) simulated data based on ImageNet, before subsequently using clinical data for fine-tuning on sparse-view and low-dose tasks. We include a reimplementation of iRadonMAP in our comparison in [23], however without a pretraining, but instead trying to learn directly from our simulated benchmark datasets while studying the performance depending on the dataset size. The results confirm that iRadonMAP requires much more training data compared to other, not fully learned approaches (which is expected to be adressed at least to some degree by pretraining as proposed in [168]).

A hierarchical approach is proposed in [124, 125], decomposing the transformation from sinogram to image space into a sequence of sparse operations. The sinogram, which contains line integrals for each angle and detector pixel, is gradually transformed via intermediate representations containing integrals of shorter, partial line segments with a reduced angle resolution, until reaching a reconstructed image. These transforms are naturally sparse and are implemented using learned sparse layers with predefined locations of the non-zero connections (like for the sinusodial back-projection layer of iRadonMAP). Both the sinograms and the images have a size of 512×512 pixels, which allows the total dimension to stay the same throughout the transform sequence. The network additionally contains convolutional layers before and after the transforms. Training is first performed on pure noise image data and corresponding forward projections without adding measurement noise in order to learn the inversion process, including an individual pretraining of the transform layers as well as a preliminary end-to-end training. Subsequently, the network is trained on simulated low-dose clinical CT data.

All previously discussed fully learned approaches incorporate some notion of back-projection or rotation in the architecture. In contrast, DUG-RECON [213] only uses standard CNN components. Its reconstruction pipeline consists of a sinogram-denoising U-Net, a reconstruction U-Net that transfers from sinogram to image domain, and a residual CNN for image super-resolution. The reconstruction U-Net is trained as the first part (G_1) of a double U-Net generator (DUG), where the second generator (G_2) transforms the image back to a sinogram, trained alternatingly with supervised losses on G_1 and G_2 individually, and on $G_2 \circ G_1$ w.r.t. the weights of G_1. The denoising and super-resolution networks are trained supervisedly, using an ℓ_2 loss for the denoising and a VGG-based perceptual loss for the super-resolution network, respectively. Experiments are shown for low-dose CT.

Another approach that only involves standard CNN architecture components is proposed in [212], using an encoder-decoder structure, but the learning of the reconstruction task is assisted by injecting down-scaled filtered back-projections as additional inputs of the final decoder blocks. This can be interpreted as blending fully learned with post-processing reconstruction. The network is called a low-resolution reconstruction aware convolutional encoder-decoder (LRR-CED) and has a U-Net-like architecture, of which two variants are studied, one using densely connected convolutional blocks and another more standard U-Net architecture. LRR-CED is applied to sparse-view CT. Both [213] and [212] are included in the publicly accessible thesis [214].

In [419], an architecture is proposed that consists of (i) a transformer-based sinogram-domain module, (ii) a fully learned residual-CNN-based module predicting errors in the filtered back-projection based on the sinogram information, and (iii) a final image-domain U-Net module. By

adding the predicted errors to the filtered back-projection before passing it to the final U-Net, the approach is similar to a learned post-processing, but here the error prediction is fully learned. Training is performed using several losses, including supervised ℓ_1 losses on each of the three modules, as well as additional losses on the sinogram transformer output, one of which compares the values of coinciding beam paths within the predicted fan-beam sinogram, and the other compares the second derivatives of the predicted sinogram with those of ground truth sinograms. The approach targets low-dose CT.

2.7 Ground-truth-free learned reconstruction

As obtaining ground truth data is usually expensive, risky (e.g. for the health of the patient) or in other ways infeasible, there is a great demand for reconstruction methods that can be learned without ground truth. Here, "without ground truth" also rules out non-paired collections of ground truth images as required e.g. by learned adversarial regularizers (cf. section 2.5.4). The training of ground-truth-free methods is unsupervised and typically involves a loss enforcing consistency of the network output with the measurements or degraded images while exploiting useful biases or effects induced by the network architecture and training strategy. In some ground-truth-free approaches, training is only performed at reconstruction time using just the measurements y^δ, while others utilize an external dataset of measurements or degraded images. Ground-truth-free approaches may also be called self-supervised, since the reconstruction is learned solely from degraded "unlabelled" input(s), but, different from typical self-supervised learning, most ground-truth-free approaches do not conduct a second supervised learning step using the self-supervisedly learned predictions [28].

One popular ground-truth-free framework is the deep image prior (DIP), which we will discuss in section 2.7.1 including several extensions. Ground-truth-free approaches using coordinate-based implicit neural representations, which deviate from the standard array-based image processing, will be covered subsequently in section 2.7.2. Finally, approaches based on Noise2Noise or Noise2Inverse are discussed in section 2.7.3, and other ground-truth-free approaches in section 2.7.4.

2.7.1 Deep image prior

A powerful unsupervised image reconstruction framework, named deep image prior (DIP), has been introduced half a decade ago [249]. It can be viewed from the classical inverse problems perspective: Given corrupted observations $y^\delta \approx A\,x$ we wish to reconstruct the true image x. Applications presented in the original paper [249] include denoising (A is identity), deblurring (A is convolution) and inpainting (A is restriction), focusing on natural images, but DIP is now also applied to various tomographic reconstruction tasks [147, 88, 146, 129, 23, 449, 104, 34, 233, 30, 224]. A main advantage of DIP lies in the fact that it does not require a training dataset, which would be difficult or impossible to obtain in some applications, but only the measurements y^δ.

The central idea of DIP is to reparameterize the image x as the output of a (usually convolutional) neural network $f_\theta(z)$. Thus, image pixel values are not optimized directly, but instead by learning the network parameters θ,

$$\theta^* \in \widetilde{\operatorname*{argmin}_{\theta \in \mathbb{R}^p}} D(A\,f_\theta(z), y^\delta),$$

reporting $x^* := f_{\theta^*}(z)$ as the reconstruction. The iterative optimization of this objective typically requires early stopping (denoted by "$\widetilde{\ }$") to avoid overfitting to the corruptions contained in y^δ. Traditionally, the network input z is chosen as a fixed noise image with i.i.d. pixel values, but

it can be any image from which the network is able to generate useful output images. The loss function can be chosen like in the classical formulation of inverse problems: For example, one can choose the negative logarithm of a likelihood function model $p(y^\delta \mid A\,x)$ for the discrepancy D, such as the mean squared error (MSE) for a Gaussian noise model, or a Poisson regression loss for a Poisson noise model (see eq. (9) in [253], used e.g. in [146, 23, 253]).

Learning via the network architecture instead of directly optimizing the image pixels is found to effectively "regularize" the solution [249]. While one might notice that minimizing $D(A\,f_\theta(z), y^\delta)$ is mathematically equivalent to the constrained minimization of $D(A\,x, y^\delta)$ s.t. $x \in \{f_\theta(z) \mid \theta \in \mathbb{R}^d\}$, this only loosely relates to the working mechanism of DIP: Overparameterized network architectures are commonly used, which are capable of outputting virtually any image including noise, and the optimization is neither able nor aiming to find a global minimizer of D, instead relying on early stopping and the so-called *inductive bias* of learning via the architecture, that leads to a faster learning of natural images than of noise. Although this empirically observed behaviour has been studied in the literature [77, 67, 175, 347], its theoretical understanding is not yet complete. Some formal results exist for special architectures: Deep decoder was proposed as an alternative underparameterized architecture for DIP without spatial convolutions and with non-learned upsampling operations, which is less prone to overfitting and is proven to be incapable of fitting white Gaussian noise in the single-layer case [174]. Another more theoretically motivated approach is called the analytic deep prior (ADP) [105], which uses an unrolled proximal gradient architecture. Existence, stability and convergence results have been proven for the ADP [19]. For other more commonly used CNN architectures, the regularizing effects largely remain an empirical finding.

The implicit regularization induced by DIP can be mixed with explicit regularization by including a regularization term $R : X \to \mathbb{R}$ in the loss, like for variational regularization:

$$\theta^* \in \widetilde{\operatorname*{argmin}_{\theta \in \mathbb{R}^p}} D(A\,f_\theta(z), y^\delta) + \alpha R(f_\theta(z)).$$

If the explicit regularization fits the targeted image class well and is weighted suitably, it plays the role of a useful additional prior that leads to reduced overfitting and can also improve the reconstruction quality. A common choice for R is total variation (TV). DIP with TV regularization (DIP+TV) has been used for compressed sensing and image restoration tasks [385, 265, 64] as well as for CT reconstruction in [23, 34, 100].

Plug-and-Play regularization for Deep Image Prior

Regularizers can also be constructed by employing an existing denoising method via the plug-and-play (PnP) priors [386] or the regularization by denoising (RED) [332] framework. Approaches combining DIP with RED and PnP have been developed, which are called DeepRED [280] and PnP-DIP [372]. Of course, the regularization (or denoiser) can also be learned from data [385, 121]. Most methods involving a more complex regularization or prior information use an alternating direction method of multipliers (ADMM) optimization scheme instead of standard gradient descent variants. The combination of DIP with learned/PnP/RED regularization has been applied to different reconstruction tasks, including the closely related modality of positron emission tomography (PET) [366]; for X-ray CT, a constrained DIP optimization with RED (cDIP-RED) has been applied for artifact removal [65], however, as a post-processing method it operates in image space only and does not involve the CT forward operator A.

Stein's unbiased risk estimator for Deep Image Prior

Employing the Stein's unbiased risk estimator (SURE) [364] as DIP's training loss presents another technique to prevent overfitting to measurement noise. While SURE itself is only applicable to denoising ($A = \mathrm{Id}$), it has been extended to inverse problems with a non-trivial forward operator A, known as generalized SURE (GSURE) [113]. If the forward operator is rank-deficient (e.g. in sparse-view CT), GSURE resorts to estimating the projected error in the range of $A^\top$, which only provides a poor estimate of the true error [284, 5]. Nevertheless, promising results utilizing GSURE have been presented for deblurring and super-resolution [1] while optimizing via ADMM, as well as for undersampled MRI [208]. Also for MRI, [5] proposes an ensembling method over different randomized sampling patterns and images to overcome the inaccuracy of the projected GSURE estimate. To the best of our knowledge, GSURE-based DIP reconstruction has not been applied to CT reconstruction yet.

As far as the aforementioned regularizing techniques lead to a stabilized DIP optimization with reduced overfitting to measurement noise, this partially mitigates the need for a suitable early stopping. However, it needs to be expected that overfitting cannot be prevented completely, and developing early stopping criteria for DIP is a relevant subject of ongoing research [391, 256, 207, 390] that complements the stabilization efforts.

Bayesian approaches for Deep Image Prior

Multiple approaches to cast the DIP to a Bayesian framework have been developed, aiming to approximate the posterior. Essential goals with Bayesian DIP frameworks include overfitting reduction and the ability to perform uncertainty estimation. A simple way consists in using Monte-Carlo dropout (MCD) [130, 238], where dropout is applied both during optimization and afterwards when evaluating the network. The mean of Monte-Carlo samples is then reported as the reconstruction. This MCD approach has been successfully applied to CT reconstruction in [100], combined with TV regularization. In the application of natural image restoration, another Bayesian DIP variant based on stochastic gradient Langevin dynamics (SGLD) [397] has been proposed in [77], which injects noise in the gradients at each iteration and collects the iterates as samples after a burn-in phase. While [77] reports SGLD-based DIP to be effective against overfitting, it is evaluated as a baseline in [238] on medical image denoising tasks, where severe overfitting behaviour is observed with SGLD-based DIP and MCD-based DIP is proposed instead. In [382], both SGLD- and MCD-based DIP are reported to overfit with medical images at some point, and instead a mean-field variational inference (MFVI) approach with an automated prior selection strategy is presented, showing reduced overfitting. However, we note that the works [77, 238, 382] do not consider CT reconstruction but image restoration tasks. A MFVI-based DIP approach with posterior temperature optimization is applied to sparse-view CT reconstruction in [239]. Bayesian DIP approaches in general allow for uncertainty estimation, which we will briefly discuss in section 2.9, including our linearized DIP predictive posterior [14] that allows for calibrated uncertainty estimation given a readily optimized DIP network, i.e. without any changes to the deep image prior architecture or optimization.

Pretrained Deep Image Prior

DIP can also be combined with pretraining. For DIP-based PET denoising, pretraining is performed in [89] on a dataset of anatomical prior images from CT or MRI and corresponding noisy PET images, where MRI or CT images serve as the network input and noisy PET images serve as the target outputs. Unlike in typical supervised training, the more available noisy PET images are used instead of ground truth images. Then, the network with parameters initialized

from pretraining is fine-tuned to denoise a particular PET image by DIP-style optimization of the last few layers while fixing the parameters of the first network part. An improvement of reconstruction accuracy is observed with the pretraining initialization compared to classical random initialization. This approach is combined with a SURE loss in [90]. We study DIP pretraining for CT reconstruction in [30]. Here, the main focus is to speed up the reconstruction, which for standard non-pretrained DIP takes a practically prohibitive amount of time (e.g., several hours), since the network is optimized from scratch for each reconstruction. Pretraining is performed as the supervised post-processing (cf. section 2.1.1) of FBPs on a simple-to-generate synthetic dataset of random ellipsoid ground truth images and corresponding FBPs from simulated noisy measurements, serving as the network output and input respectively. Significant speed-up is observed for real-measured μCT datasets [57, 99] using 2D and 3D settings. A modified, more decoder-focused U-Net architecture is used for the 3D experiments. The performance evaluation is complemented by a singular value analysis to gain insights about the pretraining mechanism.

We note that all experiments in [30] use a linear output activation, yet a sigmoid output activation can improve non-pretrained DIP, significantly accelerating its convergence and slightly improving maximum PSNR on some datasets. However, we find the optimization with sigmoid output to require gradient clipping with a suitable maximum norm parameter, and we only recently became aware of this possibility. In more recent works [307, 33] (proposing different methods based on pretrained DIP discussed in the next section), we show results using a sigmoid output, including experiments on similar settings like in [30], thus clarifying to which extent the usage of a sigmoid output with suitably gradient-clipped optimization improves DIP without pretraining (also compared to pretrained DIP). We find the advantage of using a sigmoid output to be highly data-dependent: The experiment on the Lotus root [57] in [307] shows comparable results to those in [30], whereas for the experiment on the 2D Walnut [99] in [33] (with a slightly different geometry) the use of sigmoid boosts the non-pretrained DIP very notably, while a reduced adaptation flexibility of the pretrained DIP with sigmoid is observed, leading to a slightly reduced maximum PSNR compared to the non-pretrained DIP. We also refer the reader to Appendix C of [307], where we comment on different architecture choices and the need to fine-tune the gradient clipping maximum norm when using a final sigmoid activation.

Future work on pretrained DIP could experiment with adversarial pretraining, either cGAN-based and including a supervised reconstruction loss, or fully GAN-based with random network input. The cGAN-based variant, which stays closer to the currently used post-processing pretraining, has the benefit that it can exploit operator-specific knowledge about FBP artifact removal, while the pretraining of the fully GAN-based variant would be agnostic of the forward operator, so the same pretraining could be used for different geometries, but can only learn about the image distribution.

A DIP-related approach using a Noise2Noise-inspired [248, 406] network and including a pretraining performed solely on low-dose images is proposed in [405], but we decide to discuss this work in section 2.8, since it only utilizes the network via an image-domain regularization term in a scheme that jointly optimizes over the reconstructed image and the network parameters.

Pretrained Deep Image Prior on Subspaces

In addition to initializing the DIP network with the parameters from pretraining, one can restrict the subsequent unsupervised optimization to a subspace defined via the pretraining. This promises to reduce overfitting by limiting the parameter search space. We investigate two such approaches in [307, 33].

The SVD-DIP approach [307] applies an idea that in [368] was proposed in the application of few-shot segmentation: Singular value decompositions of the pretrained parameters for the

convolutional layers are computed, and only the singular values are then optimized for the target task, while the singular vectors are kept fixed. Applied to pretrained DIP reconstruction, we find this approach to stabilize the optimization, indeed reducing the overfitting behaviour.

The Subspace-DIP approach [33] constructs a low-dimensional affine linear subspace for all network parameters based on several checkpoints collected during the pretraining. To obtain the subspace basis, first the parameter vectors from the checkpoints are treated as columns of a matrix, of which the top k singular vectors are computed. The singular vector matrix is then sparsified along the parameter dimension by only keeping a number of rows with the largest ℓ_2 norm (leverage score [109]). This sparsified matrix forms the basis of the low-dimensional affine linear subspace, and the final pretrained parameters specifies its translation. By restricting the parameters to this subspace, the optimization becomes a low-dimensional problem, which makes it feasible to apply fast approximate second order optimization methods, such as the limited-memory Broyden-Fletcher-Goldfarb-Shanno algorithm (L-BFGS) [264] and approximate natural gradient descent (NGD) [9]. These methods converge in few iterations, and the overfitting is reduced successfully with the Subspace-DIP.

2.7.2 Implicit neural representations

Coordinate-based implicit neural representations (INR), e.g. [355], use an alternative way of representing images (and signals in general), where one does not form a matrix or tensor to specify the image values on a predefined discrete grid, but instead constructs a function, which maps (continuous) image coordinates to image values at the specified coordinates. In INRs, the function is a neural network (e.g. a multi-layer perceptron, short MLP), often preceded by a function encoding the coordinates in a higher-dimensional space.

In [433], an INR is used in a DIP-like manner for sparse-view cone-beam CT, with a so-called neural attention field (NAF) module that encodes equidistantly sampled points along X-ray paths with a learned hash encoder before applying an MLP to predict the attenuation value at the encoded coordinates. Projections are synthesized from these attenuation values via summation along the X-ray paths and compared to the measured projections in the loss for unsupervised DIP-style training. To obtain a final discrete 3D reconstruction, the network is evaluated for each point on a voxel grid.

A different approach using a DIP-style learned INR as an intermediate step is proposed in [432] for limited-view, sparse-view and super-resolution CT. The INR uses a Fourier feature encoding of the image coordinates and an MLP to predict the attenuation, from which sinograms with arbitrary resolution are generated via a projection layer. Unsupervised DIP-style training is applied via a loss comparing these sinograms with the measured ones. After the optimization, the final generated sinogram is used in a regularized objective solved by a classical reconstruction algorithm.

For sparse-view sinogram interpolation, [371] proposes an INR directly operating in the sinogram domain, where a Fourier feature encoding and MLP map sinogram coordinates to sinogram values. Different reconstruction methods are explored for the subsequent reconstruction using the fully sampled sinogram, including Plug-and-Play regularization and learned post-processing. In both [432] and [371], the INR plays the role of an unsupervised pre-processing.

A test-time adaptation framework utilizing INRs for two purposes is proposed in [358], constructed around a given trained black-box model. First, an INR with SIREN architecture [355] is used to adapt an FBP from the input sinogram in order to match the black-box model's training input distribution more closely. This is done by a stopping heuristics while fitting the INR to the FBP, which selects the point minimizing the ratio of the difference between the black-box model outputs evaluated for the current and a previous INR image over the difference

of these two INR images before the ratio increases for the first time. Then, the black-box model is applied on the resulting adapted input image to produce a prior reconstruction. Finally, a second INR is optimized DIP-style using a variational objective consisting of a discrepancy term and an ℓ_1 regularization term comparing the INR output to the prior reconstruction, after an initial phase of optimizing the INR to match the prior reconstruction (like proposed in our DIP work [23]).

2.7.3 Noise2Noise, Noise2Inverse and related approaches

Noise2Noise [248] is a training strategy, which uses pairs of noisy images for the network input and target, hence not requiring ground truth images. Multiple approaches based on Noise2Noise have been developed for low-dose CT. As one usually does not have access to multiple low-dose measurements (i.e., with the same image content but different noise realizations), single low-dose measurements are split synthetically in order to generate the training data. This way, the training only requires a collection of (independent) low-dose measurements $(y_i^\delta)_{i=0,1,\dots,N-1}$, whereby the training set may be separate from the target set. Such a self-supervised approach for pure denoising is known as Noise2Self [35], which has been adapted to inverse problems including CT in Noise2Inverse [179, 178], where the noise is assumed to be element-wise independent in the projection domain, but not in the image domain.

A Noise2Noise approach is followed in [431], in which each measurement y_i^δ is split into two synthetic versions with half of the original dose according to a shifted Poisson model, which guarantees independent noise realizations. The two synthetic versions are then used as input and target for the network training, which is applied either in the projection domain or in the image domain after applying FDK reconstruction, although in the latter case the zero-mean assumption of Noise2Noise is only approximately fulfilled. The network is then used for denoising of scans acquired using half the dose compared to that of the original training measurements y_i^δ.

In Noise2Inverse [179] for CT, each measurement y_i^δ is partitioned into subsets of uniformly distributed projection angles (i.e. each of the K subsets includes every K-th angle starting at a different offset). Training pairs of input and target images are generated by selecting one subset (or multiple ones), and computing FBPs once using only the projections from the complement of the subset (forming the input image) and once using only the projections from the subset (forming the target image). At test time, a reconstruction is computed as the pixel-wise average of the network output when evaluating it on each of the complement FBPs. The authors also experiment with swapping the roles of input and target images, but find that the higher-quality complement FBPs are more useful as input images than they are as target images. Noise2Inverse has been extended to 3D, dynamic and diffraction CT in [178]. In an independent, previously published work [406], the special case for $K = 2$ of the Noise2Inverse splitting and averaging evaluation is studied, i.e. selecting the even and odd projection angles, respectively, to compute two FBPs. Here, the so-called consensus loss includes an additional ℓ_2 term encouraging equality of the network outputs for each of the FBPs. As an alternative deployment strategy circumventing projection-domain manipulation at test time, the authors of [406] suggest one could train another network supervisedly to predict the first network's outputs from such low-dose images.

A self-supervised approach combining classical iterative optimization with DIP optimization via a Noise2Noise-consensus regularization is proposed in [405]. The sinogram is split into two sets of projections, similar to the even-odd splitting in [406] but slightly randomized, from which two FBPs, i.e. two images with different noise realizations, are computed. In the reconstruction objective, the data discrepancy term is combined with three ℓ_2 losses involving the network evaluated on each of the FBPs. Two of these losses compare the output of the network when applied on one of the FBPs with the respective other FBP (like in Noise2Inverse [179] and [406]),

and one loss compares the pixel-wise average of both network outputs with the image variable. Both the image variable and the network parameters are optimized jointly in an alternating manner, using patches for the network optimization. Here the training is self-contained like that of DIP, meaning that it does not use an external dataset of measurements, but only the target measurement y^δ.

2.7.4 Other ground-truth-free learned approaches

A self-supervised and self-contained approach is proposed for temporally resolved (4D) cone-beam CT in [274]. It uses a residual CNN with dense connections to post-process a series of sparse-view phase images, each of which comprises information specific to a temporal phase, but is subject to sparse-view artifacts. To this end, a training dataset is created from the 4D cone-beam projection data by splitting it into sparse-view pseudo-average subsets incorporating projections from different breathing phases, whose reconstructions show streaking artifacts similar to those of the sparse-view phase images. Reconstructions from the sparse-view pseudo-average subset are used as training inputs, and a high-quality reconstruction from all projections (but without temporal information) is used as the training target. After training, the network is applied on the sparse-view phase images to predict a series of refined phase images.

Another self-supervised, but not self-contained approach is presented in [220], training a projection-domain denoising network via a loss based on the Poisson unbiased risk estimator (PURE) for pre-log data or a weighted Stein's unbiased risk estimator (WSURE) for post-log data. The divergence term is approximated with a single Monte-Carlo sample. After applying the projection-domain denoiser as a pre-processing, the reconstruction is obtained via FBP.

2.8 Other approaches

A very early deep learning approach is the neural network FBP (NN-FBP) [319] for sparse-view CT. A small network architecture is composed of multiple FBPs with learned filters and biases, sigmoid activation, and learned pixel-wise affine linear combination of those FBP-based images, followed by a final sigmoid activation. It is extended to cone-beam CT with FDK reconstruction instead of FBP in [232], called NN-FDK, using the same network architecture like NN-FBP.

A Fourier-based reconstruction method, directly based on the Fourier slice theorem, is proposed in [101], where the regridding, i.e. the interpolation from polar to Cartesian coordinates in Fourier domain, is realized via a multi-scale network (cf. section 1.2.1).

For limited-view CT, [160] combines the learned pre-processing from [412] with learned unrolled iterative artifact removal based on [76] (which shares similarity with the unrolled LEARN approach [71] with the identity operator). Both parts are trained separately.

A shearlet-based approach for limited-view CT is proposed in [55], first performing a non-learned reconstruction with a shearlet-domain sparsity-regularization to split the shearlet space in a visible and an invisible part. The visible part is determined by the non-zero shearlet coefficients obtained from this sparsity-regularized reconstruction, while the other coefficients, forming the invisible part, are predicted by a U-Net. This can be viewed as an inpainting task in shearlet domain, after which the final reconstruction is obtained as the inverse shearlet transform of the combination of both parts. Note that the concept of only learning the parts that can not be measured is shared with the works [190, 192, 191] (see section 2.3) and also with deep null-space learning [343], although the methodologies differ substantially.

In [326], a learned approach is proposed for estimating object boundaries in limited-view CT, which, similarly to [55], first employs a non-learned reconstruction with sparsity-regularization in

the domain of dual-tree complex wavelet coefficients, before applying a combination of morphological operations and two U-Nets. The first network serves as a learned thresholding operation to identify the visible parts of the wavefront set, and the second network extends the visible to the invisible parts. Finally, while they do not form a complete reconstructed image, the estimated boundaries can be visualized as an overlay over the reconstruction, in which the boundaries may be obfuscated by stretching artifacts due to the missing projections.

The thesis [268] is concerned with deep learning for cardiac CT reconstruction, involving learned motion and metal artifact detection and removal. Networks are employed e.g. for motion vector prediction, segmentation and inpainting.

A dual-domain architecture is proposed in [426], that contains a main branch consisting of dual-domain blocks, as well as a controller branch and a fusion branch. The input features are extracted both from the sinogram and from a preliminary reconstruction, and there are skip connections bypassing each dual-domain block, so the network does not need to fully learn the reconstruction from sinogram domain (i.e. it could learn a post-processing), yet it can combine information from both domains via the dual-domain blocks. Each dual-domain block contains sinogram-to-image and image-to-sinogram CNN blocks, which use a final adaptive average pooling layer to adapt the output dimension of the block to that of the respective target domain, followed by an image-domain self-attention module. The main branch also includes sinogram-domain skip connections. With the image-domain controller and fusion branches, the main branch block outputs are further refined, involving a scalar adjustment parameter α that is supposed to provide a trade-off between detail preservation and artifact removal. To this end, the authors propose to first train the main branch separately using image-domain and sinogram-domain ℓ_1 losses while setting $\alpha = 0$, before then keeping the main branch fixed, setting $\alpha = 1$ and training the controller and fusion branches via a combined VGG-based perceptual and WGAN loss.

In [135], a cGAN-based approach is proposed, using an architecture with parallel image-domain and projection-domain streams. Both streams are connected at multiple points by interactive information flow (IIF) network parts, which involve forward projection and FBP for the domain transfer, before they are fused in a final block, which also receives the low-dose image and includes a multi-head attention block. Note that the architecture shares some similarity with unrolled iterative networks (section 2.5.1) due to the repeated connection via IIF parts. The network is trained with a direct reconstruction loss and with adversarial losses from patch-based discriminators, one operating on the output image while also receiving the low-dose image, and the other operating on the image gradients.

A so-called multi-domain integrative Swin transformer network (MIST-net) is proposed in [311] for sparse-view CT, first including a network structure operating in both sinogram and image domain, connected via FBP and projection layers as well as residual connections, before applying a vision transformer network using shifted windows (Swin) on the output image of the first network part.

A recently studied family of generative approaches are score-based diffusion models [359]. To apply them to inverse problems, a conditional sampling algorithm is designed in [360], which combines the learned unconditional score function and data consistency with the measurements y^δ. In [84], a manifold constrained gradient correction is proposed in order to keep the sampling path on the data manifold. Both works [360, 84] show results for sparse-view CT. The learning of these models "only" requires a dataset of clean images $(x_i^*)_{i=0,1,\dots,N-1}$ and is agnostic of the forward (and noise) model, which is only used in the conditional sampling algorithm at test time. This is in contrast to the post-processing approaches based on GANs or normalizing flows discussed in section 2.1.2 and section 2.1.3, most of which require paired training data $(\tilde{x}_i, x_i^*)_{i=0,1,\dots,N-1}$, and the others require unpaired data $(\tilde{x}_i)_{i=0,1,\dots,N-1}$ and $(x_j^*)_{j=0,1,\dots,M-1}$, in either case involving degraded images $\tilde{x}_i$, which are specific to the measurement process;

additionally, the post-processing approaches do not directly involve the measurements y^δ at test time, but only the initial reconstruction. In principle, diffusion models also allow for uncertainty estimation based on conditional samples, but this possibility has not been explored in [360, 84].

The ultimate goal of CT reconstruction is usually a down-stream task such as classification or segmentation, so it stands to reason that one may specifically target those tasks when training a learned reconstruction method. To this end, a joint training of a reconstruction network and a classification or segmention network is proposed in [4]. The loss function balances an accuracy loss for the target task with a reconstruction loss via a hyperparameter, which softly enforces a meaningful intermediate output of the reconstruction network. With a sensible balancing parameter, improved performance is empirically observed, not only compared to sequential training of both networks, but also compared to end-to-end training using only the accuracy loss, i.e. the reconstruction loss appears to regularize the training.

On a related note, some down-stream tasks do not necessarily require a reconstructed image as an intermediate step, and can be solved by directly interpreting the sinogram [94, 246, 133]; this is however outside the topic of this thesis focussing on CT reconstruction.

2.9 Uncertainty estimation

Reconstructions are not error-free in practice. For some part, this is inevitable, as the acquisition process yields noisy and often undersampled or in other ways degraded measurements, which are insufficient for precise image reconstruction. Additionally, errors might be introduced by inaccurate modeling and the reconstruction method. From a Bayesian perspective, the ultimate goal is the recovery of the posterior distribution, given an accurate likelihood and a suitable prior model (see the statistical approach in section 1.1.2). Hence the reconstruction naturally involves uncertainty, which one might want to assess for multiple reasons. An accurate uncertainty estimate allows to judge the reliability and may be used for example in subsequent automated image analysis tasks or to improve the reconstruction method [32].

Most reconstruction approaches only yield a reconstructed image, but no uncertainty information. Examples of deep learning approaches including uncertainty in the reconstruction model are normalizing flows (section 2.1.3), diffusion models (section 2.8), Bayesian approaches for DIP (section 2.7.1) and uncertainty-aware unrolled iterative reconstruction as proposed in [110]. In [14], we propose an uncertainty estimation framework for DIP that is applied post-hoc after completing the DIP optimization, providing a Gaussian model approximating the posterior by linearizing the network around the DIP solution and placing priors on the convolutional kernels. Note that g-prior feature normalization is proposed in this context in [13], which we did not yet use in our work [14] on the linearized DIP uncertainty framework, but which we use in [31] for scanning angle selection with a linearized DIP using a single variance prior on all network parameters. Our angle selection approach [31] can be seen as an example for utilizing uncertainty to improve the reconstruction.

Well-calibrated uncertainty quantification is a subject of ongoing research and typically bears computational challenges; for further reading, we refer to the recent review [32] on uncertainty quantification in medical image synthesis, including CT reconstruction.

2.10 Adoption into practice

Multiple deep-learning-based CT reconstruction solutions are commercially available and have been cleared by the U.S. Food and Drug Administration (FDA) for clinical practice [373]. These

include the Advanced Intelligent Clear-IQ Engine (AiCE) by Canon Medical Systems [52, 301], TrueFidelity by GE HealthCare [187, 314], and Precise Image by Philips [6, 153]. AiCE and Precise Image are post-processing reconstruction approaches. AiCE (Canon) receives an initial reconstruction of a so-called hybrid iterative reconstruction (hybrid-IR), which is faster than model-based iterative reconstruction (MBIR), but does not yield the same image quality [7]; Precise Image (Philips) post-processes FBPs. The TrueFidelity (GE) model directly receives raw data, and no further information on the used network architecture is found in the whitepaper [187]. Besides these vendor-specific approaches, there also exist vendor-agnostic noise reduction (i.e., image post-processing) solutions, including ClariCT.AI by ClariPi Inc. [302, 78] and PixelShine by AlgoMedica [8, 365].

The recent systematic review in [345] identified several studies evaluating the reconstruction of abdominal images with TrueFidelity (GE) and AiCE (Canon). Overall, the authors conclude that the quality is improved significantly, and dose reduction is possible, although care should be taken to select an appropriate level, especially to reconstruct small liver lesions [345]. Difficulties to faithfully reconstruct small low-contrast lesions at lower doses have also been reported for non-learned iterative reconstruction (IR) by multiple studies [292]. Among the studies reviewed in [345], the majority of readers favored the deep-learning-based reconstruction over both FBP and IR at the same dose level, and a dose reduction potential of more than 50% is reported on average.

2.11 Review and outlook

Deep learning for CT reconstruction is an active field, with a great variety of approaches being studied and with first solutions being deployed in practice [373]. While the results are promising, there is clearly a demand for further research on the sensible design, training and application of deep-learning-based CT reconstruction approaches [437, 298, 345].

Demanded properties of deep-learning-based CT reconstruction approaches include accuracy (especially regarding task-relevant image features), robustness (ideally guaranteed), sufficient generalization, interpretability, satisfiable training data requirements, and computational feasibility [389, 253, 437, 298]. Achieving all of these properties simultaneously is challenging. In some aspects there are natural limitations, such as that an ill-posed (or ill-conditioned) forward operator does not allow for simultaneously accurate and stable inversion (cf. [298]).

An often endorsed paradigm is to combine deep learning with classical techniques and theory, which allows to integrate both learned and hand-crafted information, and can aid interpretability [21, 389, 434, 298]. The recent survey [298] covers learned reconstruction methods with convergence guarantees, and concludes that the classical variational regularization framework and convex analysis largely form the basis for both the design and analysis of these methods. Yet, the authors of [298] also note that guarantuee-possessing methods may be outperformed empirically by other, heuristics-based methods, whose theoretical understanding is incomplete; for example, constraining a network to be input-convex enables guarantees but may limit its expressivity [297]. The learned methods with convergence guarantees reviewed in [298] include learned regularization functionals (section 2.5.4), deep equilibrium models (section 2.5.5), some post-processing networks ([425, 343]), PnP regularization (section 2.5.3), learned optimization of variational objectives, as well as Bayesian approaches with learned PnP or generative priors. Note that not all of these methods have been applied to CT reconstruction yet, but are suitable in principle. Regarding the robustness of deep-learning-based approaches, the often observed vulnerability to adversarial attacks, which identify small input perturbations that greatly disturb the output, has raised concerns [15]. On a converse note, adversarial examples use optimized worst-case perturbations,

which are not necessarily realistic, and may also be found for classical iterative TV reconstruction [137]. Strategies to achieve robustness against adversarial examples as well as the severity of adversarial vulnerability are studied in the literature e.g. for MRI [63, 295]. As mentioned in [298], such robustness investigations typically cover supervisedly learned end-to-end methods, but no unsupervised approaches (e.g. PnP regularization, learned regularization functionals, DIP, etc.), which could be subject of future research.

In practice, the acquisition of training data is challenging. The supervised training of a learned low-dose reconstruction method relies on a paired dataset of low-dose measurements (or initial reconstructions) and ground truth. While a large paired dataset of high quality offers great opportunities for data-driven knowledge extraction, its construction bears difficulties. Ground truth images (or sinograms) are especially difficult to obtain, since scanning with sufficient resolution and dosage is expensive or infeasible [389] and potentially harmful to the scanned subject. Consequently, reconstructions from reasonable high-dose acquisitions still contain imperfections, but are nevertheless commonly used as ground truth (e.g. in the training of the commercial AiCE [52] and TrueFidelity [187]). An alternative way to construct ground truth is to synthesize virtual images, possibly also using deep learning techniques [389], but naturally bears the risk of not being realistic and representative of the true image distribution. Besides these difficulties to obtain ground truth (which of course also apply to unsupervised methods requiring ground truth images), the creation of a paired dataset for supervised training involves an additional challenge by requiring corresponding low-dose (or otherwise degraded) measurements. If a pair of real high-dose and real low-dose measurements should be used, it must be ensured that the imaged content remains unchanged between the two measurements. Alternatively, low-dose measurements may be simulated from the high-dose measurements [291] or from the ground truth images in more or less sophisticated ways [389, 252], again bearing a risk of being unrealistic. If the ground truth is synthetic, simulation is in fact the only option to obtain low-dose measurements. Limitations of the dataset, such as imperfections in the ground truth or simulation of the measurements, or the under-representation of rare but important pathologies [434], may impair the learning. Some learned methods require a large training dataset to perform well [23].

Even though the commercial introduction of supervisedly trained reconstruction methods has demonstrated the possibility of constructing acceptable training data, the aforementioned challenges motivate the investigation of methods with low data requirements, especially for CT applications with scarce data availability. These include unsupervised methods, which do not need paired data, and in particular ground-truth-free methods (section 2.7). Intermediate strategies to deal with limited data availability consist in transfer learning [437] and training on simulated dataset followed by fine-tuning on real data [389]. Following a similar direction in the context of ground-truth-free methods, the pretrained DIP [30] and methods based thereon [307, 33] combine a pretraining on an easy-to-generate simulated paired dataset with unsupervised DIP-style fine-tuning.

To evaluate the use of deep-learning-based reconstruction and to identify dose reduction potential, the image quality must be assessed, ideally considering the down-stream task such as a medical diagnosis. In methodological research, standard image metrics such as PSNR or SSIM are commonly employed to compare image quality. While easy to quantify and reproducible, they might not be indicative of its the actual value for the down-stream task [202, 290]; thus, expert reader studies, although laborious and inherently subjective to some degree, are needed for image quality assessment, unless the down-stream task is fully automated and its correct results (labels) are known. Besides the reconstructed image itself, decision-making can potentially benefit from estimated reconstruction uncercainty [32], interpretability of how the learned method determines the reconstruction (i.e., understanding its internal "reasoning" if possible) [389, 434], and from external information [434].

Bibliography

[1] S. Abu-Hussein et al. "Image Restoration by Deep Projected GSURE". In: *2022 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2022, pp. 91–100. DOI: 10.1109/WACV51458.2022.00017.

[2] J. Adler and O. Öktem. "Learned Primal-Dual Reconstruction". In: *IEEE Transactions on Medical Imaging* 37.6 (2018), pp. 1322–1332. DOI: 10.1109/TMI.2018.2799231.

[3] J. Adler and O. Öktem. "Solving ill-posed inverse problems using iterative deep neural networks". In: *Inverse Problems* 33.12 (2017), p. 124007. DOI: 10.1088/1361-6420/aa9581.

[4] J. Adler et al. "Task adapted reconstruction for inverse problems". In: *Inverse Problems* 38.7 (2022), p. 075006. DOI: 10.1088/1361-6420/ac28ec.

[5] H. K. Aggarwal, A. Pramanik, and M. Jacob. "Ensure: Ensemble Stein's Unbiased Risk Estimator for Unsupervised Learning". In: *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2021, pp. 1160–1164. DOI: 10.1109/ICASSP39728.2021.9414513.

[6] *AI for significantly lower dose and improved image quality - Precise Image*. URL: https://www.philips.com/c-dam/b2bhc/master/resource-catalog/landing/precise-suite/incisive_precise_image.pdf (visited on 07/21/2023).

[7] M. Akagi et al. "Deep learning reconstruction improves image quality of abdominal ultra-high-resolution CT". In: *European Radiology* 29.11 (2019), pp. 6163–6171. ISSN: 1432-1084. DOI: 10.1007/s00330-019-06170-3.

[8] AlgoMedica. *What's that noise? How deep learning can elevate CT image quality*. URL: https://cdn.b12.io/client_media/MylCrHXR/b4795b8a-4e4f-11ed-aff9-0242ac110003-1015WHAT_IS_THAT_NOISE_101522-modified.pdf (visited on 07/19/2023).

[9] S.-i. Amari. "Natural Gradient Works Efficiently in Learning". In: *Neural Computation* 10.2 (1998), pp. 251–276. ISSN: 0899-7667. DOI: 10.1162/089976698300017746.

[10] B. Amos, L. Xu, and J. Z. Kolter. "Input Convex Neural Networks". In: *Proceedings of the 34th International Conference on Machine Learning*. Ed. by D. Precup and Y. W. Teh. Vol. 70. Proceedings of Machine Learning Research. PMLR, 2017, pp. 146–155. URL: https://proceedings.mlr.press/v70/amos17b.html.

[11] A. Andersen and A. Kak. "Simultaneous Algebraic Reconstruction Technique (SART): A superior implementation of the ART algorithm". In: *Ultrasonic Imaging* 6.1 (1984), pp. 81–94. ISSN: 0161-7346. DOI: https://doi.org/10.1016/0161-7346(84)90008-7.

[12] R. Anirudh et al. "Lose the Views: Limited Angle CT Reconstruction via Implicit Sinogram Completion". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 6343–6352. DOI: 10.1109/CVPR.2018.00664.

[13] J. Antoran et al. “Sampling-based inference for large linear models, with application to linearised Laplace”. In: *International Conference on Learning Representations*. 2023. URL: https://openreview.net/forum?id=aoDyX6vSqsd.

[14] J. Antoran et al. “Uncertainty Estimation for Computed Tomography with a Linearised Deep Image Prior”. In: *Transactions on Machine Learning Research* (2023). ISSN: 2835-8856. URL: https://openreview.net/forum?id=FWyabz82fH.

[15] V. Antun et al. “On instabilities of deep learning in image reconstruction and the potential costs of AI”. In: *Proceedings of the National Academy of Sciences* 117.48 (2020), pp. 30088–30095. DOI: 10.1073/pnas.1907377117.

[16] L. Ardizzone et al. *Guided Image Generation with Conditional Invertible Neural Networks*. 2019. DOI: 10.48550/ARXIV.1907.02392.

[17] M. Arjovsky, S. Chintala, and L. Bottou. “Wasserstein Generative Adversarial Networks”. In: *Proceedings of the 34th International Conference on Machine Learning*. Ed. by D. Precup and Y. W. Teh. Vol. 70. Proceedings of Machine Learning Research. PMLR, 2017, pp. 214–223. URL: https://proceedings.mlr.press/v70/arjovsky17a.html.

[18] S. G. Armato III et al. “The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): A Completed Reference Database of Lung Nodules on CT Scans”. In: *Med. Phys.* 38.2 (2011), pp. 915–931. ISSN: 0094-2405. DOI: 10.1118/1.3528204.

[19] C. Arndt. “Regularization theory of the analytic deep prior approach”. In: *Inverse Problems* 38.11 (2022), p. 115005. DOI: 10.1088/1361-6420/ac9011.

[20] C. Arndt et al. “In Focus - hybrid deep learning approaches to the HDC2021 challenge”. In: *Inverse Problems and Imaging* 17.5 (2023), pp. 908–924. ISSN: 1930-8337. DOI: 10.3934/ipi.2022061.

[21] S. Arridge et al. “Solving inverse problems using data-driven models”. In: *Acta Numerica* 28 (2019), pp. 1–174. DOI: 10.1017/S0962492919000059.

[22] K. Arulkumaran et al. “Deep Reinforcement Learning: A Brief Survey”. In: *IEEE Signal Processing Magazine* 34.6 (2017), pp. 26–38. DOI: 10.1109/MSP.2017.2743240.

[23] D. O. Baguer, J. Leuschner, and M. Schmidt. “Computed tomography reconstruction using deep image prior and learned reconstruction methods”. In: *Inverse Problems* 36.9 (2020), p. 094004. DOI: 10.1088/1361-6420/aba415.

[24] J. Bai et al. “Limited-view CT Reconstruction Based on Autoencoder-like Generative Adversarial Networks with Joint Loss”. In: *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. 2018, pp. 5570–5574. DOI: 10.1109/EMBC.2018.8513659.

[25] J. Bai, Y. Liu, and H. Yang. “Sparse-View CT Reconstruction Based on a Hybrid Domain Model with Multi-Level Wavelet Transform”. In: *Sensors* 22.9 (2022). ISSN: 1424-8220. DOI: 10.3390/s22093228.

[26] S. Bai, J. Z. Kolter, and V. Koltun. “Deep Equilibrium Models”. In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach et al. Vol. 32. Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper_files/paper/2019/file/01386bd6d8e091c2ab4c7c7de644d37b-Paper.pdf.

[27] Y. Bai et al. “Multi-Scale Hierarchy Feature Fusion Generative Adversarial Network for Low-Dose CT Denoising”. In: *2020 9th International Conference on Bioinformatics and Biomedical Science*. ICBBS ’20. Xiamen, China: Association for Computing Machinery, 2021, pp. 102–106. ISBN: 9781450388658. DOI: 10.1145/3431943.3432286.

[28] R. Balestriero et al. *A Cookbook of Self-Supervised Learning*. 2023. DOI: 10.48550/ARXIV.2304.12210.

[29] R. Barbano et al. "A Probabilistic Deep Image Prior over Image Space". In: *Fourth Symposium on Advances in Approximate Bayesian Inference*. 2022. URL: https://openreview.net/forum?id=qtFPfwJWowM.

[30] R. Barbano et al. "An Educated Warm Start for Deep Image Prior-Based Micro CT Reconstruction". In: *IEEE Transactions on Computational Imaging* 8 (2022), pp. 1210–1222. DOI: 10.1109/TCI.2022.3233188.

[31] R. Barbano et al. *Bayesian Experimental Design for Computed Tomography with the Linearised Deep Image Prior*. Presented at ICML Workshop on Adaptive Experimental Design and Active Learning in the Real World (ReALML) 2022, July 22, Baltimore, MD, USA. 2022. DOI: 10.48550/ARXIV.2207.05714.

[32] R. Barbano et al. "Chapter 26 - Uncertainty quantification in medical image synthesis". In: *Biomedical Image Synthesis and Simulation*. Ed. by N. Burgos and D. Svoboda. The MICCAI Society book Series. Academic Press, 2022, pp. 601–641. ISBN: 978-0-12-824349-7. DOI: https://doi.org/10.1016/B978-0-12-824349-7.00033-5.

[33] R. Barbano et al. *Image Reconstruction in Deep Image Prior Subspaces*. 2023. DOI: 10.48550/ARXIV.2302.10279.

[34] S. Barutcu et al. "Limited-angle computed tomography with deep image and physics priors". In: *Scientific Reports* 11.1 (2021), p. 17740. ISSN: 2045-2322. URL: https://doi.org/10.1038/s41598-021-97226-2.

[35] J. Batson and L. Royer. "Noise2Self: Blind Denoising by Self-Supervision". In: *Proceedings of the 36th International Conference on Machine Learning*. Ed. by K. Chaudhuri and R. Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, 2019, pp. 524–533. URL: https://proceedings.mlr.press/v97/batson19a.html.

[36] D. F. Bauer et al. "End-to-End Deep Learning CT Image Reconstruction for Metal Artifact Reduction". In: *Applied Sciences* 12.1 (2022). ISSN: 2076-3417. DOI: 10.3390/app12010404.

[37] T. Bayes and R. Price. "LII. An essay towards solving a problem in the doctrine of chances. By the late Rev. Mr. Bayes, F. R. S. communicated by Mr. Price, in a letter to John Canton, A. M. F. R. S". In: *Philosophical Transactions of the Royal Society of London* 53 (1763), pp. 370–418. DOI: 10.1098/rstl.1763.0053.

[38] J. Beaudry, P. L. Esquinas, and C.-C. Shieh. "Learning from our neighbours: a novel approach on sinogram completion using bin-sharing and deep learning to reconstruct high quality 4DCBCT". In: *Medical Imaging 2019: Physics of Medical Imaging*. Ed. by T. G. Schmidt, G.-H. Chen, and H. Bosmans. Vol. 10948. International Society for Optics and Photonics. SPIE, 2019, p. 1094847. DOI: 10.1117/12.2513168.

[39] M. Beckmann and A. Iske. "Error estimates for filtered back projection". In: *2015 International Conference on Sampling Theory and Applications (SampTA)*. 2015, pp. 553–557. DOI: 10.1109/SAMPTA.2015.7148952.

[40] F. J. Beekman and C. Kamphuis. "Ordered subset reconstruction for x-ray CT". In: *Physics in Medicine & Biology* 46.7 (2001), p. 1835. DOI: 10.1088/0031-9155/46/7/307.

[41] J. Behrmann et al. "Invertible Residual Networks". In: *Proceedings of the 36th International Conference on Machine Learning*. Ed. by K. Chaudhuri and R. Salakhutdinov. Vol. 97. Proceedings of Machine Learning Research. PMLR, 2019, pp. 573–582. URL: https://proceedings.mlr.press/v97/behrmann19a.html.

[42] M. Beister, D. Kolditz, and W. A. Kalender. "Iterative reconstruction methods in X-ray CT". In: *Physica Medica* 28.2 (2012), pp. 94–108. ISSN: 1120-1797. DOI: https://doi.org/10.1016/j.ejmp.2012.01.003.

[43] A. Ben-Israel and A. Charnes. "Contributions to the Theory of Generalized Inverses". In: *Journal of the Society for Industrial and Applied Mathematics* 11.3 (1963), pp. 667–699. DOI: 10.1137/0111051.

[44] A. Benali Amjoud and M. Amrouch. "Convolutional Neural Networks Backbones for Object Detection". In: *Image and Signal Processing*. Ed. by A. El Moataz et al. Cham: Springer International Publishing, 2020, pp. 282–289. ISBN: 978-3-030-51935-3. DOI: https://doi.org/10.1007/978-3-030-51935-3_30.

[45] M. Benning and M. Burger. "Modern regularization methods for inverse problems". In: *Acta Numerica* 27 (2018), pp. 1–111. DOI: 10.1017/S0962492918000016.

[46] T. M. Benson et al. "Block-based iterative coordinate descent". In: *IEEE Nuclear Science Symposium & Medical Imaging Conference*. 2010, pp. 2856–2859. DOI: 10.1109/NSSMIC.2010.5874316.

[47] F. J. Beutler. "The operator theory of the pseudo-inverse I. Bounded operators". In: *Journal of Mathematical Analysis and Applications* 10.3 (1965), pp. 451–470. ISSN: 0022-247X. DOI: 10.1016/0022-247X(65)90108-3.

[48] F. J. Beutler. "The operator theory of the pseudo-inverse II. Unbounded operators with arbitrary range". In: *Journal of Mathematical Analysis and Applications* 10.3 (1965), pp. 471–493. ISSN: 0022-247X. DOI: 10.1016/0022-247X(65)90109-5.

[49] D. Bianchi, G. Lai, and W. Li. "Uniformly convex neural networks and non-stationary iterated network Tikhonov (iNETT) method". In: *Inverse Problems* 39.5 (2023), p. 055002. DOI: 10.1088/1361-6420/acc2b6.

[50] G. Blanchard and P. Mathé. "Discrepancy principle for statistical inverse problems with application to conjugate gradient iteration*". In: *Inverse Problems* 28.11 (2012), p. 115011. DOI: 10.1088/0266-5611/28/11/115011.

[51] S. E. Blanke, B. N. Hahn, and A. Wald. "Inverse problems with inexact forward operator: iterative regularization and application in dynamic imaging". In: *Inverse Problems* 36.12 (2020), p. 124001. DOI: 10.1088/1361-6420/abb5e1.

[52] K. Boedeker. *AiCE Deep Learning Reconstruction: Bringing the power of Ultra-High Resolution CT to routine imaging*. 2019. URL: https://canonmedical.widen.net/content/kgyfdcnsdd/original/6368371730332299940U.pdf (visited on 07/26/2023).

[53] S. Bond-Taylor et al. "Deep Generative Modelling: A Comparative Review of VAEs, GANs, Normalizing Flows, Energy-Based and Autoregressive Models". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.11 (2022), pp. 7327–7347. DOI: 10.1109/TPAMI.2021.3116668.

[54] D. J. Brenner et al. "Cancer risks attributable to low doses of ionizing radiation: assessing what we really know". eng. In: *Proceedings of the National Academy of Sciences of the United States of America* 100 (24 2003), pp. 13761–6. DOI: 10.1073/pnas.2235592100.

[55] T. A. Bubba et al. "Learning the invisible: a hybrid deep learning-shearlet framework for limited angle computed tomography". In: *Inverse Problems* 35.6 (2019), p. 064002. DOI: 10.1088/1361-6420/ab10ca.

[56] T. A. Bubba et al. “Deep Neural Networks for Inverse Problems with Pseudodifferential Operators: An Application to Limited-Angle Tomography”. In: *SIAM Journal on Imaging Sciences* 14.2 (2021), pp. 470–505. DOI: 10.1137/20M1343075.

[57] T. A. Bubba et al. *Tomographic X-ray data of a lotus root filled with attenuating objects*. 2016. DOI: 10.48550/ARXIV.1609.07299.

[58] M. Burger and S. Osher. “Convergence rates of convex variational regularization”. In: *Inverse Problems* 20.5 (2004), p. 1411. DOI: 10.1088/0266-5611/20/5/005.

[59] L. Butzhammer and T. Hausotte. “Effect of iterative sparse-view CT reconstruction with task-specific projection angles on dimensional measurements”. In: *9th Conference on Industrial Computed Tomography, Padova, Italy (iCT2019)*. 2019.

[60] T. Buzug. *Computed Tomography: From Photon Statistics to Modern Cone-Beam CT*. Springer Berlin Heidelberg, 2008. ISBN: 9783540394082. DOI: 10.1007/978-3-540-39408-2.

[61] T. M. Buzug. “Computed Tomography”. In: *Springer Handbook of Medical Technology*. Ed. by R. Kramme, K.-P. Hoffmann, and R. S. Pozos. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 311–342. ISBN: 978-3-540-74658-4. DOI: 10.1007/978-3-540-74658-4_16.

[62] G. T. Buzzard et al. “Plug-and-Play Unplugged: Optimization-Free Reconstruction Using Consensus Equilibrium”. In: *SIAM Journal on Imaging Sciences* 11.3 (2018), pp. 2001–2020. DOI: 10.1137/17M1122451.

[63] F. Calivá et al. “Adversarial Robust Training of Deep Learning MRI Reconstruction Models”. In: *Machine Learning for Biomedical Imaging* 1 (MIDL 2020 special issue 2021), pp. 1–32. ISSN: 2766-905X. DOI: https://doi.org/10.59275/j.melba.2021-df47.

[64] P. Cascarano et al. “Combining Weighted Total Variation and Deep Image Prior for natural and medical image restoration via ADMM”. In: *2021 21st International Conference on Computational Science and Its Applications (ICCSA)*. 2021, pp. 39–46. DOI: 10.1109/ICCSA54496.2021.00016.

[65] P. Cascarano et al. “Constrained and unconstrained deep image prior optimization models with automatic regularization”. In: *Computational Optimization and Applications* 84.1 (2023), pp. 125–149. ISSN: 1573-2894. DOI: 10.1007/s10589-022-00392-w.

[66] Y. Censor et al. “On Diagonally Relaxed Orthogonal Projection Methods”. In: *SIAM Journal on Scientific Computing* 30.1 (2008), pp. 473–504. DOI: 10.1137/050639399.

[67] P. Chakrabarty and S. Maji. *The Spectral Bias of the Deep Image Prior*. Presented at Bayesian Deep Learning Workshop, Neural Information Processing Systems (NeurIPS) 2019, December 13, Vancouver, Canada. 2019. DOI: 10.48550/ARXIV.1912.08905.

[68] A. Chambolle and T. Pock. “A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging”. In: *Journal of Mathematical Imaging and Vision* 40.1 (2011), pp. 120–145. ISSN: 1573-7683. DOI: 10.1007/s10851-010-0251-1.

[69] G. Chen et al. “4D-AirNet: a temporally-resolved CBCT slice reconstruction method synergizing analytical and iterative method with deep learning”. In: *Physics in Medicine & Biology* 65.17 (2020), p. 175020. DOI: 10.1088/1361-6560/ab9f60.

[70] G. Chen et al. “AirNet: Fused analytical and iterative reconstruction with deep neural network regularization for sparse-data CT”. In: *Medical Physics* 47.7 (2020), pp. 2916–2930. DOI: https://doi.org/10.1002/mp.14170.

[71] H. Chen et al. "LEARN: Learned Experts' Assessment-Based Reconstruction Network for Sparse-Data CT". In: *IEEE Transactions on Medical Imaging* 37.6 (2018), pp. 1333–1347. DOI: 10.1109/TMI.2018.2805692.

[72] H. Chen et al. "Low-dose CT denoising with convolutional neural network". In: *2017 IEEE 14th International Symposium on Biomedical Imaging (ISBI 2017)*. 2017, pp. 143–146. DOI: 10.1109/ISBI.2017.7950488.

[73] H. Chen et al. "Low-dose CT restoration with deep neural network". In: *Proc. 14th Int. Meeting Fully Three-Dimensional Image Reconstruction Radiol. Nucl. Med.* 2017, pp. 25–28. DOI: 10.12059/Fully3D.2017-11-3202013.

[74] H. Chen et al. "Low-Dose CT With a Residual Encoder-Decoder Convolutional Neural Network". In: *IEEE Transactions on Medical Imaging* 36.12 (2017), pp. 2524–2535. DOI: 10.1109/TMI.2017.2715284.

[75] Y.-J. Chen et al. "CT Image Denoising With Encoder-Decoder Based Graph Convolutional Networks". In: *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*. 2021, pp. 400–404. DOI: 10.1109/ISBI48211.2021.9433900.

[76] Y. Chen, W. Yu, and T. Pock. "On learning optimized reaction diffusion processes for effective image restoration". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 5261–5269. DOI: 10.1109/CVPR.2015.7299163.

[77] Z. Cheng et al. "A Bayesian Perspective on the Deep Image Prior". In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 5438–5446. DOI: 10.1109/CVPR.2019.00559.

[78] H. Choi et al. "Dose reduction potential of vendor-agnostic deep learning model in comparison with deep learning-based image reconstruction algorithm on CT: a phantom study". In: *European Radiology* 32.2 (2022), pp. 1247–1255. ISSN: 1432-1084. DOI: 10.1007/s00330-021-08199-9.

[79] C.-Y. Chou et al. "A fast forward projection using multithreads for multirays on GPUs in medical image reconstruction". In: *Medical Physics* 38.7 (2011), pp. 4052–4065. DOI: https://doi.org/10.1118/1.3591994.

[80] I. Y. Chun and J. A. Fessler. "Convolutional Analysis Operator Learning: Acceleration and Convergence". In: *IEEE Transactions on Image Processing* 29 (2020), pp. 2108–2122. DOI: 10.1109/TIP.2019.2937734.

[81] I. Y. Chun and J. A. Fessler. "Convolutional Dictionary Learning: Acceleration and Convergence". In: *IEEE Transactions on Image Processing* 27.4 (2018), pp. 1697–1712. DOI: 10.1109/TIP.2017.2761545.

[82] I. Y. Chun et al. "Momentum-Net: Fast and convergent iterative neural network for inverse problems". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2020), pp. 1–1. DOI: 10.1109/TPAMI.2020.3012955.

[83] I. Y. Chun et al. "Sparse-view X-ray CT reconstruction using ℓ_1 regularization with learned sparsifying transform". In: *Proc. Intl. Mtg. on Fully 3D Image Recon. in Rad. and Nuc. Med.* 2017, pp. 115–9. DOI: 10.12059/Fully3D.2017-11-3109002.

[84] H. Chung et al. "Improving Diffusion Models for Inverse Problems using Manifold Constraints". In: *Advances in Neural Information Processing Systems*. Ed. by A. H. Oh et al. 2022. URL: https://openreview.net/forum?id=nJJjv0JDJju.

[85] K. J. Chung et al. “Low-dose CT Enhancement Network with a Perceptual Loss Function in the Spatial Frequency and Image Domains”. In: *Medical Imaging with Deep Learning*. 2020. URL: https://openreview.net/forum?id=rw5BswbvMB.

[86] S. B. Coban et al. *Parallel-beam X-ray CT datasets of apples with internal defects and label balancing for machine learning*. 2020. DOI: 10.48550/ARXIV.2012.13346.

[87] F.-A. Croitoru et al. “Diffusion Models in Vision: A Survey”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023), pp. 1–20. DOI: 10.1109/TPAMI.2023.3261988.

[88] J. Cui et al. “PET image denoising using unsupervised deep learning”. In: *European Journal of Nuclear Medicine and Molecular Imaging* 46.13 (2019), pp. 2780–2789. ISSN: 1619-7089. DOI: 10.1007/s00259-019-04468-4.

[89] J. Cui et al. “Populational and individual information based PET image denoising using conditional unsupervised learning”. In: *Physics in Medicine & Biology* 66.15 (2021), p. 155001. DOI: 10.1088/1361-6560/ac108e.

[90] J. Cui et al. “SURE-based Stopping Strategy for Fine-tunable Supervised PET Image Denoising”. In: *2021 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*. 2021, pp. 1–3. DOI: 10.1109/NSS/MIC44867.2021.9875858.

[91] G. Cybenko. “Approximation by superpositions of a sigmoidal function”. In: *Mathematics of Control, Signals and Systems* 2.4 (1989), pp. 303–314. ISSN: 1435-568X. DOI: 10.1007/BF02551274.

[92] K. Dabov et al. “Image Denoising by Sparse 3-D Transform-Domain Collaborative Filtering”. In: *IEEE Transactions on Image Processing* 16.8 (2007), pp. 2080–2095. DOI: 10.1109/TIP.2007.901238.

[93] M. E. Davison. “The Ill-Conditioned Nature of the Limited Angle Tomography Problem”. In: *SIAM Journal on Applied Mathematics* 43.2 (1983), pp. 428–448. DOI: 10.1137/0143028.

[94] Q. De Man et al. “A two-dimensional feasibility study of deep learning-based feature detection and characterization directly from CT sinograms”. In: *Medical Physics* 46.12 (2019), e790–e800. DOI: https://doi.org/10.1002/mp.13640.

[95] J. Deng et al. “ImageNet: A large-scale hierarchical image database”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[96] A. Denker et al. “Conditional Invertible Neural Networks for Medical Imaging”. In: *Journal of Imaging* 7.11 (2021). ISSN: 2313-433X. DOI: 10.3390/jimaging7110243.

[97] A. Denker et al. *Conditional Normalizing Flows for Low-Dose Computed Tomography Image Reconstruction*. Presented at ICML Workshop on Invertible Neural Networks, Normalizing Flows, and Explicit Likelihood Models 2020, July 18, Vienna, Austria. 2020. URL: https://invertibleworkshop.github.io/INNF_2020/accepted_papers/index.html.

[98] F. Dennerlein. “Image Reconstruction from Fan-Beam and Cone-Beam Projections”. doctoralthesis. Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU), 2009. URL: https://nbn-resolving.org/urn:nbn:de:bvb:29-opus-12575.

[99] H. Der Sarkissian et al. “A cone-beam X-ray computed tomography data collection designed for machine learning”. In: *Scientific Data* 6.1 (2019), p. 215. ISSN: 2052-4463. DOI: 10.1038/s41597-019-0235-y.

[100] Q. Ding et al. “A dataset-free deep learning method for low-dose CT image reconstruction”. In: *Inverse Problems* 38.10 (2022), p. 104003. DOI: 10.1088/1361-6420/ac8ac6.

[101] Q. Ding et al. “Learnable Multi-scale Fourier Interpolation for Sparse View CT Image Reconstruction”. In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2021*. Ed. by M. de Bruijne et al. Cham: Springer International Publishing, 2021, pp. 286–295. ISBN: 978-3-030-87231-1. DOI: https://doi.org/10.1007/978-3-030-87231-1_28.

[102] L. Dinh, D. Krueger, and Y. Bengio. *NICE: Non-linear Independent Components Estimation*. 2014. DOI: 10.48550/ARXIV.1410.8516.

[103] L. Dinh, J. Sohl-Dickstein, and S. Bengio. “Density estimation using Real NVP”. In: *International Conference on Learning Representations*. 2017. URL: https://openreview.net/forum?id=HkpbnH9lx.

[104] S. Dittmer et al. “Deep image prior for 3D magnetic particle imaging: A quantitative comparison of regularization techniques on Open MPI dataset”. In: *International Journal on Magnetic Particle Imaging* 7.1 (2021). DOI: 10.18416/IJMPI.2021.2103001.

[105] S. Dittmer et al. “Regularization by Architecture: A Deep Prior Approach for Inverse Problems”. In: *Journal of Mathematical Imaging and Vision* 62.3 (2020), pp. 456–470. ISSN: 1573-7683. DOI: 10.1007/s10851-019-00923-x.

[106] J. Dong, J. Fu, and Z. He. “A deep learning reconstruction framework for X-ray computed tomography with incomplete data”. In: *PLOS ONE* 14.11 (2019), pp. 1–17. DOI: 10.1371/journal.pone.0224426.

[107] X. Dong, S. Vekhande, and G. Cao. “Sinogram interpolation for sparse-view micro-CT with deep learning neural network”. In: *Medical Imaging 2019: Physics of Medical Imaging*. Ed. by T. G. Schmidt, G.-H. Chen, and H. Bosmans. Vol. 10948. International Society for Optics and Photonics. SPIE, 2019, 109482O. DOI: 10.1117/12.2512979.

[108] K. Dremel. “Modellbildung des Messprozesses und Umsetzung eines modellbasierten iterativen Lösungsverfahrens der Schnittbild-Rekonstruktion für die Röntgen-Computertomographie”. doctoralthesis. Universität Würzburg, 2018. URL: https://opus.bibliothek.uni-wuerzburg.de/opus4-wuerzburg/frontdoor/deliver/index/docId/15771/file/Dremel_Kilian_Dissertation.pdf (visited on 07/27/2023).

[109] P. Drineas et al. “Fast Approximation of Matrix Coherence and Statistical Leverage”. In: *Journal of Machine Learning Research* 13.111 (2012), pp. 3475–3506. URL: http://jmlr.org/papers/v13/drineas12a.html.

[110] C. Ekmekci and M. Cetin. “Uncertainty Quantification for Deep Unrolling-Based Computational Imaging”. In: *IEEE Transactions on Computational Imaging* 8 (2022), pp. 1195–1209. DOI: 10.1109/TCI.2022.3233185.

[111] M. Elasri et al. “Image Generation: A Review”. In: *Neural Processing Letters* 54.5 (2022), pp. 4609–4646. ISSN: 1573-773X. DOI: 10.1007/s11063-022-10777-x.

[112] I. A. Elbakri and J. A. Fessler. “Segmentation-free statistical image reconstruction for polyenergetic x-ray computed tomography with experimental validation”. In: *Physics in Medicine & Biology* 48.15 (2003), p. 2453. DOI: 10.1088/0031-9155/48/15/314.

[113] Y. C. Eldar. “Generalized SURE for Exponential Families: Applications to Regularization”. In: *IEEE Transactions on Signal Processing* 57.2 (2009), pp. 471–481. DOI: 10.1109/TSP.2008.2008212.

[114] H. W. Engl. “Discrepancy principles for Tikhonov regularization of ill-posed problems leading to optimal convergence rates”. In: *Journal of Optimization Theory and Applications* 52.2 (1987), pp. 209–215. ISSN: 1573-2878. DOI: 10.1007/BF00941281.

[115] H. Engl, M. Hanke, and A. Neubauer. *Regularization of Inverse Problems*. Mathematics and Its Applications. Springer Netherlands, 2000. ISBN: 9780792361404.

[116] H. W. Engl. "Necessary and sufficient conditions for convergence of regularization methods for solving linear operator equations of the first kind". In: *Numerical Functional Analysis and Optimization* 3.2 (1981), pp. 201–222. DOI: 10.1080/01630568108816087.

[117] I. Erdelyi. "A generalized inverse for arbitrary operators between Hilbert spaces". In: *Mathematical Proceedings of the Cambridge Philosophical Society* 71.1 (1972), pp. 43–50. DOI: 10.1017/S0305004100050246.

[118] C. Etmann, R. Ke, and C.-B. Schönlieb. "iUNets: Learnable Invertible Up- and Downsampling for Large-Scale Inverse Problems". In: *2020 IEEE 30th International Workshop on Machine Learning for Signal Processing (MLSP)*. 2020, pp. 1–6. DOI: 10.1109/MLSP49062.2020.9231874.

[119] T. G. Feeman. *The Mathematics of Medical Imaging*. Springer International Publishing, 2015. DOI: 10.1007/978-3-319-22665-1.

[120] L. A. Feldkamp, L. C. Davis, and J. W. Kress. "Practical cone-beam algorithm". In: *J. Opt. Soc. Am. A* 1.6 (1984), pp. 612–619. DOI: 10.1364/JOSAA.1.000612.

[121] R. Fermanian, M. Le Pendu, and C. Guillemot. "Regularizing the Deep Image Prior with a Learned Denoiser for Linear Inverse Problems". In: *2021 IEEE 23rd International Workshop on Multimedia Signal Processing (MMSP)*. 2021, pp. 1–6. DOI: 10.1109/MMSP53017.2021.9733691.

[122] J. Flemming and B. Hofmann. "A New Approach to Source Conditions in Regularization with General Residual Term". In: *Numerical Functional Analysis and Optimization* 31.3 (2010), pp. 254–284. DOI: 10.1080/01630561003765721.

[123] J. Frikel. "Reconstructions in limited angle x-ray tomography: Characterization of classical reconstructions and adapted curvelet sparse regularization". doctoralthesis. 2013. URL: https://nbn-resolving.de/urn/resolver.pl?urn:nbn:de:bvb:91-diss-20130328-1115037-0-3.

[124] L. Fu and B. De Man. "A hierarchical approach to deep learning and its application to tomographic reconstruction". In: *15th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*. Ed. by S. Matej and S. D. Metzler. Vol. 11072. International Society for Optics and Photonics. SPIE, 2019, p. 1107202. DOI: 10.1117/12.2534615.

[125] L. Fu and B. De Man. "Deep learning tomographic reconstruction through hierarchical decomposition of domain transforms". In: *Visual Computing for Industry, Biomedicine, and Art* 5.1 (2022), p. 30. ISSN: 2524-4442. DOI: 10.1186/s42492-022-00127-y.

[126] L. Fu et al. "Comparison Between Pre-Log and Post-Log Statistical Models in Ultra-Low-Dose CT Reconstruction". In: *IEEE Transactions on Medical Imaging* 36.3 (2017), pp. 707–720. DOI: 10.1109/TMI.2016.2627004.

[127] K. Fukushima. "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position". In: *Biological Cybernetics* 36.4 (1980), pp. 193–202. ISSN: 1432-0770. DOI: 10.1007/BF00344251.

[128] S. W. Fung et al. "JFB: Jacobian-Free Backpropagation for Implicit Networks". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 36.6 (2022), pp. 6648–6656. DOI: 10.1609/aaai.v36i6.20619.

[129] M. Gadelha, R. Wang, and S. Maji. "Shape Reconstruction Using Differentiable Projections and Deep Priors". In: *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2019, pp. 22–30. DOI: 10.1109/ICCV.2019.00011.

[130] Y. Gal and Z. Ghahramani. "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning". In: *Proceedings of The 33rd International Conference on Machine Learning*. Ed. by M. F. Balcan and K. Q. Weinberger. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, 2016, pp. 1050–1059. URL: https://proceedings.mlr.press/v48/gal16.html.

[131] H. Gao. "Fused analytical and iterative reconstruction (AIR) via modified proximal forward–backward splitting: a FDK-based iterative image reconstruction example for CBCT". In: *Physics in Medicine & Biology* 61.19 (2016), p. 7187. DOI: 10.1088/0031-9155/61/19/7187.

[132] Y. Gao et al. "A machine learning approach to construct a tissue-specific texture prior from previous full-dose CT for Bayesian reconstruction of current ultralow-dose CT images". In: *15th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*. Ed. by S. Matej and S. D. Metzler. Vol. 11072. International Society for Optics and Photonics. SPIE, 2019, p. 1107204. DOI: 10.1117/12.2534441.

[133] Y. Gao et al. "Improved computer-aided detection of pulmonary nodules via deep learning in the sinogram domain". In: *Visual Computing for Industry, Biomedicine, and Art* 2.1 (2019), p. 15. ISSN: 2524-4442. DOI: 10.1186/s42492-019-0029-2.

[134] C. Garbin, X. Zhu, and O. Marques. "Dropout vs. batch normalization: an empirical study of their impact to deep learning". In: *Multimedia Tools and Applications* 79.19 (2020), pp. 12777–12815. ISSN: 1573-7721. DOI: 10.1007/s11042-019-08453-9.

[135] R. Ge et al. "DDPNet: A Novel Dual-Domain Parallel Network for Low-Dose CT Reconstruction". In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2022*. Ed. by L. Wang et al. Cham: Springer Nature Switzerland, 2022, pp. 748–757. ISBN: 978-3-031-16446-0. DOI: 10.1007/978-3-031-16446-0_71.

[136] Y. Ge et al. "ADAPTIVE-NET: deep computed tomography reconstruction network with analytical domain transformation knowledge". In: *Quantitative Imaging in Medicine and Surgery* 10.2 (2020). ISSN: 2223-4306. URL: https://qims.amegroups.com/article/view/34393.

[137] M. Genzel, J. Macdonald, and M. März. "Solving Inverse Problems With Deep Neural Networks – Robustness Included?" In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 45.1 (2023), pp. 1119–1134. DOI: 10.1109/TPAMI.2022.3148324.

[138] M. U. Ghani and W. C. Karl. "Deep Learning Based Sinogram Correction for Metal Artifact Reduction". In: *Electronic Imaging* 30.15 (2018), pp. 472–1–4728. DOI: 10.2352/issn.2470-1173.2018.15.coimg-472.

[139] M. U. Ghani and W. C. Karl. "Deep Learning-Based Sinogram Completion for Low-Dose CT". In: *2018 IEEE 13th Image, Video, and Multidimensional Signal Processing Workshop (IVMSP)*. 2018, pp. 1–5. DOI: 10.1109/IVMSPW.2018.8448403.

[140] M. U. Ghani and W. C. Karl. "Fast Enhanced CT Metal Artifact Reduction Using Data Domain Deep Learning". In: *IEEE Transactions on Computational Imaging* 6 (2020), pp. 181–193. DOI: 10.1109/TCI.2019.2937221.

[141] P. Gilbert. "Iterative methods for the three-dimensional reconstruction of an object from projections". In: *Journal of Theoretical Biology* 36.1 (1972), pp. 105–117. ISSN: 0022-5193. DOI: https://doi.org/10.1016/0022-5193(72)90180-4.

[142] D. Gilton, G. Ongie, and R. Willett. "Deep Equilibrium Architectures for Inverse Problems in Imaging". In: *IEEE Transactions on Computational Imaging* 7 (2021), pp. 1123–1133. DOI: 10.1109/TCI.2021.3118944.

[143] L. Gjesteby et al. "Deep learning methods to guide CT image reconstruction and reduce metal artifacts". In: *Medical Imaging 2017: Physics of Medical Imaging*. Ed. by T. G. Flohr, J. Y. Lo, and T. G. Schmidt. Vol. 10132. International Society for Optics and Photonics. SPIE, 2017, 101322W. DOI: 10.1117/12.2254091.

[144] L. Gjesteby et al. "Metal Artifact Reduction in CT: Where Are We After Four Decades?" In: *IEEE Access* 4 (2016), pp. 5826–5849. DOI: 10.1109/ACCESS.2016.2608621.

[145] X. Glorot, A. Bordes, and Y. Bengio. "Deep Sparse Rectifier Neural Networks". In: *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. Ed. by G. Gordon, D. Dunson, and M. Dudík. Vol. 15. Proceedings of Machine Learning Research. Fort Lauderdale, FL, USA: PMLR, 2011, pp. 315–323. URL: https://proceedings.mlr.press/v15/glorot11a.html.

[146] K. Gong et al. "Low-dose dual energy CT image reconstruction using non-local deep image prior". In: *2019 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*. 2019, pp. 1–2. DOI: 10.1109/NSS/MIC42101.2019.9060001.

[147] K. Gong et al. "PET Image Reconstruction Using Deep Image Prior". In: *IEEE Transactions on Medical Imaging* 38.7 (2019), pp. 1655–1665. DOI: 10.1109/TMI.2018.2888491.

[148] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[149] I. J. Goodfellow et al. "Generative Adversarial Nets". In: *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*. Ed. by Z. Ghahramani et al. 2014, pp. 2672–2680. URL: https://proceedings.neurips.cc/paper/2014/hash/5ca3e9b122f61f8f06494c97b1afccf3-Abstract.html.

[150] R. Gordon, R. Bender, and G. T. Herman. "Algebraic Reconstruction Techniques (ART) for three-dimensional electron microscopy and X-ray photography". In: *Journal of Theoretical Biology* 29.3 (1970), pp. 471–481. ISSN: 0022-5193. DOI: https://doi.org/10.1016/0022-5193(70)90109-8.

[151] R. Gothwal, S. Tiwari, and S. Shivani. "Computational Medical Image Reconstruction Techniques: A Comprehensive Review". In: *Archives of Computational Methods in Engineering* 29.7 (2022), pp. 5635–5662. ISSN: 1886-1784. DOI: 10.1007/s11831-022-09785-w.

[152] A. Goujon et al. *A Neural-Network-Based Convex Regularizer for Inverse Problems*. 2023. DOI: 10.48550/ARXIV.2211.12461.

[153] J. Greffier et al. "First Results of a New Deep Learning Reconstruction Algorithm on Image Quality and Liver Metastasis Conspicuity for Abdominal Low-Dose CT". In: *Diagnostics* 13.6 (2023). ISSN: 2075-4418. DOI: 10.3390/diagnostics13061182.

[154] J. Gui et al. "A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications". In: *IEEE Transactions on Knowledge and Data Engineering* (2021), pp. 1–1. DOI: 10.1109/TKDE.2021.3130191.

[155] M.-H. Guo et al. "Attention mechanisms in computer vision: A survey". In: *Computational Visual Media* 8.3 (2022), pp. 331–368. ISSN: 2096-0662. DOI: 10.1007/s41095-022-0271-y.

[156] Y. Guo et al. "Dual domain closed-loop learning for sparse-view CT reconstruction". In: *7th International Conference on Image Formation in X-Ray Computed Tomography*. Ed. by J. W. Stayman. Vol. 12304. International Society for Optics and Photonics. SPIE, 2022, p. 123040M. DOI: 10.1117/12.2646639.

[157] H. Gupta et al. "CNN-Based Projected Gradient Descent for Consistent CT Image Reconstruction". In: *IEEE Transactions on Medical Imaging* 37.6 (2018), pp. 1440–1453. DOI: 10.1109/TMI.2018.2832656.

[158] A. Guzhva, S. Dolenko, and I. Persiantsev. "Multifold Acceleration of Neural Network Computations Using GPU". In: *Artificial Neural Networks – ICANN 2009*. Ed. by C. Alippi et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 373–380. ISBN: 978-3-642-04274-4. DOI: 10.1007/978-3-642-04274-4_39.

[159] J. Hadamard. *Lectures on Cauchy's Problem in Linear Partial Differential Equations*. Yale University Press, 1923.

[160] K. Hammernik et al. "A Deep Learning Architecture for Limited-Angle Computed Tomography Reconstruction". In: *Bildverarbeitung für die Medizin 2017*. Ed. by K. H. Maier-Hein geb. Fritzsche et al. Berlin, Heidelberg: Springer Berlin Heidelberg, 2017, pp. 92–97. ISBN: 978-3-662-54345-0. DOI: 10.1007/978-3-662-54345-0_25.

[161] Y. S. Han, J. Yoo, and J. C. Ye. *Deep Residual Learning for Compressed Sensing CT Reconstruction via Persistent Homology Analysis*. 2016. DOI: 10.48550/ARXIV.1611.06391.

[162] M. Hanke, A. Neubauer, and O. Scherzer. "A convergence analysis of the Landweber iteration for nonlinear ill-posed problems". In: *Numerische Mathematik* 72.1 (1995), pp. 21–37. ISSN: 0945-3245. DOI: 10.1007/s002110050158.

[163] P. C. Hansen. "The discrete picard condition for discrete ill-posed problems". In: *BIT Numerical Mathematics* 30.4 (1990), pp. 658–672. ISSN: 1572-9125. DOI: 10.1007/BF01933214.

[164] X. L. Hao Wu Qi Liu. "A Review on Deep Learning Approaches to Image Classification and Object Segmentation". In: *Computers, Materials & Continua* 60.2 (2019), pp. 575–597. ISSN: 1546-2226. DOI: 10.32604/cmc.2019.03595.

[165] J. Harms et al. "Paired cycle-GAN-based image correction for quantitative cone-beam computed tomography". In: *Medical Physics* 46.9 (2019), pp. 3998–4009. DOI: https://doi.org/10.1002/mp.13656.

[166] M. Hasani and H. Khotanlou. "An Empirical Study on Position of the Batch Normalization Layer in Convolutional Neural Networks". In: *2019 5th Iranian Conference on Signal Processing and Intelligent Systems (ICSPIS)*. 2019, pp. 1–4. DOI: 10.1109/ICSPIS48872.2019.9066113.

[167] A. Hauptmann et al. "Multi-Scale Learned Iterative Reconstruction." eng. In: *IEEE transactions on computational imaging* 6 (2020), pp. 843–856. DOI: 10.1109/TCI.2020.2990299.

[168] J. He, Y. Wang, and J. Ma. "Radon Inversion via Deep Learning". In: *IEEE Transactions on Medical Imaging* 39.6 (2020), pp. 2076–2087. DOI: 10.1109/TMI.2020.2964266.

[169] J. He et al. "Optimizing a Parameterized Plug-and-Play ADMM for Iterative Low-Dose CT Reconstruction". In: *IEEE Transactions on Medical Imaging* 38.2 (2019), pp. 371–382. DOI: 10.1109/TMI.2018.2865202.

[170] K. He et al. “Deep Residual Learning for Image Recognition”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[171] K. He et al. “Identity Mappings in Deep Residual Networks”. In: *Computer Vision – ECCV 2016*. Ed. by B. Leibe et al. Cham: Springer International Publishing, 2016, pp. 630–645. ISBN: 978-3-319-46493-0. DOI: 10.1007/978-3-319-46493-0_38.

[172] H. Heaton et al. “Feasibility-based fixed point networks”. In: *Fixed Point Theory and Algorithms for Sciences and Engineering* 2021.1 (2021), p. 21. ISSN: 2730-5422. DOI: 10.1186/s13663-021-00706-3.

[173] D. O. Hebb. “The organization of behavior, London (Chapman & Hall) 1949.” In: 1949.

[174] R. Heckel and P. Hand. “Deep Decoder: Concise Image Representations from Untrained Non-convolutional Networks”. In: *International Conference on Learning Representations*. 2019. URL: https://openreview.net/forum?id=rylV-2C9KQ.

[175] R. Heckel and M. Soltanolkotabi. “Denoising and Regularization via Exploiting the Structural Bias of Convolutional Generators”. In: *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL: https://openreview.net/forum?id=HJeqhA4YDS.

[176] M. P. Heinrich, M. Stille, and T. M. Buzug. “Residual U-Net Convolutional Neural Network Architecture for Low-Dose CT Denoising”. In: *Current Directions in Biomedical Engineering* 4.1 (2018), pp. 297–300. DOI: doi:10.1515/cdbme-2018-0072.

[177] O. J. Hénaff and E. P. Simoncelli. “Geodesics of learned representations”. In: *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*. Ed. by Y. Bengio and Y. LeCun. 2016. DOI: 10.48550/ARXIV.1511.06394.

[178] A. A. Hendriksen et al. “Deep denoising for multi-dimensional synchrotron X-ray tomography without high-quality reference data”. In: *Scientific Reports* 11.1 (2021), p. 11895. ISSN: 2045-2322. DOI: 10.1038/s41598-021-91084-8.

[179] A. A. Hendriksen, D. M. Pelt, and K. J. Batenburg. “Noise2Inverse: Self-Supervised Deep Convolutional Denoising for Tomography”. In: *IEEE Transactions on Computational Imaging* 6 (2020), pp. 1320–1335. DOI: 10.1109/TCI.2020.3019647.

[180] G. Herl, A. Maier, and S. Zabler. “X-ray CT data completeness condition for sets of arbitrary projections”. In: *7th International Conference on Image Formation in X-Ray Computed Tomography*. Ed. by J. W. Stayman. Vol. 12304. International Society for Optics and Photonics. SPIE, 2022, p. 123040C. DOI: 10.1117/12.2646435.

[181] G. E. Hinton, S. Osindero, and Y.-W. Teh. “A Fast Learning Algorithm for Deep Belief Nets”. In: *Neural Computation* 18.7 (2006), pp. 1527–1554. ISSN: 0899-7667. DOI: 10.1162/neco.2006.18.7.1527.

[182] S. Hochreiter et al. “Gradient flow in recurrent nets: the difficulty of learning long-term dependencies”. In: *A Field Guide to Dynamical Recurrent Neural Networks*. Ed. by S. C. Kremer and J. F. Kolen. IEEE Press, 2001. URL: https://ieeexplore.ieee.org/document/5264952.

[183] S. Hochreiter and J. Schmidhuber. “Long Short-Term Memory”. In: *Neural Computation* 9.8 (1997), pp. 1735–1780. ISSN: 0899-7667. DOI: 10.1162/neco.1997.9.8.1735.

[184] B. Hofmann and S. Kindermann. "On the degree of ill-posedness for linear problems with noncompact operators". In: *Methods and Applications of Analysis* 17.4 (2010), pp. 445–462. DOI: 10.4310/MAA.2010.v17.n4.a8.

[185] K. Hornik. "Approximation capabilities of multilayer feedforward networks". In: *Neural Networks* 4.2 (1991), pp. 251–257. ISSN: 0893-6080. DOI: 10.1016/0893-6080(91)90009-T.

[186] K. Hornik, M. Stinchcombe, and H. White. "Multilayer feedforward networks are universal approximators". In: *Neural Networks* 2.5 (1989), pp. 359–366. ISSN: 0893-6080. DOI: 10.1016/0893-6080(89)90020-8.

[187] J. Hsieh et al. *A new era of image reconstruction: TrueFidelity™ - Technical white paper on deep learning image reconstruction*. 2019. URL: https://www.gehealthcare.com/-/jssmedia/040dd213fa89463287155151fdb01922.pdf (visited on 07/18/2023).

[188] D. Hu et al. "Hybrid-Domain Neural Network Processing for Sparse-View CT Reconstruction". In: *IEEE Transactions on Radiation and Plasma Medical Sciences* 5.1 (2021), pp. 88–98. DOI: 10.1109/TRPMS.2020.3011413.

[189] D. Hu et al. "PRIOR: Prior-Regularized Iterative Optimization Reconstruction For 4D CBCT". In: *IEEE Journal of Biomedical and Health Informatics* 26.11 (2022), pp. 5551–5562. DOI: 10.1109/JBHI.2022.3201232.

[190] Y. Huang et al. "Data Consistent Artifact Reduction for Limited Angle Tomography with Deep Learning Prior". In: *Machine Learning for Medical Image Reconstruction*. Ed. by F. Knoll et al. Cham: Springer International Publishing, 2019, pp. 101–112. ISBN: 978-3-030-33843-5. DOI: 10.1007/978-3-030-33843-5_10.

[191] Y. Huang et al. "Data Extrapolation From Learned Prior Images for Truncation Correction in Computed Tomography". In: *IEEE Transactions on Medical Imaging* 40.11 (2021), pp. 3042–3053. DOI: 10.1109/TMI.2021.3072568.

[192] Y. Huang et al. "Field of View Extension in Computed Tomography Using Deep Learning Prior". In: *Bildverarbeitung für die Medizin 2020*. Ed. by T. Tolxdorff et al. Wiesbaden: Springer Fachmedien Wiesbaden, 2020, pp. 186–191. ISBN: 978-3-658-29267-6. DOI: 10.1007/978-3-658-29267-6_40.

[193] Z. Huang et al. "DU-GAN: Generative Adversarial Networks With Dual-Domain U-Net-Based Discriminators for Low-Dose CT Denoising". In: *IEEE Transactions on Instrumentation and Measurement* 71 (2022), pp. 1–12. DOI: 10.1109/TIM.2021.3128703.

[194] D. H. Hubel and T. N. Wiesel. "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex." eng. In: *The Journal of physiology* 160 (1 1962), pp. 106–54. DOI: 10.1113/jphysiol.1962.sp006837.

[195] J. Idier. *Bayesian Approach to Inverse Problems*. ISTE. Wiley, 2013. ISBN: 9781118623695.

[196] J. A. Iglesias, G. Mercier, and O. Scherzer. "A note on convergence of solutions of total variation regularized linear inverse problems". In: *Inverse Problems* 34.5 (2018), p. 055011. DOI: 10.1088/1361-6420/aab92a.

[197] S. Ioffe and C. Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". In: *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*. Ed. by F. R. Bach and D. M. Blei. Vol. 37. JMLR Workshop and Conference Proceedings. JMLR.org, 2015, pp. 448–456. URL: http://proceedings.mlr.press/v37/ioffe15.html.

[198] P. Isola et al. “Image-to-Image Translation with Conditional Adversarial Networks”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 5967–5976. DOI: 10.1109/CVPR.2017.632.

[199] A. G. Ivakhnenko. “Polynomial Theory of Complex Systems”. In: *IEEE Transactions on Systems, Man, and Cybernetics* SMC-1.4 (1971), pp. 364–378. DOI: 10.1109/TSMC.1971.4308320.

[200] A. Ivakhnenko et al. *Cybernetics and Forecasting Techniques*. Modern analytic and computational methods in science and mathematics. American Elsevier Publishing Company, 1967. ISBN: 9780444000200.

[201] J.-H. Jacobsen, A. W. Smeulders, and E. Oyallon. “i-RevNet: Deep Invertible Networks”. In: *International Conference on Learning Representations*. 2018. URL: https://openreview.net/forum?id=HJsjkMb0Z.

[202] M. Jessop et al. “Image quality in low-dose CT: Correlation of Image Quality Metrics (IQM) and Observer Performance”. In: *Journal of Medical Imaging and Radiation Sciences* 53.4, Supplement 1 (2022). ISRRT conference proceedings, S15. ISSN: 1939-8654. DOI: https://doi.org/10.1016/j.jmir.2022.10.051.

[203] L. Jiang et al. “Sparse-View CBCT Reconstruction Using Combined DRUNet and HQS”. In: *2022 5th International Conference on Pattern Recognition and Artificial Intelligence (PRAI)*. 2022, pp. 1051–1054. DOI: 10.1109/PRAI55851.2022.9904278.

[204] F. Jiao et al. “A Dual-Domain CNN-Based Network for CT Reconstruction”. In: *IEEE Access* 9 (2021), pp. 71091–71103. DOI: 10.1109/ACCESS.2021.3079323.

[205] K. H. Jin et al. “Deep Convolutional Neural Network for Inverse Problems in Imaging”. In: *IEEE Transactions on Image Processing* 26.9 (2017), pp. 4509–4522. DOI: 10.1109/TIP.2017.2713099.

[206] Q. Jin and M. Zhong. “Nonstationary iterated Tikhonov regularization in Banach spaces with uniformly convex penalty terms”. In: *Numerische Mathematik* 127.3 (2014), pp. 485–513. ISSN: 0945-3245. DOI: 10.1007/s00211-013-0594-9.

[207] Y. Jo, S. Y. Chun, and J. Choi. “Rethinking Deep Image Prior for Denoising”. In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 5067–5076. DOI: 10.1109/ICCV48922.2021.00504.

[208] M. John et al. *Deep Image Prior using Stein’s Unbiased Risk Estimator: SURE-DIP*. 2021. DOI: 10.48550/ARXIV.2111.10892.

[209] S. Kabri et al. *Convergent Data-driven Regularizations for CT Reconstruction*. 2022. DOI: 10.48550/ARXIV.2212.07786.

[210] J. P. Kaipio and E. Somersalo. “Statistical Inversion Theory”. In: *Statistical and Computational Inverse Problems*. New York, NY: Springer New York, 2005, pp. 49–114. DOI: 10.1007/0-387-27132-5_3.

[211] A. C. Kak and M. Slaney. “Algorithms for Reconstruction with Nondiffracting Sources”. In: *Principles of Computerized Tomographic Imaging*. 2001, pp. 49–112. DOI: 10.1137/1.9780898719277.ch3.

[212] V. S. S. Kandarpa et al. “LRR-CED: low-resolution reconstruction-aware convolutional encoder–decoder network for direct sparse-view CT image reconstruction”. In: *Physics in Medicine & Biology* 67.15 (2022), p. 155007. DOI: 10.1088/1361-6560/ac7bce.

[213] V. S. S. Kandarpa et al. "DUG-RECON: A Framework for Direct Image Reconstruction Using Convolutional Generative Networks". In: *IEEE Transactions on Radiation and Plasma Medical Sciences* 5.1 (2021), pp. 44–53. DOI: 10.1109/TRPMS.2020.3033172.

[214] V. S. S. Kandarpa. "Tomographic image reconstruction with direct neural network approaches". Theses. Université de Bretagne occidentale - Brest, 2022. URL: https://theses.hal.science/tel-03844434.

[215] E. Kang, J. Min, and J. C. Ye. "A deep convolutional neural network using directional wavelets for low-dose X-ray CT reconstruction". In: *Medical Physics* 44.10 (2017), e360–e375. DOI: https://doi.org/10.1002/mp.12344.

[216] E. Kang et al. "Cycle-consistent adversarial denoising network for multiphase coronary CT angiography". In: *Medical Physics* 46.2 (2019), pp. 550–562. DOI: https://doi.org/10.1002/mp.13284.

[217] D. Karimi and R. Ward. "Interpolation of CT Projections by Exploiting Their Self-Similarity and Smoothness". In: *2021 IEEE International Conference on Image Processing (ICIP)*. 2021, pp. 165–169. DOI: 10.1109/ICIP42928.2021.9506466.

[218] J. H. J. Ketola et al. "Deep learning-based sinogram extension method for interior computed tomography". In: *Medical Imaging 2021: Physics of Medical Imaging*. Ed. by H. Bosmans, W. Zhao, and L. Yu. Vol. 11595. International Society for Optics and Photonics. SPIE, 2021, 115953Q. DOI: 10.1117/12.2580886.

[219] S. Khan et al. "Transformers in Vision: A Survey". In: *ACM Comput. Surv.* 54.10s (2022). ISSN: 0360-0300. DOI: 10.1145/3505244.

[220] K. Kim, S. Soltanayev, and S. Y. Chun. "Unsupervised Training of Denoisers for Low-Dose CT Reconstruction Without Full-Dose Ground Truth". In: *IEEE Journal of Selected Topics in Signal Processing* 14.6 (2020), pp. 1112–1125. DOI: 10.1109/JSTSP.2020.3007326.

[221] D. P. Kingma and J. Ba. "Adam: A Method for Stochastic Optimization". In: *ICLR (Poster)*. 2015. DOI: 10.48550/ARXIV.1412.6980.

[222] D. P. Kingma and M. Welling. "Auto-Encoding Variational Bayes". In: *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*. Ed. by Y. Bengio and Y. LeCun. 2014. DOI: 10.48550/ARXIV.1312.6114.

[223] T. N. Kipf and M. Welling. "Semi-Supervised Classification with Graph Convolutional Networks". In: *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL: https://openreview.net/forum?id=SJU4ayYgl.

[224] T. Knopp and M. Grosser. "Warmstart Approach for Accelerating Deep Image Prior Reconstruction in Dynamic Tomography". In: *Medical Imaging with Deep Learning*. 2022. URL: https://openreview.net/forum?id=aWD0kzMmyD_.

[225] E. Kobler et al. "Total Deep Variation for Linear Inverse Problems". In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2020, pp. 7546–7555. DOI: 10.1109/CVPR42600.2020.00757.

[226] A. Kofler et al. "Neural networks-based regularization for large-scale medical image reconstruction". In: *Physics in Medicine & Biology* 65.13 (2020), p. 135003. DOI: 10.1088/1361-6560/ab990e.

[227] A. Kofler et al. “A U-Nets Cascade for Sparse View Computed Tomography”. In: *Machine Learning for Medical Image Reconstruction*. Ed. by F. Knoll, A. Maier, and D. Rueckert. Cham: Springer International Publishing, 2018, pp. 91–99. ISBN: 978-3-030-00129-2. DOI: 10.1007/978-3-030-00129-2_11.

[228] R. Kress. *Numerical Analysis*. Graduate Texts in Mathematics. Springer New York, 2012. ISBN: 9781461205999. DOI: 10.1007/978-1-4612-0599-9.

[229] A. Krizhevsky, I. Sutskever, and G. E. Hinton. “ImageNet Classification with Deep Convolutional Neural Networks”. In: *Commun. ACM* 60.6 (2017), pp. 84–90. ISSN: 0001-0782. DOI: 10.1145/3065386.

[230] A. Kudo et al. “Virtual Thin Slice: 3D Conditional GAN-based Super-Resolution for CT Slice Interval”. In: *Machine Learning for Medical Image Reconstruction*. Ed. by F. Knoll et al. Cham: Springer International Publishing, 2019, pp. 91–100. ISBN: 978-3-030-33843-5. DOI: 10.1007/978-3-030-33843-5_9.

[231] S. Kyung et al. “MTD-GAN: Multi-task Discriminator Based Generative Adversarial Networks for Low-Dose CT Denoising”. In: *Machine Learning for Medical Image Reconstruction*. Ed. by N. Haq et al. Cham: Springer International Publishing, 2022, pp. 133–144. ISBN: 978-3-031-17247-2. DOI: 10.1007/978-3-031-17247-2_14.

[232] M. J. Lagerwerf et al. “A Computationally Efficient Reconstruction Algorithm for Circular Cone-Beam Computed Tomography Using Shallow Neural Networks”. In: *Journal of Imaging* 6.12 (2020). ISSN: 2313-433X. DOI: 10.3390/jimaging6120135.

[233] H. Lan et al. “Compressed sensing for photoacoustic computed tomography based on an untrained neural network with a shape prior”. In: *Biomed. Opt. Express* 12.12 (2021), pp. 7835–7848. DOI: 10.1364/BOE.441901.

[234] K. Lange and R. Carson. “EM reconstruction algorithms for emission and transmission tomography”. eng. In: *Journal of computer assisted tomography* 8 (2 1984), pp. 306–16. URL: https://www.researchgate.net/publication/279200328.

[235] K. Lange and J. Fessler. “Globally convergent algorithms for maximum a posteriori transmission tomography”. In: *IEEE Transactions on Image Processing* 4.10 (1995), pp. 1430–1438. DOI: 10.1109/83.465107.

[236] K. Lange. “Overview of Bayesian methods in image reconstruction”. In: *Digital Image Synthesis and Inverse Optics*. Ed. by A. F. Gmitro, P. S. Idell, and I. J. LaHaie. Vol. 1351. International Society for Optics and Photonics. SPIE, 1990, pp. 270–287. DOI: 10.1117/12.23640.

[237] G. A. Latham. “Best Tikhonov analogue for Landweber iteration”. In: *Inverse Problems* 14.6 (1998), p. 1527. DOI: 10.1088/0266-5611/14/6/011.

[238] M.-H. Laves, M. Tölle, and T. Ortmaier. “Uncertainty Estimation in Medical Image Denoising with Bayesian Deep Image Prior”. In: *Uncertainty for Safe Utilization of Machine Learning in Medical Imaging, and Graphs in Biomedical Image Analysis*. Ed. by C. H. Sudre et al. Cham: Springer International Publishing, 2020, pp. 81–96. ISBN: 978-3-030-60365-6. DOI: 10.1007/978-3-030-60365-6_9.

[239] M.-H. Laves et al. “Posterior temperature optimized Bayesian models for inverse problems in medical imaging”. In: *Medical Image Analysis* 78 (2022), p. 102382. ISSN: 1361-8415. DOI: https://doi.org/10.1016/j.media.2022.102382.

[240] Y. LeCun et al. “Backpropagation Applied to Handwritten Zip Code Recognition”. In: *Neural Computation* 1.4 (1989), pp. 541–551. DOI: 10.1162/neco.1989.1.4.541.

[241] Y. LeCun and Y. Bengio. "Convolutional Networks for Images, Speech and Time Series". In: *The Handbook of Brain Theory and Neural Networks*. Ed. by M. A. Arbib. The MIT Press, 1995, pp. 255–258. URL: http://www.iro.umontreal.ca/~lisa/pointeurs/handbook-convo.pdf.

[242] Y. Lecun et al. "A tutorial on energy-based learning". English (US). In: *Predicting structured data*. Ed. by G. Bakir et al. MIT Press, 2006. URL: http://yann.lecun.com/exdb/publis/pdf/lecun-06.pdf.

[243] D. Lee, S. Choi, and H.-J. Kim. "High quality imaging from sparsely sampled computed tomography data with deep learning and wavelet transform in various domains". In: *Medical Physics* 46.1 (2019), pp. 104–115. DOI: https://doi.org/10.1002/mp.13258.

[244] H. Lee, J. Lee, and S. Cho. "View-interpolation of sparsely sampled sinogram using convolutional neural network". In: *Medical Imaging 2017: Image Processing*. Ed. by M. A. Styner and E. D. Angelini. Vol. 10133. International Society for Optics and Photonics. SPIE, 2017, p. 1013328. DOI: 10.1117/12.2254244.

[245] H. Lee et al. "Deep-Neural-Network-Based Sinogram Synthesis for Sparse-View CT Image Reconstruction". In: *IEEE Transactions on Radiation and Plasma Medical Sciences* 3.2 (2019), pp. 109–119. DOI: 10.1109/TRPMS.2018.2867611.

[246] H. Lee et al. "Machine Friendly Machine Learning: Interpretation of Computed Tomography Without Image Reconstruction". In: *Scientific Reports* 9.1 (2019), p. 15540. ISSN: 2045-2322. DOI: 10.1038/s41598-019-51779-5.

[247] J. Lee, H. Lee, and S. Cho. "Sinogram synthesis using convolutional-neural-network for sparsely view-sampled CT". In: *Medical Imaging 2018: Image Processing*. Ed. by E. D. Angelini and B. A. Landman. Vol. 10574. International Society for Optics and Photonics. SPIE, 2018, 105742A. DOI: 10.1117/12.2293244.

[248] J. Lehtinen et al. "Noise2Noise: Learning Image Restoration without Clean Data". In: *Proceedings of the 35th International Conference on Machine Learning*. Ed. by J. Dy and A. Krause. Vol. 80. Proceedings of Machine Learning Research. PMLR, 2018, pp. 2965–2974. URL: https://proceedings.mlr.press/v80/lehtinen18a.html.

[249] V. Lempitsky, A. Vedaldi, and D. Ulyanov. "Deep Image Prior". In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2018, pp. 9446–9454. DOI: 10.1109/CVPR.2018.00984.

[250] J. Lesaint. "Data consistency conditions in X-ray transmission imaging and their application to the self-calibration problem." Theses. Université Grenoble Alpes, 2018. URL: https://theses.hal.science/tel-01896806.

[251] J. Leuschner et al. *jleuschn/dival:* version v0.6.1. 2021. DOI: 10.5281/zenodo.4696478.

[252] J. Leuschner et al. "LoDoPaB-CT, a benchmark dataset for low-dose computed tomography reconstruction". In: *Scientific Data* 8.1 (2021), p. 109. ISSN: 2052-4463. DOI: 10.1038/s41597-021-00893-z.

[253] J. Leuschner et al. "Quantitative Comparison of Deep Learning-Based Image Reconstruction Methods for Low-Dose and Sparse-Angle CT Applications". In: *Journal of Imaging* 7.3 (2021). ISSN: 2313-433X. DOI: 10.3390/jimaging7030044.

[254] H. Li et al. "NETT: solving inverse problems with deep neural networks". In: *Inverse Problems* 36.6 (2020), p. 065005. DOI: 10.1088/1361-6420/ab6d57.

[255] Q. Li et al. “Low-dose computed tomography image reconstruction via a multistage convolutional neural network with autoencoder perceptual loss network.” eng. In: *Quantitative imaging in medicine and surgery* 12 (3 2022), pp. 1929–1957. DOI: 10.21037/qims-21-465.

[256] T. Li et al. *Self-Validation: Early Stopping for Single-Instance Deep Generative Priors*. 2021. DOI: 10.48550/ARXIV.2110.12271.

[257] X. Li et al. “Understanding the Disharmony Between Dropout and Batch Normalization by Variance Shift”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2019, Long Beach, CA, USA, June 16-20, 2019*. Computer Vision Foundation / IEEE, 2019, pp. 2682–2690. DOI: 10.1109/CVPR.2019.00279.

[258] Y. Li et al. “Learning to Reconstruct Computed Tomography Images Directly From Sinogram Data Under A Variety of Data Acquisition Conditions”. In: *IEEE Transactions on Medical Imaging* 38.10 (2019), pp. 2469–2481. DOI: 10.1109/TMI.2019.2910760.

[259] Z. Li et al. “SUPER Learning: A Supervised-Unsupervised Framework for Low-Dose CT Image Reconstruction”. In: *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. 2019, pp. 3959–3968. DOI: 10.1109/ICCVW.2019.00490.

[260] Z. Li et al. “Promising Generative Adversarial Network Based Sinogram Inpainting Method for Ultra-Limited-Angle Computed Tomography Imaging”. In: *Sensors* 19.18 (2019). ISSN: 1424-8220. DOI: 10.3390/s19183941.

[261] K. Liang et al. “Improve angular resolution for sparse-view CT with residual convolutional neural network”. In: *Medical Imaging 2018: Physics of Medical Imaging*. Ed. by J. Y. Lo, T. G. Schmidt, and G.-H. Chen. Vol. 10573. International Society for Optics and Photonics. SPIE, 2018, 105731K. DOI: 10.1117/12.2293319.

[262] W.-A. Lin et al. “DuDoNet: Dual Domain Network for CT Metal Artifact Reduction”. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 2019, pp. 10504–10513. DOI: 10.1109/CVPR.2019.01076.

[263] S. Linnainmaa. “Taylor expansion of the accumulated rounding error”. In: *BIT Numerical Mathematics* 16.2 (1976), pp. 146–160. ISSN: 1572-9125. DOI: 10.1007/BF01931367.

[264] D. C. Liu and J. Nocedal. “On the limited memory BFGS method for large scale optimization”. In: *Mathematical Programming* 45.1 (1989), pp. 503–528. ISSN: 1436-4646. DOI: 10.1007/BF01589116.

[265] J. Liu et al. “Image Restoration Using Total Variation Regularized Deep Image Prior”. In: *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2019, pp. 7715–7719. DOI: 10.1109/ICASSP.2019.8682856.

[266] J. Liu et al. “Online Deep Equilibrium Learning for Regularization by Denoising”. In: *Advances in Neural Information Processing Systems*. Ed. by A. H. Oh et al. 2022. URL: https://openreview.net/forum?id=4RC_vI0OgIS.

[267] J. Liu et al. “Stochastic Deep Unfolding for Imaging Inverse Problems”. In: *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2021, pp. 1395–1399. DOI: 10.1109/ICASSP39728.2021.9414332.

[268] T. Loßau. “Machine learning in cardiac CT image reconstruction”. doctoralThesis. Technische Universität Hamburg, 2020. DOI: 10.15480/882.2906.

[269] H. Lu et al. “Noise properties of low-dose CT projections and noise treatment by scale transformations”. In: *2001 IEEE Nuclear Science Symposium Conference Record (Cat. No.01CH37310)*. Vol. 3. 2001, 1662–1666 vol.3. DOI: 10.1109/NSSMIC.2001.1008660.

[270] S. Lunz, O. Öktem, and C.-B. Schönlieb. "Adversarial Regularizers in Inverse Problems". In: *Advances in Neural Information Processing Systems*. Ed. by S. Bengio et al. Vol. 31. Curran Associates, Inc., 2018. URL: https://proceedings.neurips.cc/paper/2018/file/d903e9608cfbf08910611e4346a0ba44-Paper.pdf.

[271] P. Luo et al. "Towards Understanding Regularization in Batch Normalization". In: *International Conference on Learning Representations*. 2019. URL: https://openreview.net/forum?id=HJlLKjR9FQ.

[272] T. Lütjen et al. *Learning-based approaches for reconstructions with inexact operators in nanoCT applications*. 2023. DOI: 10.48550/ARXIV.2307.10474.

[273] Y.-J. Ma et al. "Sinogram denoising via attention residual dense convolutional neural network for low-dose computed tomography". In: *Nuclear Science and Techniques* 32.4 (2021), p. 41. ISSN: 2210-3147. DOI: 10.1007/s41365-021-00874-2.

[274] F. Madesta et al. "Self-contained deep learning-based boosting of 4D cone-beam CT reconstruction". In: *Medical Physics* 47.11 (2020), pp. 5619–5631. DOI: https://doi.org/10.1002/mp.14441.

[275] S. Majee et al. "Multi-Slice Fusion for Sparse-View and Limited-Angle 4D CT Reconstruction". In: *IEEE Transactions on Computational Imaging* 7 (2021), pp. 448–462. DOI: 10.1109/TCI.2021.3074881.

[276] A. Malusek, M. Sandborg, and G. A. Carlsson. "CTmod—A toolkit for Monte Carlo simulation of projections including scatter in computed tomography". In: *Computer Methods and Programs in Biomedicine* 90.2 (2008), pp. 167–178. ISSN: 0169-2607. DOI: https://doi.org/10.1016/j.cmpb.2007.12.005.

[277] S. H. Manglos et al. "Transmission maximum-likelihood reconstruction with ordered subsets for cone beam CT". In: *Physics in Medicine & Biology* 40.7 (1995), p. 1225. DOI: 10.1088/0031-9155/40/7/006.

[278] X. Mao et al. "Least Squares Generative Adversarial Networks". In: *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017, pp. 2813–2821. DOI: 10.1109/ICCV.2017.304.

[279] L. Marcos, J. Alirezaie, and P. Babyn. "Low Dose CT Denoising by ResNet With Fused Attention Modules and Integrated Loss Functions". In: *Frontiers in Signal Processing* 1 (2022). ISSN: 2673-8198. DOI: 10.3389/frsip.2021.812193.

[280] G. Mataev, P. Milanfar, and M. Elad. "DeepRED: Deep Image Prior Powered by RED". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) Workshops*. 2019. URL: https://openaccess.thecvf.com/content_ICCVW_2019/papers/LCI/Mataev_DeepRED_Deep_Image_Prior_Powered_by_RED_ICCVW_2019_paper.pdf.

[281] J. L. McClelland, D. E. Rumelhart, and P. R. Group. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Psychological and Biological Models*. The MIT Press, 1987. ISBN: 9780262291262. DOI: 10.7551/mitpress/5237.001.0001.

[282] W. S. McCulloch and W. Pitts. "A logical calculus of the ideas immanent in nervous activity". In: *The bulletin of mathematical biophysics* 5.4 (1943), pp. 115–133. ISSN: 1522-9602. DOI: 10.1007/BF02478259.

[283] M. Meng et al. "Semi-supervised learned sinogram restoration network for low-dose CT image reconstruction". In: *Medical Imaging 2020: Physics of Medical Imaging*. Ed. by G.-H. Chen and H. Bosmans. Vol. 11312. International Society for Optics and Photonics. SPIE, 2020, 113120B. DOI: 10.1117/12.2548985.

[284] C. A. Metzler et al. *Unsupervised Learning with Stein's Unbiased Risk Estimator*. 2018. DOI: 10.48550/ARXIV.1805.10531.

[285] E. Meyer et al. "Normalized metal artifact reduction (NMAR) in computed tomography". In: *Medical Physics* 37.10 (2010), pp. 5482–5493. DOI: https://doi.org/10.1118/1.3484090.

[286] E. Min et al. "A Survey of Clustering With Deep Learning: From the Perspective of Network Architecture". In: *IEEE Access* 6 (2018), pp. 39501–39514. DOI: 10.1109/ACCESS.2018.2855437.

[287] S. Minaee et al. "Image Segmentation Using Deep Learning: A Survey". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 44.7 (2022), pp. 3523–3542. DOI: 10.1109/TPAMI.2021.3059968.

[288] M. Minsky and S. Papert. "Perceptrons - an introduction to computational geometry". In: *Cambridge tiass., HIT* 479 (1969), p. 480. URL: https://mitpress.mit.edu/9780262630221/perceptrons/.

[289] M. Mirza and S. Osindero. *Conditional Generative Adversarial Nets*. 2014. DOI: 10.48550/ARXIV.1411.1784.

[290] M. W. Mirza, A. Siddiq, and I. R. Khan. "A comparative study of medical image enhancement algorithms and quality assessment metrics on COVID-19 CT images". In: *Signal, Image and Video Processing* 17.4 (2023), pp. 915–924. ISSN: 1863-1711. DOI: 10.1007/s11760-022-02214-2.

[291] T. R. Moen et al. "Low-dose CT image and projection dataset". In: *Medical Physics* 48.2 (2021), pp. 902–911. DOI: https://doi.org/10.1002/mp.14594.

[292] P. Mohammadinejad et al. "CT Noise-Reduction Methods for Lower-Dose Scanning: Strengths and Weaknesses of Iterative Reconstruction Algorithms and New Techniques". In: *RadioGraphics* 41.5 (2021). PMID: 34469209, pp. 1493–1508. DOI: 10.1148/rg.2021200196.

[293] E. H. Moore. "On the reciprocal of the general algebraic matrix". In: *Bull. Am. Math. Soc.* 26 (1920), pp. 394–395.

[294] V. A. Morozov. "On the solution of functional equations by the method of regularization". In: *Doklady Mathematics* 7 (1966), pp. 414–417.

[295] J. N. Morshuis et al. "Adversarial Robustness of MR Image Reconstruction Under Realistic Perturbations". In: *Machine Learning for Medical Image Reconstruction*. Ed. by N. Haq et al. Cham: Springer International Publishing, 2022, pp. 24–33. ISBN: 978-3-031-17247-2.

[296] S. Mukherjee et al. "End-to-end reconstruction meets data-driven regularization for inverse problems". In: *Advances in Neural Information Processing Systems*. Ed. by M. Ranzato et al. Vol. 34. Curran Associates, Inc., 2021, pp. 21413–21425. URL: https://proceedings.neurips.cc/paper/2021/file/b2df0a0d4116c55f81fd5aa1ef876510-Paper.pdf.

[297] S. Mukherjee et al. *Learned convex regularizers for inverse problems*. 2020. DOI: 10.48550/ARXIV.2008.02839.

[298] S. Mukherjee et al. "Learned Reconstruction Methods With Convergence Guarantees: A survey of concepts and applications". In: *IEEE Signal Processing Magazine* 40.1 (2023), pp. 164–182. DOI: 10.1109/MSP.2022.3207451.

[299] M. T. Nair, M. Hegland, and R. S. Anderssen. "The Trade-Off Between Regularity and Stability in Tikhonov Regularization". In: *Mathematics of Computation* 66.217 (1997), pp. 193–206. ISSN: 00255718, 10886842. URL: http://www.jstor.org/stable/2153649 (visited on 11/17/2022).

[300] H. Nakai et al. “Quantitative and Qualitative Evaluation of Convolutional Neural Networks with a Deeper U-Net for Sparse-View Computed Tomography Reconstruction”. In: *Academic Radiology* 27.4 (2020), pp. 563–574. ISSN: 1076-6332. DOI: https://doi.org/10.1016/j.acra.2019.05.016.

[301] Y. Nakamura et al. “Diagnostic value of deep learning reconstruction for radiation dose reduction at abdominal ultra-high-resolution CT”. In: *European Radiology* 31.7 (2021), pp. 4700–4709. ISSN: 1432-1084. DOI: 10.1007/s00330-020-07566-2.

[302] J. G. Nam et al. “Image quality of ultralow-dose chest CT using deep learning techniques: potential superiority of vendor-agnostic post-processing over vendor-specific techniques”. In: *European Radiology* 31.7 (2021), pp. 5139–5147. ISSN: 1432-1084. DOI: 10.1007/s00330-020-07537-7.

[303] M. Nashed. “A new approach to classification and regularization of ill-posed operator equations”. In: *Inverse and ill-posed problems*. Ed. by H. W. Engl and C. Groetsch. Academic Press, 1987, pp. 53–75. ISBN: 978-0-12-239040-1. DOI: 10.1016/B978-0-12-239040-1.50009-0.

[304] F. Natterer. *The Mathematics of Computerized Tomography*. Vieweg+Teubner Verlag, 1986. DOI: 10.1007/978-3-663-01409-6.

[305] P. Netrapalli. “Stochastic Gradient Descent and Its Variants in Machine Learning”. In: *Journal of the Indian Institute of Science* 99.2 (2019), pp. 201–213. ISSN: 0019-4964. DOI: 10.1007/s41745-019-0098-4.

[306] Y. Nievergelt. “Elementary Inversion of Radon’s Transform”. In: *SIAM Review* 28.1 (1986), pp. 79–84. DOI: 10.1137/1028005.

[307] M. Nittscher et al. “SVD-DIP: Overcoming the Overfitting Problem in DIP-based CT Reconstruction”. In: *Medical Imaging with Deep Learning*. 2023. URL: https://openreview.net/forum?id=ivC7VP2mof.

[308] A. Odena, V. Dumoulin, and C. Olah. “Deconvolution and Checkerboard Artifacts”. In: *Distill* (2016). DOI: 10.23915/distill.00003.

[309] K.-S. Oh and K. Jung. “GPU implementation of neural networks”. In: *Pattern Recognition* 37.6 (2004), pp. 1311–1314. ISSN: 0031-3203. DOI: 10.1016/j.patcog.2004.01.013.

[310] A. van den Oord et al. “Conditional Image Generation with PixelCNN Decoders”. In: *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*. Ed. by D. D. Lee et al. 2016, pp. 4790–4798. URL: https://proceedings.neurips.cc/paper/2016/hash/b1301141feffabac455e1f90a7de2054-Abstract.html.

[311] J. Pan et al. “Multi-domain integrative Swin transformer network for sparse-view tomographic reconstruction”. In: *Patterns* 3.6 (2022), p. 100498. DOI: 10.1016/j.patter.2022.100498.

[312] Y. Pang et al. “Image-to-Image Translation: Methods and Applications”. In: *IEEE Transactions on Multimedia* 24 (2022), pp. 3859–3881. DOI: 10.1109/TMM.2021.3109419.

[313] G. Papamakarios et al. “Normalizing Flows for Probabilistic Modeling and Inference”. In: *J. Mach. Learn. Res.* 22 (2021), 57:1–57:64. URL: http://jmlr.org/papers/v22/19-1028.html.

[314] C. Park et al. “CT iterative vs deep learning reconstruction: comparison of noise and sharpness”. In: *European Radiology* 31.5 (2021), pp. 3156–3164. ISSN: 1432-1084. DOI: 10.1007/s00330-020-07358-8.

[315] H. S. Park et al. "Unpaired Image Denoising Using a Generative Adversarial Network in X-Ray CT". In: *IEEE Access* 7 (2019), pp. 110414–110425. DOI: 10.1109/ACCESS.2019.2934178.

[316] D. L. Parker. "Optimal short scan convolution reconstruction for fan beam CT". In: *Medical Physics* 9.2 (1982), pp. 254–257. DOI: https://doi.org/10.1118/1.595078.

[317] D. M. Pelt, K. J. Batenburg, and J. A. Sethian. "Improving Tomographic Reconstruction from Limited Data Using Mixed-Scale Dense Convolutional Neural Networks". In: *Journal of Imaging* 4.11 (2018). ISSN: 2313-433X. DOI: 10.3390/jimaging4110128.

[318] D. M. Pelt et al. "Cycloidal CT with CNN-based sinogram completion and in-scan generation of training data". In: *Scientific Reports* 12.1 (2022), p. 893. ISSN: 2045-2322. DOI: 10.1038/s41598-022-04910-y.

[319] D. M. Pelt and K. J. Batenburg. "Fast Tomographic Reconstruction From Limited Data Using Artificial Neural Networks". In: *IEEE Transactions on Image Processing* 22.12 (2013), pp. 5238–5251. DOI: 10.1109/TIP.2013.2283142.

[320] R. Penrose. "A generalized inverse for matrices". In: *Mathematical Proceedings of the Cambridge Philosophical Society* 51.3 (1955), pp. 406–413. DOI: 10.1017/S0305004100030401.

[321] A. Perelli et al. "Compressive Computed Tomography Reconstruction through Denoising Approximate Message Passing". In: *SIAM Journal on Imaging Sciences* 13.4 (2020), pp. 1860–1897. DOI: 10.1137/19M1310013.

[322] L. Pinheiro Cinelli et al. "Variational Autoencoder". In: *Variational Methods for Machine Learning with Applications to Deep Networks*. Cham: Springer International Publishing, 2021, pp. 111–149. ISBN: 978-3-030-70679-1. DOI: 10.1007/978-3-030-70679-1_5.

[323] N. Qian and T. J. Sejnowski. "Predicting the secondary structure of globular proteins using neural network models". In: *Journal of Molecular Biology* 202.4 (1988), pp. 865–884. ISSN: 0022-2836. DOI: 10.1016/0022-2836(88)90564-5.

[324] A. Radford, L. Metz, and S. Chintala. "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks". In: *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*. Ed. by Y. Bengio and Y. LeCun. 2016. DOI: 10.48550/ARXIV.1511.06434.

[325] J. Radon. "On the determination of functions from their integral values along certain manifolds". In: *IEEE Transactions on Medical Imaging* 5.4 (1986), pp. 170–176. DOI: 10.1109/TMI.1986.4307775.

[326] S. Rautio et al. *Learning a microlocal prior for limited-angle tomography*. 2022. DOI: 10.48550/ARXIV.2201.00656.

[327] J. Redmon and A. Farhadi. "YOLO9000: Better, Faster, Stronger". In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 6517–6525. DOI: 10.1109/CVPR.2017.690.

[328] E. T. Reehorst and P. Schniter. "Regularization by Denoising: Clarifications and New Interpretations". In: *IEEE Transactions on Computational Imaging* 5.1 (2019), pp. 52–67. DOI: 10.1109/TCI.2018.2880326.

[329] P. Ren et al. "A Comprehensive Survey of Neural Architecture Search: Challenges and Solutions". In: *ACM Comput. Surv.* 54.4 (2021). ISSN: 0360-0300. DOI: 10.1145/3447582.

[330] D. J. Rezende and S. Mohamed. "Variational Inference with Normalizing Flows". In: *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*. Ed. by F. R. Bach and D. M. Blei. Vol. 37. JMLR Workshop and Conference Proceedings. JMLR.org, 2015, pp. 1530–1538. URL: http://proceedings.mlr.press/v37/rezende15.html.

[331] P. Rodríguez. "Total Variation Regularization Algorithms for Images Corrupted with Different Noise Models: A Review". In: *JECE* 2013 (2013). ISSN: 2090-0147. DOI: 10.1155/2013/217021.

[332] Y. Romano, M. Elad, and P. Milanfar. "The Little Engine That Could: Regularization by Denoising (RED)". In: *SIAM Journal on Imaging Sciences* 10.4 (2017), pp. 1804–1844. DOI: 10.1137/16M1102884.

[333] O. Ronneberger, P. Fischer, and T. Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Ed. by N. Navab et al. Cham: Springer International Publishing, 2015, pp. 234–241. ISBN: 978-3-319-24574-4. DOI: https://doi.org/10.1007/978-3-319-24574-4_28.

[334] F. Rosenblatt. "The perceptron: a probabilistic model for information storage and organization in the brain." eng. In: *Psychological review* 65 (6 1958), pp. 386–408. DOI: 10.1037/h0042519.

[335] L. I. Rudin, S. Osher, and E. Fatemi. "Nonlinear total variation based noise removal algorithms". In: *Physica D: Nonlinear Phenomena* 60.1 (1992), pp. 259–268. ISSN: 0167-2789. DOI: 10.1016/0167-2789(92)90242-F.

[336] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. "Learning representations by back-propagating errors". In: *Nature* 323.6088 (1986), pp. 533–536. ISSN: 1476-4687. DOI: 10.1038/323533a0.

[337] S. Santurkar et al. "How Does Batch Normalization Help Optimization?" In: *Advances in Neural Information Processing Systems*. Ed. by S. Bengio et al. Vol. 31. Curran Associates, Inc., 2018. URL: https://proceedings.neurips.cc/paper/2018/file/905056c1ac1dad141560467e0a99e1cf-Paper.pdf.

[338] K. Sauer and C. Bouman. "A local update strategy for iterative reconstruction from projections". In: *IEEE Transactions on Signal Processing* 41.2 (1993), pp. 534–548. DOI: 10.1109/78.193196.

[339] J. Schmidhuber. "Deep learning in neural networks: An overview". In: *Neural Networks* 61 (2015), pp. 85–117. ISSN: 0893-6080. DOI: 10.1016/j.neunet.2014.09.003.

[340] M. Schmidt, A. Denker, and J. Leuschner. "The Deep Capsule Prior – advantages through complexity?" In: *PAMM* 21.1 (2021), e202100166. DOI: 10.1002/pamm.202100166.

[341] R. Schulze et al. "Artefacts in CBCT: a review". In: *Dentomaxillofacial Radiology* 40.5 (2011). PMID: 21697151, pp. 265–273. DOI: 10.1259/dmfr/30642039.

[342] S. Schulze, J. Leuschner, and E. J. King. "Blind Source Separation in Polyphonic Music Recordings Using Deep Neural Networks Trained via Policy Gradients". In: *Signals* 2.4 (2021), pp. 637–661. ISSN: 2624-6120. DOI: 10.3390/signals2040039.

[343] J. Schwab, S. Antholzer, and M. Haltmeier. "Deep null space learning for inverse problems: convergence analysis and rates". In: *Inverse Problems* 35.2 (2019), p. 025008. DOI: 10.1088/1361-6420/aaf14a.

[344] H. Shan et al. "Competitive performance of a modularized deep neural network compared to commercial algorithms for low-dose CT image reconstruction". In: *Nature Machine Intelligence* 1.6 (2019), pp. 269–276. ISSN: 2522-5839. DOI: 10.1038/s42256-019-0057-9.

[345] M. A. Shehata et al. "Deep-learning CT reconstruction in clinical scans of the abdomen: a systematic review and meta-analysis". In: *Abdominal Radiology* 48.8 (2023), pp. 2724–2756. ISSN: 2366-0058. DOI: 10.1007/s00261-023-03966-2.

[346] C. Shen, G. Ma, and X. Jia. "Low-dose CT reconstruction assisted by a global CT image manifold prior". In: *15th International Meeting on Fully Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine*. Ed. by S. Matej and S. D. Metzler. Vol. 11072. International Society for Optics and Photonics. SPIE, 2019, p. 1107205. DOI: 10.1117/12.2534959.

[347] Z. Shi et al. "On Measuring and Controlling the Spectral Bias of the Deep Image Prior". In: *International Journal of Computer Vision* 130.4 (2022), pp. 885–908. ISSN: 1573-1405. DOI: 10.1007/s11263-021-01572-7.

[348] P. P. Shinde and S. Shah. "A Review of Machine Learning and Deep Learning Applications". In: *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*. 2018, pp. 1–6. DOI: 10.1109/ICCUBEA.2018.8697857.

[349] Y. Shinya, E. Simo-Serra, and T. Suzuki. "Understanding the Effects of Pre-Training for Object Detectors via Eigenspectrum". In: *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. 2019, pp. 1931–1941. DOI: 10.1109/ICCVW.2019.00242.

[350] C. Shorten and T. M. Khoshgoftaar. "A survey on Image Data Augmentation for Deep Learning". In: *J. Big Data* 6 (2019), p. 60. DOI: 10.1186/s40537-019-0197-0.

[351] N. Siddique et al. "U-Net and Its Variants for Medical Image Segmentation: A Review of Theory and Applications". In: *IEEE Access* 9 (2021), pp. 82031–82057. DOI: 10.1109/ACCESS.2021.3086020.

[352] E. Y. Sidky and X. Pan. "Image reconstruction in circular cone-beam computed tomography by constrained, total-variation minimization". In: *Physics in Medicine & Biology* 53.17 (2008), p. 4777. DOI: 10.1088/0031-9155/53/17/021.

[353] E. Y. Sidky, C.-M. Kao, and X. Pan. "Accurate image reconstruction from few-views and limited-angle data in divergent-beam CT". In: *Journal of X-Ray Science and Technology* 14.2 (2006), pp. 119–139. URL: https://content.iospress.com/articles/journal-of-x-ray-science-and-technology/xst00155.

[354] K. Simonyan and A. Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Y. Bengio and Y. LeCun. 2015. DOI: 10.48550/ARXIV.1409.1556.

[355] V. Sitzmann et al. "Implicit Neural Representations with Periodic Activation Functions". In: *Advances in Neural Information Processing Systems*. Ed. by H. Larochelle et al. Vol. 33. Curran Associates, Inc., 2020, pp. 7462–7473. URL: https://proceedings.neurips.cc/paper_files/paper/2020/file/53c04118df112c13a8c34b38343b9c10-Paper.pdf.

[356] S. L. Smith et al. "On the Origin of Implicit Regularization in Stochastic Gradient Descent". In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=rq_Qr0c1Hyo.

[357] J. Sohl-Dickstein et al. "Deep Unsupervised Learning using Nonequilibrium Thermodynamics". In: *Proceedings of the 32nd International Conference on Machine Learning*. Ed. by F. Bach and D. Blei. Vol. 37. Proceedings of Machine Learning Research. Lille, France: PMLR, 2015, pp. 2256–2265. URL: https://proceedings.mlr.press/v37/sohl-dickstein15.html.

[358] B. Song, L. Shen, and L. Xing. "PINER: Prior-Informed Implicit Neural Representation Learning for Test-Time Adaptation in Sparse-View CT Reconstruction". In: *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*. 2023, pp. 1928–1938. URL: https://openaccess.thecvf.com/content/WACV2023/html/Song_PINER_Prior-Informed_Implicit_Neural_Representation_Learning_for_Test-Time_Adaptation_in_WACV_2023_paper.html.

[359] Y. Song et al. "Score-Based Generative Modeling through Stochastic Differential Equations". In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=PxTIG12RRHS.

[360] Y. Song et al. "Solving Inverse Problems in Medical Imaging with Score-Based Generative Models". In: *International Conference on Learning Representations*. 2022. URL: https://openreview.net/forum?id=vaRCHVj0uGI.

[361] J. T. Springenberg et al. "Striving for Simplicity: The All Convolutional Net". In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Workshop Track Proceedings*. Ed. by Y. Bengio and Y. LeCun. 2015. DOI: 10.48550/ARXIV.1412.6806.

[362] V. Sridhar et al. "Distributed Iterative CT Reconstruction Using Multi-Agent Consensus Equilibrium". In: *IEEE Transactions on Computational Imaging* 6 (2020), pp. 1153–1166. DOI: 10.1109/TCI.2020.3008782.

[363] N. Srivastava et al. "Dropout: A Simple Way to Prevent Neural Networks from Overfitting". In: *Journal of Machine Learning Research* 15.56 (2014), pp. 1929–1958. URL: http://jmlr.org/papers/v15/srivastava14a.html.

[364] C. M. Stein. "Estimation of the Mean of a Multivariate Normal Distribution". In: *The Annals of Statistics* 9.6 (1981), pp. 1135–1151. DOI: 10.1214/aos/1176345632.

[365] A. Steuwe et al. "Influence of a novel deep-learning based reconstruction software on the objective and subjective image quality in low-dose abdominal computed tomography". In: *The British Journal of Radiology* 94.1117 (2021). PMID: 33095654, p. 20200677. DOI: 10.1259/bjr.20200677.

[366] H. Sun et al. "Dynamic PET Image Denoising Using Deep Image Prior Combined With Regularization by Denoising". In: *IEEE Access* 9 (2021), pp. 52378–52392. DOI: 10.1109/ACCESS.2021.3069236.

[367] R. Sun. "Artificial Intelligence: Connectionist and Symbolic Approaches". In: *International Encyclopedia of the Social & Behavioral Sciences*. Ed. by N. J. Smelser and P. B. Baltes. Oxford: Pergamon, 2001, pp. 783–789. ISBN: 978-0-08-043076-8. DOI: 10.1016/B0-08-043076-7/00553-2.

[368] Y. Sun et al. "Singular Value Fine-tuning: Few-shot Segmentation requires Few-parameters Fine-tuning". In: *Advances in Neural Information Processing Systems*. Ed. by A. H. Oh et al. 2022. URL: https://openreview.net/forum?id=LEqYZz7cZOI.

[369] Y. Sun, J. Liu, and U. Kamilov. "Block Coordinate Regularization by Denoising". In: *Advances in Neural Information Processing Systems*. Ed. by H. Wallach et al. Vol. 32. Curran Associates, Inc., 2019. URL: https://proceedings.neurips.cc/paper/2019/file/9872ed9fc22fc182d371c3e9ed316094-Paper.pdf.

[370] Y. Sun et al. "Async-{RED}: A Provably Convergent Asynchronous Block Parallel Stochastic Method using Deep Denoising Priors". In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=9EsrXMzlFQY.

[371] Y. Sun et al. "CoIL: Coordinate-Based Internal Learning for Tomographic Imaging". In: *IEEE Transactions on Computational Imaging* 7 (2021), pp. 1400–1412. DOI: 10.1109/TCI.2021.3125564.

[372] Z. Sun et al. "A Plug-and-Play Deep Image Prior". In: *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2021, pp. 8103–8107. DOI: 10.1109/ICASSP39728.2021.9414879.

[373] T. P. Szczykutowicz et al. "A Review of Deep Learning CT Reconstruction: Concepts, Limitations, and Promise in Clinical Practice". In: *Current Radiology Reports* 10.9 (2022), pp. 101–115. ISSN: 2167-4825. DOI: 10.1007/s40134-022-00399-5.

[374] C. Szegedy et al. "Going deeper with convolutions". In: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015, pp. 1–9. DOI: 10.1109/CVPR.2015.7298594.

[375] C. Szegedy et al. "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 31.1 (2017). DOI: 10.1609/aaai.v31i1.11231.

[376] C. Tang et al. "Unpaired Low-Dose CT Denoising Network Based on Cycle-Consistent Generative Adversarial Network with Prior Image Information". In: *Computational and Mathematical Methods in Medicine* 2019 (2019), p. 8639825. ISSN: 1748-670X. DOI: 10.1155/2019/8639825.

[377] Y. Tang et al. "A primal dual proximal point method of Chambolle-Pock algorithms for ℓ1-TV minimization problems in image reconstruction". In: *2012 5th International Conference on BioMedical Engineering and Informatics*. 2012, pp. 12–16. DOI: 10.1109/BMEI.2012.6513092.

[378] F. Thaler et al. "Sparse-View CT Reconstruction Using Wasserstein GANs". In: *Machine Learning for Medical Image Reconstruction*. Ed. by F. Knoll, A. Maier, and D. Rueckert. Cham: Springer International Publishing, 2018, pp. 75–82. ISBN: 978-3-030-00129-2. DOI: 10.1007/978-3-030-00129-2_9.

[379] N. Thanh Trung et al. "Dilated Residual Convolutional Neural Networks for Low-Dose CT Image Denoising". In: *2020 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)*. 2020, pp. 189–192. DOI: 10.1109/APCCAS50809.2020.9301693.

[380] J.-B. Thibault et al. "A three-dimensional statistical approach to improved image quality for multislice helical CT". In: *Medical Physics* 34.11 (2007), pp. 4526–4544. DOI: https://doi.org/10.1118/1.2789499.

[381] Z. Tian et al. "Low-dose CT reconstruction via edge-preserving total variation regularization". In: *Physics in Medicine & Biology* 56.18 (2011), p. 5949. DOI: 10.1088/0031-9155/56/18/011.

[382] M. Tölle, M.-H. Laves, and A. Schlaefer. "A Mean-Field Variational Inference Approach to Deep Image Prior for Inverse Problems in Medical Imaging". In: *Proceedings of the Fourth Conference on Medical Imaging with Deep Learning*. Ed. by M. Heinrich et al. Vol. 143. Proceedings of Machine Learning Research. PMLR, 2021, pp. 745–760. URL: https://proceedings.mlr.press/v143/tolle21a.html.

[383] J. M. Tomczak. *Deep Generative Modeling*. Springer, 2022. ISBN: 978-3-030-93157-5. DOI: 10.1007/978-3-030-93158-2.

[384] Y. Y. Tseng. "Generalized inverses of unbounded operators between two unitary spaces". In: *Dokl. Akad. Nauk. SSSR*. Vol. 67. 1949, pp. 431–434.

[385] D. Van Veen et al. *Compressed Sensing with Deep Image Prior and Learned Regularization*. 2018. DOI: 10.48550/ARXIV.1806.06438.

[386] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg. "Plug-and-Play priors for model based reconstruction". In: *2013 IEEE Global Conference on Signal and Information Processing*. 2013, pp. 945–948. DOI: 10.1109/GlobalSIP.2013.6737048.

[387] F. Wagner et al. "On the Benefit of Dual-Domain Denoising in a Self-Supervised Low-Dose CT Setting". In: *2023 IEEE 20th International Symposium on Biomedical Imaging (ISBI)*. 2023, pp. 1–5. DOI: 10.1109/ISBI53787.2023.10230511.

[388] C. Wang et al. "DuDoTrans: Dual-Domain Transformer for Sparse-View CT Reconstruction". In: *Machine Learning for Medical Image Reconstruction*. Ed. by N. Haq et al. Cham: Springer International Publishing, 2022, pp. 84–94. ISBN: 978-3-031-17247-2. DOI: 10.1007/978-3-031-17247-2_9.

[389] G. Wang, J. C. Ye, and B. De Man. "Deep learning for tomographic image reconstruction". In: *Nature Machine Intelligence* 2.12 (2020), pp. 737–748. ISSN: 2522-5839. DOI: 10.1038/s42256-020-00273-z.

[390] G. Wang et al. "Reference-driven undersampled MRI reconstruction using automated stopping deep image prior". In: *Fourteenth International Conference on Digital Image Processing (ICDIP 2022)*. Ed. by X. Jiang et al. Vol. 12342. International Society for Optics and Photonics. SPIE, 2022, p. 1234219. DOI: 10.1117/12.2644282.

[391] H. Wang et al. "Early Stopping for Deep Image Prior". In: *Transactions on Machine Learning Research* (2023). Under review. ISSN: 2835-8856. URL: https://openreview.net/forum?id=231ZzrLC8X.

[392] J. Wang et al. "Deep learning based image reconstruction algorithm for limited-angle translational computed tomography". In: *PLOS ONE* 15.1 (2020), pp. 1–20. DOI: 10.1371/journal.pone.0226963.

[393] Q. Wang et al. "A Comprehensive Survey of Loss Functions in Machine Learning". In: *Annals of Data Science* 9.2 (2022), pp. 187–212. ISSN: 2198-5812. DOI: 10.1007/s40745-020-00253-5.

[394] S. Wang, X. Li, and P. Chen. "ADMM-SVNet: An ADMM-Based Sparse-View CT Reconstruction Network". In: *Photonics* 9.3 (2022). ISSN: 2304-6732. DOI: 10.3390/photonics9030186.

[395] T. Wang et al. "A review on medical imaging synthesis using deep learning and its clinical applications". In: *Journal of Applied Clinical Medical Physics* 22.1 (2021), pp. 11–36. DOI: https://doi.org/10.1002/acm2.13121.

[396] K. Wei et al. “TFPnP: Tuning-free Plug-and-Play Proximal Algorithms with Applications to Inverse Imaging Problems”. In: *Journal of Machine Learning Research* 23.16 (2022), pp. 1–48. URL: http://jmlr.org/papers/v23/20-1297.html.

[397] M. Welling and Y. W. Teh. “Bayesian Learning via Stochastic Gradient Langevin Dynamics”. In: *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*. Ed. by L. Getoor and T. Scheffer. Omnipress, 2011, pp. 681–688. URL: https://icml.cc/2011/papers/398%5C_icmlpaper.pdf.

[398] S. Wesarg, M. Ebert, and T. Bortfeld. “Parker weights revisited”. In: *Medical Physics* 29.3 (2002), pp. 372–378. DOI: https://doi.org/10.1118/1.1450132.

[399] B. Widrow and M. E. Hoff. “Adaptive Switching Circuits”. In: *1960 IRE WESCON Convention Record, Part 4*. New York: IRE, 1960, pp. 96–104. URL: https://www-isl.stanford.edu/~widrow/papers/c1960adaptiveswitching.pdf.

[400] C. Winkler et al. *Learning Likelihoods with Conditional Normalizing Flows*. 2019. DOI: 10.48550/ARXIV.1912.00042.

[401] P. J. Withers et al. “X-ray computed tomography”. In: *Nature Reviews Methods Primers* 1.1 (2021), p. 18. ISSN: 2662-8449. DOI: 10.1038/s43586-021-00015-4.

[402] D. H. Wolpert. “The Lack of A Priori Distinctions Between Learning Algorithms”. In: *Neural Computation* 8.7 (1996), pp. 1341–1390. ISSN: 0899-7667. DOI: 10.1162/neco.1996.8.7.1341.

[403] J. M. Wolterink et al. “Generative Adversarial Networks for Noise Reduction in Low-Dose CT”. In: *IEEE Transactions on Medical Imaging* 36.12 (2017), pp. 2536–2545. DOI: 10.1109/TMI.2017.2708987.

[404] D. Wu, K. Kim, and Q. Li. “Computationally efficient deep neural network for computed tomography image reconstruction”. In: *Medical Physics* 46.11 (2019), pp. 4763–4776. DOI: https://doi.org/10.1002/mp.13627.

[405] D. Wu, K. Kim, and Q. Li. “Low-dose CT reconstruction with Noise2Noise network and testing-time fine-tuning”. In: *Medical Physics* 48.12 (2021), pp. 7657–7672. DOI: https://doi.org/10.1002/mp.15101.

[406] D. Wu et al. “Consensus Neural Network for Medical Imaging Denoising with Only Noisy Training Samples”. In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*. Ed. by D. Shen et al. Cham: Springer International Publishing, 2019, pp. 741–749. ISBN: 978-3-030-32251-9.

[407] W. Wu et al. “DRONE: Dual-Domain Residual-based Optimization NEtwork for Sparse-View CT Reconstruction”. In: *IEEE Transactions on Medical Imaging* 40.11 (2021), pp. 3002–3014. DOI: 10.1109/TMI.2021.3078067.

[408] W. Wu et al. “Stabilizing deep tomographic reconstruction: Part A. Hybrid framework and experimental results”. In: *Patterns* 3.5 (2022), p. 100474. ISSN: 2666-3899. DOI: https://doi.org/10.1016/j.patter.2022.100474.

[409] W. Wu et al. “Stabilizing deep tomographic reconstruction: Part B. Convergence analysis and adversarial attacks”. In: *Patterns* 3.5 (2022), p. 100475. ISSN: 2666-3899. DOI: https://doi.org/10.1016/j.patter.2022.100475.

[410] X. Wu et al. “PGNet: Projection generative network for sparse-view reconstruction of projection-based magnetic particle imaging”. In: *Med Phys*. n/a.n/a (2022). ISSN: 0094-2405. DOI: 10.1002/mp.16048.

[411] Y. Wu and K. He. "Group Normalization". In: *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XIII*. Ed. by V. Ferrari et al. Vol. 11217. Lecture Notes in Computer Science. Springer, 2018, pp. 3–19. DOI: 10.1007/978-3-030-01261-8_1.

[412] T. Würfl et al. "Deep Learning Computed Tomography". In: *Medical Image Computing and Computer-Assisted Intervention - MICCAI 2016*. Ed. by S. Ourselin et al. Cham: Springer International Publishing, 2016, pp. 432–440. ISBN: 978-3-319-46726-9. DOI: 10.1007/978-3-319-46726-9_50.

[413] T. Würfl et al. "Deep Learning Computed Tomography: Learning Projection-Domain Weights From Image Domain in Limited Angle Problems". In: *IEEE Transactions on Medical Imaging* 37.6 (2018), pp. 1454–1463. DOI: 10.1109/TMI.2018.2833499.

[414] M. Xie et al. "Joint Reconstruction and Calibration Using Regularization by Denoising with Application to Computed Tomography". In: *2021 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. 2021, pp. 4011–4020. DOI: 10.1109/ICCVW54120.2021.00448.

[415] S. Xie et al. "Artifact Removal using Improved GoogLeNet for Sparse-view CT Reconstruction". In: *Scientific Reports* 8.1 (2018), p. 6700. ISSN: 2045-2322. DOI: 10.1038/s41598-018-25153-w.

[416] F. Xu et al. "On the efficiency of iterative ordered subset reconstruction algorithms for acceleration on GPUs". In: *Computer Methods and Programs in Biomedicine* 98.3 (2010). HP-MICCAI 2008, pp. 261–270. ISSN: 0169-2607. DOI: https://doi.org/10.1016/j.cmpb.2009.09.003.

[417] W. Xu and K. Mueller. "Evaluating popular non-linear image processing filters for their use in regularized iterative CT". In: *IEEE Nuclear Science Symposuim & Medical Imaging Conference*. 2010, pp. 2864–2865. DOI: 10.1109/NSSMIC.2010.5874318.

[418] Y. Xu and W. Yin. "A Block Coordinate Descent Method for Regularized Multiconvex Optimization with Applications to Nonnegative Tensor Factorization and Completion". In: *SIAM Journal on Imaging Sciences* 6.3 (2013), pp. 1758–1789. DOI: 10.1137/120887795.

[419] L. Yang et al. "Low-Dose CT Denoising via Sinogram Inner-Structure Transformer". In: *IEEE Transactions on Medical Imaging* (2022), pp. 1–1. DOI: 10.1109/TMI.2022.3219856.

[420] Q. Yang et al. "Low-Dose CT Image Denoising Using a Generative Adversarial Network With Wasserstein Distance and Perceptual Loss". In: *IEEE Transactions on Medical Imaging* 37.6 (2018), pp. 1348–1357. DOI: 10.1109/TMI.2018.2827462.

[421] X. Yang et al. "A Survey on Deep Semi-Supervised Learning". In: *IEEE Transactions on Knowledge and Data Engineering* (2022), pp. 1–20. DOI: 10.1109/TKDE.2022.3220219.

[422] Y. Yao, L. Rosasco, and A. Caponnetto. "On Early Stopping in Gradient Descent Learning". In: *Constructive Approximation* 26.2 (2007), pp. 289–315. ISSN: 1432-0940. DOI: 10.1007/s00365-006-0663-2.

[423] D. H. Ye et al. "DEEP BACK PROJECTION FOR SPARSE-VIEW CT RECONSTRUCTION". In: *2018 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*. 2018, pp. 1–5. DOI: 10.1109/GlobalSIP.2018.8646669.

[424] D. H. Ye et al. "Deep Residual Learning for Model-Based Iterative CT Reconstruction Using Plug-and-Play Framework". In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2018, pp. 6668–6672. DOI: 10.1109/ICASSP.2018.8461408.

[425] J. C. Ye, Y. Han, and E. Cha. "Deep Convolutional Framelets: A General Deep Learning Framework for Inverse Problems". In: *SIAM Journal on Imaging Sciences* 11.2 (2018), pp. 991–1048. DOI: 10.1137/17M1141771.

[426] X. Ye et al. "Low-Dose CT Reconstruction via Dual-Domain Learning and Controllable Modulation". In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2022*. Ed. by L. Wang et al. Cham: Springer Nature Switzerland, 2022, pp. 549–559. ISBN: 978-3-031-16446-0. DOI: 10.1007/978-3-031-16446-0_52.

[427] Z. Yin et al. "Unpaired Image Denoising via Wasserstein GAN in Low-Dose CT Image with Multi-Perceptual Loss and Fidelity Loss". In: *Symmetry* 13.1 (2021). ISSN: 2073-8994. DOI: 10.3390/sym13010126.

[428] C. You et al. "Structurally-Sensitive Multi-Scale Deep Neural Network for Low-Dose CT Denoising". In: *IEEE Access* 6 (2018), pp. 41839–41855. DOI: 10.1109/ACCESS.2018.2858196.

[429] L. Yu et al. "Deep Sinogram Completion With Image Prior for Metal Artifact Reduction in CT Images". In: *IEEE Transactions on Medical Imaging* 40.1 (2021), pp. 228–238. DOI: 10.1109/TMI.2020.3025064.

[430] H. Yuan, J. Jia, and Z. Zhu. "SIPID: A deep learning framework for sinogram interpolation and image denoising in low-dose CT reconstruction". In: *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*. 2018, pp. 1521–1524. DOI: 10.1109/ISBI.2018.8363862.

[431] N. Yuan, J. Zhou, and J. Qi. "Half2Half: deep neural network based CT image denoising without independent reference data". In: *Physics in Medicine & Biology* 65.21 (2020), p. 215020. DOI: 10.1088/1361-6560/aba939.

[432] G. Zang et al. "IntraTomo: Self-supervised Learning-based Tomography via Sinogram Synthesis and Prediction". In: *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*. 2021, pp. 1940–1950. DOI: 10.1109/ICCV48922.2021.00197.

[433] R. Zha, Y. Zhang, and H. Li. "NAF: Neural Attenuation Fields for Sparse-View CBCT Reconstruction". In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2022*. Ed. by L. Wang et al. Cham: Springer Nature Switzerland, 2022, pp. 442–452. ISBN: 978-3-031-16446-0. DOI: 10.1007/978-3-031-16446-0_42.

[434] H.-M. Zhang and B. Dong. "A Review on Deep Learning in Medical Image Reconstruction". In: *Journal of the Operations Research Society of China* 8.2 (2020), pp. 311–340. ISSN: 2194-6698. DOI: 10.1007/s40305-019-00287-4.

[435] H. Zhang, B. Dong, and B. Liu. "JSR-Net: A Deep Network for Joint Spatial-radon Domain CT Reconstruction from Incomplete Data". In: *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2019, pp. 3657–3661. DOI: 10.1109/ICASSP.2019.8682178.

[436] H. Zhang and J.-J. Sonke. "Directional sinogram interpolation for sparse angular acquisition in cone-beam computed tomography". In: *Journal of X-Ray Science and Technology* 21.4 (2013), pp. 481–496. DOI: 10.3233/XST-130401.

[437] M. Zhang, S. Gu, and Y. Shi. "The use of deep learning methods in low-dose computed tomography image reconstruction: a systematic review". In: *Complex & Intelligent Systems* 8.6 (2022), pp. 5545–5561. ISSN: 2198-6053. DOI: 10.1007/s40747-022-00724-7.

[438] Q. Zhang et al. "Learning a Dilated Residual Network for SAR Image Despeckling". In: *Remote Sensing* 10.2 (2018). ISSN: 2072-4292. DOI: 10.3390/rs10020196.

[439] W. Zhang et al. “Computerized detection of clustered microcalcifications in digital mammograms using a shift-invariant artificial neural network”. In: *Medical Physics* 21.4 (1994), pp. 517–524. DOI: 10.1118/1.597177.

[440] W. Zhang et al. “Image processing of human corneal endothelium based on a learning network”. In: *Appl. Opt.* 30.29 (1991), pp. 4211–4217. DOI: 10.1364/AO.30.004211.

[441] W. Zhang et al. “Parallel distributed processing model with local space-invariant interconnections and its optical architecture”. In: *Appl. Opt.* 29.32 (1990), pp. 4790–4797. DOI: 10.1364/AO.29.004790.

[442] Y. Zhang and H. Yu. “Convolutional Neural Network Based Metal Artifact Reduction in X-Ray Computed Tomography”. In: *IEEE Transactions on Medical Imaging* 37.6 (2018), pp. 1370–1381. DOI: 10.1109/TMI.2018.2823083.

[443] Y. Zhang et al. “LEARN++: Recurrent Dual-Domain Reconstruction Network for Compressed Sensing CT”. In: *IEEE Transactions on Radiation and Plasma Medical Sciences* (2022), pp. 1–1. DOI: 10.1109/TRPMS.2022.3222213.

[444] Z. Zhang et al. “A Sparse-View CT Reconstruction Method Based on Combination of DenseNet and Deconvolution”. In: *IEEE Transactions on Medical Imaging* 37.6 (2018), pp. 1407–1417. DOI: 10.1109/TMI.2018.2823338.

[445] Z. Zhang et al. “TransCT: Dual-Path Transformer for Low Dose Computed Tomography”. In: *Medical Image Computing and Computer Assisted Intervention – MICCAI 2021*. Ed. by M. de Bruijne et al. Cham: Springer International Publishing, 2021, pp. 55–64. ISBN: 978-3-030-87231-1.

[446] J. Zhao et al. “Few-View CT reconstruction method based on deep learning”. In: *2016 IEEE Nuclear Science Symposium, Medical Imaging Conference and Room-Temperature Semiconductor Detector Workshop (NSS/MIC/RTSD)*. 2016, pp. 1–4. DOI: 10.1109/NSSMIC.2016.8069593.

[447] A. Zheng et al. “A dual-domain deep learning-based reconstruction method for fully 3D sparse data helical CT”. In: *Physics in Medicine & Biology* 65.24 (2020), p. 245030. DOI: 10.1088/1361-6560/ab8fc1.

[448] S. Zhi et al. “Artifacts reduction method for phase-resolved Cone-Beam CT (CBCT) images via a prior-guided CNN”. In: *Medical Imaging 2019: Physics of Medical Imaging*. Ed. by T. G. Schmidt, G.-H. Chen, and H. Bosmans. Vol. 10948. International Society for Optics and Photonics. SPIE, 2019, p. 1094828. DOI: 10.1117/12.2513128.

[449] K. C. Zhou and R. Horstmeyer. “Diffraction tomography with a deep image prior”. In: *Opt. Express* 28.9 (2020), pp. 12872–12896. DOI: 10.1364/OE.379200.

[450] J.-Y. Zhu et al. “Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks”. In: *2017 IEEE International Conference on Computer Vision (ICCV)*. 2017, pp. 2242–2251. DOI: 10.1109/ICCV.2017.244.

[451] L. Zhu et al. “Completion of Metal-Damaged Traces Based on Deep Learning in Sinogram Domain for Metal Artifacts Reduction in CT Images”. In: *Sensors* 21.24 (2021). ISSN: 1424-8220. DOI: 10.3390/s21248164.

[452] L. Zhu et al. “Metal Artifact Reduction for X-Ray Computed Tomography Using U-Net in Image Domain”. In: *IEEE Access* 7 (2019), pp. 98743–98754. DOI: 10.1109/ACCESS.2019.2930302.

[453] E. A. Zwanenburg, M. A. Williams, and J. M. Warnett. “Review of high-speed imaging with lab-based x-ray computed tomography”. In: *Measurement Science and Technology* 33.1 (2021), p. 012003. DOI: 10.1088/1361-6501/ac354a.

Part II

Papers

Table of Contents

Paper 1

LoDoPaB-CT, a benchmark dataset for low-dose computed tomography reconstruction

www.nature.com/scientificdata

SCIENTIFIC DATA

Check for updates

OPEN

DATA DESCRIPTOR

LoDoPaB-CT, a benchmark dataset for low-dose computed tomography reconstruction

Johannes Leuschner[1,2]✉, Maximilian Schmidt[1,2]✉, Daniel Otero Baguer[1] & Peter Maass[1]

Deep learning approaches for tomographic image reconstruction have become very effective and have been demonstrated to be competitive in the field. Comparing these approaches is a challenging task as they rely to a great extent on the data and setup used for training. With the Low-Dose Parallel Beam (LoDoPaB)-CT dataset, we provide a comprehensive, open-access database of computed tomography images and simulated low photon count measurements. It is suitable for training and comparing deep learning methods as well as classical reconstruction approaches. The dataset contains over 40000 scan slices from around 800 patients selected from the LIDC/IDRI database. The data selection and simulation setup are described in detail, and the generating script is publicly accessible. In addition, we provide a Python library for simplified access to the dataset and an online reconstruction challenge. Furthermore, the dataset can also be used for transfer learning as well as sparse and limited-angle reconstruction scenarios.

Background & Summary

Tomographic image reconstruction is an extensively studied field. One popular imaging modality in clinical and industrial applications is computed tomography (CT). It allows for the non-invasive acquisition of the inside of an object or the human body. The measurements are based on the attenuation of X-ray beams. To obtain the internal distribution of the body from these measurements, an inverse problem must be solved. Traditionally, analytical methods, like filtered back-projection (FBP) or iterative reconstruction (IR) techniques, are used for this task. These methods are the gold standard in the presence of enough high-dose/low-noise measurements. However, as high doses of applied radiation are potentially harmful to the patients, modern scanners aim at reducing the radiation dose. There exist several strategies, but all introduce specific challenges for the reconstruction algorithm, e.g. undersampling or increased noise levels, which require more sophisticated reconstruction methods. The higher the noise or undersampling, the more prior knowledge about the target reconstructions is needed to improve the final quality[1]. Analytical methods are only able to use very limited prior information. Alternatively, machine learning approaches are able to learn underlying distributions and typical image features, which constitute a much larger and flexible prior. Recent image reconstruction approaches involving machine learning, in particular deep learning (DL), have been developed and demonstrated to be very competitive[2–8].

DL-based approaches benefit strongly from the availability of comprehensive datasets. In the last years, a wide variety of CT data has been published, covering different body parts and scan scenarios. For the training of reconstruction models, the projections (measured data) are crucial but are rarely made available. Recently, Low Dose CT Image and Projection Data (LDCT-and-Projection-data)[9] was published by investigators from the Mayo Clinic, which include measured normal-dose projection data of 299 patients in the new open DICOM-CT-PD format. The AAPM Low Dose CT Grand Challenge data[10] includes simulated measurements, featuring 30 different patients. The Finish Inverse Problems Society (FIPS) provides multiple measurements of a walnut[11] and a lotus root[12] aimed at sparse data tomography. Recently, Der Sarkissian *et al.*[13] published cone-beam CT projection data and reconstructions of 42 walnuts. Their dataset is directly aimed at the training and comparison of machine learning methods. In magnetic resonance imaging, fastMRI[14] with 1600 scans of humans knees is another prominent example.

[1]Center for Industrial Mathematics, University of Bremen, Bibliothekstr. 5, 28359, Bremen, Germany. [2]These authors contributed equally: Johannes Leuschner, Maximilian Schmidt. ✉e-mail: jleuschn@uni-bremen.de; maximilian.schmidt@uni-bremen.de

Other CT datasets focus on the detection and segmentation of special structures like lesions in the reconstructions for the development of computer-aided diagnostic (CAD) methods[15–20]. Therefore, they do not include the projection data. The LIDC/IDRI database[15], which we use for the ground truth of our dataset (cf. section "Methods"), targets lung nodule detection. FUMPE[16] contains CT angiography images of 35 subjects for the detection of pulmonary embolisms. KiTS2019[17] is built around the segmentation of kidney tumours in CT images. The Japanese Society of Radiology Technology (JSRT) database[18] and the National Lung Screening Trial (NLST) in cooperation with the CT Image Library (CTIL)[19,20] each contain scans of the lung. These datasets can also be used for the investigation of reconstruction methods by simulating the missing measurements.

Different learned methods have been successfully applied to the task of low-dose reconstruction[7]. However, comparing these approaches is a challenging task since they highly rely on the data and the setup that is used for training. The main goal of this work is to provide a standard dataset that can be used to train and benchmark learned low-dose CT reconstruction methods. To this end, we introduce the Low-Dose Parallel Beam (LoDoPaB)-CT dataset, which uses the public LIDC/IDRI database[15,21,22] of human chest CT reconstructions. We consider these, in the form of 2D images, to be the so-called ground truth. The projections are created by simulating low photon count CT measurements with a parallel beam scan geometry. Due to the slice-based 2D setup, each of the generated measurements corresponds directly to a ground truth slice. Thus, the reconstruction process can be carried out slice-wise without rebinning[23], which would have to be applied to the measurements for 3D helical cone-beam geometries commonly used in modern scanners[9] to allow for the slice-wise use of a 2D reconstruction algorithm. In order to generalise from our dataset to the clinical 3D setup, the effect of rebinning needs to be evaluated. Also, learned algorithms directly targeted at 3D reconstruction should be considered in this case, which at the moment are barely computationally feasible[24], but presumably outperform 2D reconstruction algorithms applied to rebinned measurements. Despite the generalisation to the 3D case not being straight-forward, our dataset allows to train and compare a large number of approaches applicable to the 2D scenario, which we expect to yield insights for the design of 3D algorithms as well.

Paired samples constitute the most complete training data and could be used for all kinds of learning. In particular, methods that require independent samples from the distributions of images and measurements, or only from one of these distributions, can still make use of the dataset. In total, the dataset features more than 40000 sample pairs from over 800 different patients. This amount of data and variability can be necessary to successfully train deep neural networks[25]. It also qualifies the dataset for transfer learning. In addition, the included measurements can be easily modified for sparse and limited angle scan scenarios.

Methods

In this section, the considered mathematical model of CT is stated first, followed by a detailed description of the dataset generation. This starts with the LIDC/IDRI database[15], from which we extract the ground truth reconstructions. Finally, the data processing steps are described, which are also summarised in a semi-formal manner at the end of the section. As a technical reference, the script[26] used for generation is available online (https://github.com/jleuschn/lodopab_tech_ref).

Parallel beam CT model. We consider the inverse problem of computed tomography given by

$$\mathcal{A}x + \varepsilon(\mathcal{A}x) = y^\delta \tag{1}$$

with:

- $\mathcal{A}$ the linear ray transform defined by the scan geometry,
- x the unknown interior distribution of the X-ray attenuation coefficient in the body, also called image,
- ε a sample from a noise distribution that may depend on the ideal measurement $\mathcal{A}x$,
- y^δ the noisy CT measurement, also called projections or sinogram.

More specifically, we choose a two-dimensional parallel beam geometry, for which the ray transform $\mathcal{A}$ is the Radon transform[27]. It integrates the values of $x{:}\mathbb{R}^2 \to \mathbb{R}$ fulfilling some regularity conditions (cf. Radon[27]) along the X-ray lines

$$L_{s,\varphi}(t) := s\omega(\varphi) + t\omega^\perp(\varphi), \quad \omega(\varphi) := \begin{bmatrix}\cos(\varphi)\\ \sin(\varphi)\end{bmatrix}, \quad \omega^\perp(\varphi) := \begin{bmatrix}-\sin(\varphi)\\ \cos(\varphi)\end{bmatrix}, \tag{2}$$

for all parameters $s \in \mathbb{R}$ and $\varphi \in [0, \pi)$, which denote the distance from the origin and the angle, respectively (cf. Figure 1). In mathematical terms, the image is transformed into a function of (s, φ),

$$\mathcal{A}x(s, \varphi) := \int_{\mathbb{R}} x(L_{s,\varphi}(t))\,\mathrm{d}t, \tag{3}$$

which is called projection, since for each fixed angle φ the 2D image x is projected onto a line parameterised by s, namely the detector. Visualisations of projections as images themselves are called sinograms (cf. Figure 2). The projection relates to the ideal intensity measurements $I_1(s, \varphi)$ at the detector according to Beer-Lambert's law by

$$\mathcal{A}(s, \varphi) = -\ln\left(\frac{I_1(s, \varphi)}{I_0}\right) = y(s, \varphi), \tag{4}$$

where I_0 is the intensity of an unattenuated beam.

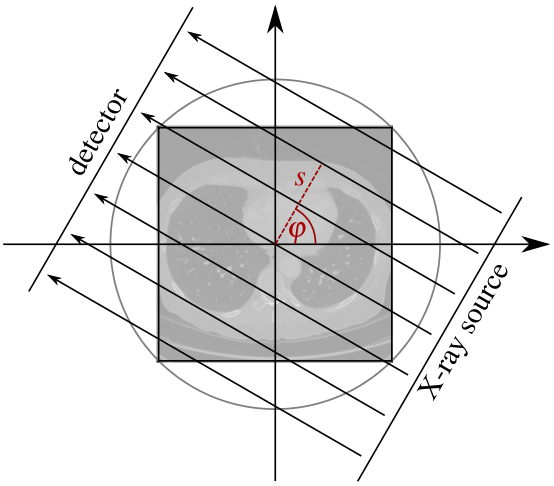

Fig. 1 Visualisation[25] of the parallel beam geometry.

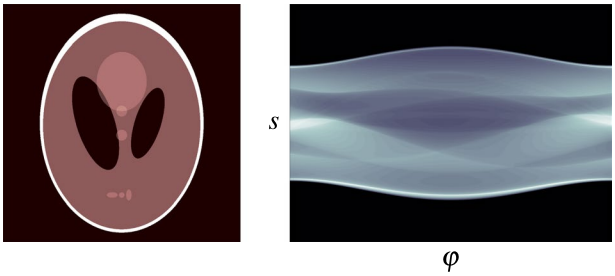

Fig. 2 The Shepp-Logan phantom (left) and its corresponding sinogram (right).

In practice, the measured intensities are noisy. The noise can be classified into *quantum noise* and *detector noise*. Quantum noise stems from the process of photon generation, attenuation and detection, which as a whole can be modelled by a Poisson distribution[28]. The detector noise stems from the electronic data acquisition system and is usually assumed to be Gaussian. It would play an important role in ultra-low-dose CT with very small numbers of detected photons[29] but is neglected in our case. Thus we model the number of detected photons and, by this, the measured intensity ratio with

$$\widetilde{N}_1(s,\varphi) \sim \mathrm{Pois}(N_0 \exp(-\mathscr{A}x(s,\varphi))), \qquad \frac{\widetilde{I}_1(s,\varphi)}{I_0} = \frac{\widetilde{N}_1(s,\varphi)}{N_0}, \tag{5}$$

where N_0 is the mean photon count without attenuation and $\mathrm{Pois}(\lambda)$ denotes the probability distribution defined by

$$P(X=k) = \frac{\lambda^k e^{-\lambda}}{k!}, \quad k \in \mathbb{N}_0. \tag{6}$$

For practical application, the model needs to be discretised. The forward operator is then a finite-dimensional linear map $A : \mathbb{R}^n \to \mathbb{R}^m$, where n is the number of image pixels and m is the product of the number of detector pixels and the number of angles for which measurements are obtained. The discrete model reads

$$Ax + \varepsilon(Ax) = y^\delta, \quad \varepsilon(Ax) = -Ax - \ln(\widetilde{N}_1/N_0), \quad \widetilde{N}_1 \sim \mathrm{Pois}(N_0 \exp(-Ax)). \tag{7}$$

Here, $\mathrm{Pois}(\lambda)$ denotes the joint distribution of m Poisson distributed observations with parameters $\lambda_1, \ldots, \lambda_m$, respectively. Note that since the negative logarithm is applied to the observations, the noisy post-log values y^δ do not follow a Poisson distribution but the distribution resulting from this log-transformation. However, taking the negative logarithm is required to obtain the linear model and therefore is most commonly applied as a preprocessing step. For our dataset, we consider post-log values by default.

The Radon transform is a linear and compact operator. Therefore, the continuous inverse problem of CT is mildly ill-posed in the sense of Nashed[30,31]. This means that small variations in the measurements can lead to significant differences in the reconstruction (unstable inversion). While the discretised inverse problem is not ill-posed, it is typically ill-conditioned[28], which leads to artefacts in reconstructions obtained by direct inversion from noisy measurements.

Fig. 3 Scans from the LIDC/IDRI database[15] with poor quality, good quality and an artefact. The shown HU window is [−1024, 1023].

For our discrete simulation setting, we use the described model with following dimensions and parameters:

- Image resolution of 362 px × 362 px on a domain of size 26 cm × 26 cm.
- 513 equidistant detector bins s spanning the image diameter.
- 1000 equidistant angles φ between 0 and π.
- Mean photon count per detector bin without attenuation $N_0 = 4096$.

LIDC/IDRI database and data selection. The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI) published the LIDC/IDRI database[15,21,22] to support the development of CAD methods for the detection of lung nodules. The dataset consists of 1018 helical thoracic CT scans of 1010 individuals. Seven academic centres and eight medical imaging companies collaborated for the creation of the database. As a result, the data is heterogeneous with respect to the technical parameters and scanner models.

Both standard-dose and lower-dose scans are part of the dataset. Tube peak voltages range from 120 kV to 140 kV and tube current from 40 mA to 627 mA with a mean of 222.1 mA. Labels for the lung nodules were created by a group of 12 radiologists in a two-phase process. The image reconstruction was performed with different filters, depending on the manufacturer of the scanner. Figure 3 shows examples of the provided reconstructions. The LIDC/IDRI database is freely available from The Cancer Imaging Archive (TCIA)[22]. It is published under the Creative Commons Attribution 3.0 Unported License (https://creativecommons.org/licenses/by/3.0/).

The LoDoPaB-CT dataset is based on the LIDC/IDRI scans. Our dataset is intended for the evaluation of reconstruction methods in a low-dose setting. Therefore, we simulate the projection data, which is not included in the LIDC/IDRI database. In order to enable a fair comparison with good ground truth, scans that are too noisy were removed in a manual selection process (cf. section "Technical Validation"). Additional scans were excluded due to their geometric properties, namely an image size different from 512 px × 512 px, a too small area of valid pixel values (cf. subsection "Ground truth image extraction" below), or a different patient orientation. The complete lists of excluded scan series are given in file series_list.json in the technical reference repository[26]. In the end, 812 patients remain in the LoDoPaB-CT dataset.

The dataset is split into four parts: three parts for training, validation and testing, respectively, and a "challenge" part reserved for the LoDoPaB-CT Challenge (https://lodopab.grand-challenge.org/). Each part contains scans from a distinct set of patients, as we want to study the case of learned reconstructors being applied to patients that are not known from training. The training set features scans from 632 patients, while the other parts contain scans from 60 patients each. Every scan contains multiple slices (2D images) for different z-positions, of which only a subset is included. The amount of extracted slices depends on the slice thickness obtained from the metadata. As slices with small distances are similar, they may not provide much additional information while increasing the chances to overfit. The distances of the extracted slices are larger than 5.0 mm for >45% and larger than 2.5 mm for >75% of the slices. In total, the dataset contains 35820 training images, 3522 validation images, 3553 test images and 3678 challenge images.

Remark. We propose to use our default dataset split, as it allows for a fair comparison with other methods that use the same split. However, users are free to remix or re-split the dataset parts. For this purpose, randomised patient IDs are provided, i.e., the same random ID is given for all slices obtained from one patient. Thus, when creating custom splits it can be regulated whether—and to what extent—data from the same patients are contained in different splits.

Ground truth image extraction. First, each image is cropped to the central rectangle of 362 px × 362 px. This is done because most of the images contain (approximately) circle-shaped reconstructions with a diameter of 512 px (cf. Figure 3). After the crop, the image only contains pixels that lie inside this circle, which avoids value jumps occurring at the border of the circle. While this yields natural ground truth images, we need to point out that the cropped images, in general, do not show the full subject but some interior part. Hence, it is unlikely for methods trained with this dataset to perform well on full-subject measurements.

$(\mu_{\text{air}}, \mu_{\text{water}}, \mu_{\text{max}}) \leftarrow (0.02, 20.0, 81.35858)$

procedure EXTRACTGROUNDTRUTH(z) ▷ z is an image from the LIDC dataset in the DICOM data format

  $\bar{z} \leftarrow \text{transpose}(\text{center\_crop}(z, 362 \times 362))$ ▷ Take center crop of size 362×362

  $image\_hu \leftarrow \bar{z} * z.RescaleSlope + z.RescaleIntercept$ ▷ Transform pixels to HU values

  $image\_hu \leftarrow image\_hu + \text{noise}$ ▷ Add dequantisation noise uniformly distributed in $[0, 1]$ for each pixel

  $image\_mu \leftarrow image\_hu * (\mu_{\text{water}} - \mu_{\text{air}})/1000 + \mu_{\text{water}}$ ▷ Convert HU values to linear attenuation

  return $\text{clip}(image\_mu/\mu_{\text{max}}, min = 0, max = 1)$ ▷ Clip all pixel values to be within $[0, 1]$

procedure GENERATEPROJECTION(x) ▷ x is a ground truth generated by the previous procedure

  $\bar{x} \leftarrow \text{bilinear\_interpolation}(\mu_{\text{max}} * x, 1000 \times 1000)$ ▷ Interpolate ground truth to avoid inverse crime

  $y \leftarrow \text{ray\_trafo}(\bar{x})$ ▷ ODL `RayTransform` with 1000 angles, 513 detector pixels and image domain $[-0.13, 0.13]^2$

  $photons \leftarrow \max(\text{poisson}(\exp(-y) * 4096), 0.1)$ ▷ Scale and add Poisson noise. Max is taken pixel-wise.

  return $-\log(photons/4096)/\mu_{\text{max}}$ ▷ Apply log-transform and normalisation constant

Fig. 4 Data generation algorithm.

For some scan series, the circle is subject to a geometric transformation either shrinking or expanding the circle in some directions. In particular, for a few scan series, the circle is shrunk such that it is smaller than the cropped rectangle. We exclude these series, i.e. those with patient IDs 0004, 0032, 0102, 0116, 0120, 0289, 0368, 0418, 0541, 0798, 0926, 0972 and 1000, from our dataset, which allows to crop all included images consistently to 362 px × 362 px.

The integer Hounsfield unit (HU) values obtained from the DICOM files are dequantised by adding uniform noise from the interval [0, 1). By adding this noise, the discrete distribution of stored values is transformed into a continuous distribution (up to the floating-point precision), which is a common assumption of image models. For example, the meaningful evaluation of densities learned by generative networks requires dequantization[32], which in some works[33] is more refined than the uniform dequantization applied to the HU values in our dataset.

In the next step, the linear attenuations μ are computed from the dequantised HU values using the definition of the HU,

$$\text{HU} = 1000\frac{\mu - \mu_{\text{water}}}{\mu_{\text{water}} - \mu_{\text{air}}} \quad \Leftrightarrow \quad \mu = \text{HU}\frac{\mu_{\text{water}} - \mu_{\text{air}}}{1000} + \mu_{\text{water}}, \tag{8}$$

where we use the linear attenuation coefficients

$$\mu_{\text{water}} = 20/\text{m}, \quad \mu_{\text{air}} = 0.02/\text{m}, \tag{9}$$

which approximately correspond to an X-ray energy of 60 keV[34]. Finally, the μ-values are normalised into [0, 1] by dividing by

$$\mu_{\text{max}} = 3071\frac{\mu_{\text{water}} - \mu_{\text{air}}}{1000} + \mu_{\text{water}} = 81.35858/\text{m}, \tag{10}$$

which corresponds to the largest HU value that can be represented with the standard 12-bit encoding, i.e. $(2^{12}–1–1024)\text{HU} = 3071$ HU, followed by the clipping of all values into the range [0, 1],

$$\hat{\mu} = \text{clip}(\mu/\mu_{\text{max}}, [0, 1]) = \begin{cases} 0 & ,\mu/\mu_{\text{max}} \leq 0 \\ \mu/\mu_{\text{max}} & ,0 < \mu/\mu_{\text{max}} \leq 1. \\ 1 & ,1 < \mu/\mu_{\text{max}} \end{cases} \tag{11}$$

The Eqs. (8) and (11) are applied pixel-wise to the images.

Projection data generation. To simulate the measurements based on the virtual ground truth images, the main step is to apply the forward operator, which is the ray transform (Radon transform in 2D) for CT. For this task we utilise the Operator Discretization Library[35] (ODL) with the '`astra_cpu`' backend[36].

Remark. We choose '`astra_cpu`' over the usually favoured '`astra_cuda`' because of small inaccuracies observed in the sinograms when using '`astra_cuda`', specifically at angles 0, $\frac{\pi}{2}$ and π and detector positions $-1/\sqrt{2}\frac{l}{2}$ and $1/\sqrt{2}\frac{l}{2}$ with l being the length of the detector. The used version is `astra-toolbox==1.8.3` on Python 3.6. The tested CUDA version is 9.1 combined with `cudatoolkit==8.0`.

In order to avoid "committing the inverse crime"[37], which, in our scenario, would be to use the same discrete model both for simulation and reconstruction, we use a higher resolution for the simulation. Otherwise, good performance of reconstructors for the specific resolution of this dataset (362 px × 362 px) could also stem from the properties of the specific discretised problem, rather than from good inversion of the analytical model. We use bilinear interpolation for the upscaling of the virtual ground truth from 362 px × 362 px to 1000 px × 1000 px.

The non-normalised, upscaled image is projected by the ray transform. Based on this projection, Ax, the measured photon counts $\widetilde{N}_1$ are sampled according to Eq. (7). The sampling in some cases yields photon counts of zero, which we then replace by photon counts of 0.1. Hereby strictly positive values are ensured, which is a prerequisite for the log-transform in the next step (cf. Wang *et al.*[38]). The negative logarithm of the photon counts quotient $\max(0, 1, \widetilde{N}_1)N_0$ is taken, resulting in the post-log measurements y^δ according to Eq. (7) (up to the 0.1 photon count approximation). Finally, y^δ is divided by $\mu_{\max}$ to match the normalised ground truth images. A summary of all steps can be found in Fig. 4 (Data generation algorithm).

Remark. Although the linear model obtained by the log-transform is easier to study, in some cases pre-log models are more accurate. See Fu *et al.*[29] for a detailed comparison. For applying a pre-log method, the stored observation data $\hat{y} = y^\delta/\mu_{\max}$ must be back-transformed by $\exp(-\mu_{\max} \cdot \hat{y})$. To create physically consistent data pairs, the ground truth images should then be multiplied with $\mu_{\max}$, too.

Remark. Note that the minimum photon count of 0.1 can be adapted subsequently. This is most easily done by filtering out the highest observation values and replacing them with $-\log(\varepsilon_0/4096)/\mu_{\max}$, where ε_0 is the new minimum photon count.

Data Records

The LoDoPaB-CT dataset is published as open access on Zenodo (https://zenodo.org) in two repositories. The main data repository[39] (https://doi.org/10.5281/zenodo.3384092) has a size of around 55GB and contains observations and ground truth data of the train, validation and test set. For each subset, represented by *, the following files are included:

- CSV files `patient_ids_rand_*.csv` include randomised patient IDs of the samples. The patient IDs of the train, validation and test parts are integers in the range of 0–631, 632–691 and 692–751, respectively. The ID of each sample is stored in a single row.
- Zip archives `ground_truth_*.zip` contain HDF5[40] files of the ground truth reconstructions.
- Zip archives `observation_*.zip` contain HDF5 files of the simulated low-dose measurements.
- Each HDF5 file contains one HDF5 dataset named data, that provides several samples (128 except for the last file in each ZIP file). For example, the n-th training sample pair is stored in the HDF5 `files observation_train_%03d.hdf5` and `ground_truth_train_%03d.hdf5` where the placeholder `%03d` is floor (n/128). Within these HDF5 files, the observation or ground truth is stored at entry (n mod 128) of the HDF5 dataset data.

The second repository[41] for the challenge data (https://doi.org/10.5281/zenodo.3874937) consists of a single zip archive:

- `observation_challenge.zip` contains HDF5 files of the simulated low-dose measurements.

The structure inside the HDF5 files is the same as in the main repository.

Technical Validation

Ground truth & data selection. Creating high-quality ground truth images for tomographic image reconstruction is a challenging and time-consuming task. In computed tomography, one option is to cut open the object after the scan or use 3D printing[42], whereby the digital template of the object is the reference. In general, this also involves high radiation doses and many scanning angles. This combination makes it even harder to generate ground truth images for medical applications.

For low-dose CT reconstruction models, the primary goal is to match the normal-dose reconstruction quality of methods currently in use. Therefore, normal-dose reconstructions from classical methods, e.g. filtered back-projection, are an adequate choice as ground truth. This simplifies the process considerably.

The ground truth CT reconstructions of LoDoPaB-CT are taken from the established and well-documented LIDC/IDRI database. An independent visual inspection of one 2D slice per scan was performed by three of the authors. Figure 3 shows three examples of such slices. A five-star rating system was used to evaluate the image quality and remove noisy ground truth data, like the first slice in Fig. 3. Scans with artefacts, e.g. from photon starvation due to dense material (cf. Figure 3 (right)), were in general not removed, as the artefacts only affect a few slices of the whole scan. The slice in the middle of Fig. 3 represents an ideal ground truth. The following procedure was then used to exclude scans based on their rating:

1. Centring of the ratings from each evaluator around the value 3.
2. Calculation of the mean rating and the variance for each looked at 2D slice.
3. For a variance <1, the mean was used as the rating score. Otherwise, the scan is evaluated by all three authors together.
4. All scans with a rating ≤ 2 are excluded from the dataset.

These excluded scans are listed at key "`series_excluded_manual_low_q_filter`" in file series_list.json in the technical reference repository[26].

Reference reconstructions & quantitative results. To validate the usability of the proposed dataset for machine learning approaches, we provide reference reconstructions and quantitative results for the standard

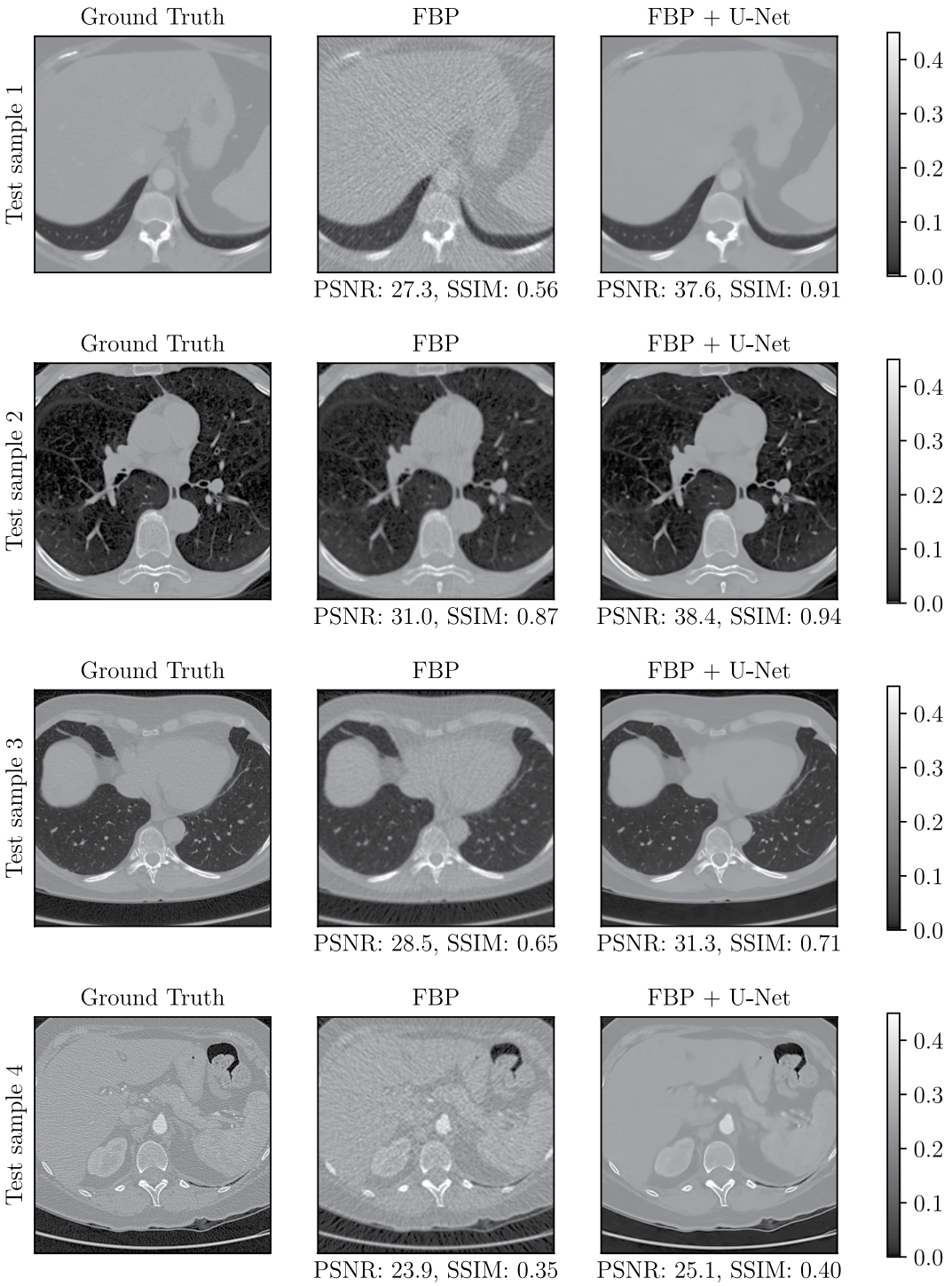

Fig. 5 Different baseline reconstructions from the FBP and FBP + U-Net methods. The ground truth images are part of the LoDoPaB-CT test set. The window [0, 0.45] corresponds to a HU range of ≈[−1001, 831].

filtered back-projection (FBP) and a learned post-processing method (FBP + U-Net). FBP is a widely used analytical reconstruction technique (cf. Buzug[28] for an introduction). If the measurements are noisy (due to the low dose), FBP reconstructions tend to include streaking artefacts. A typical approach to overcome this problem is to apply some post-processing such as denoising. Recent works[3,4,8] have successfully used convolutional neural networks, such as the U-Net[43]. The idea is to train a neural network to create clean reconstructions out of the noisy FBP results.

In this initial study, for the FBP, we used the Hann filter with a frequency scaling of 0.641. We selected these parameters based on the performance over the first 100 samples of the validation dataset. For the post-processing approach (FBP + U-Net), we used a U-Net-like architecture with 5 scales. We trained it using the proposed

	training set		validation set		test set	
	PSNR (dB)	SSIM	PSNR (dB)	SSIM	PSNR (dB)	SSIM
FBP	30.45 ± 2.65	0.7415 ± 0.1314	30.75 ± 2.52	0.7577 ± 0.1231	30.52 ± 3.10	0.7372 ± 0.1467
FBP + U-Net	36.17 ± 3.75	0.8623 ± 0.1228	36.74 ± 3.28	0.8819 ± 0.1017	35.84 ± 4.59	0.8443 ± 0.1501

Table 1. Baseline performance. Values are the mean and standard deviation over all samples.

dataset by minimising the mean squared error loss with the Adam algorithm[44] for a maximum of 250 epochs with batch size 32. Additionally, we used an initial learning rate of 10^{-3}, decayed using cosine annealing until 10^{-4}. The model with the highest mean peak signal-to-noise ratio (PSNR) on the validation set was selected from the models obtained during training. Sample reconstructions are shown in Fig. 5.

Table 1 depicts the obtained results in terms of the peak signal-to-noise ratio (PSNR) and structural similarity[45] (SSIM) metrics (cf. "Evaluation practice" in the next section for a detailed explanation). As it can be observed, the post-processing approach, which was trained using the proposed dataset, outperforms the classical FBP reconstructions by a margin of 5 dB. This demonstrates that the dataset indeed contains valuable data ready to be used for training machine learning methods to obtain CT reconstructions with higher quality than the standard methods.

Usage Notes

Download & Easy access. The whole LoDoPaB-CT dataset[39,41] can be downloaded directly from the Zenodo website. However, we recommend the Python library DIV$\alpha\ell$[46] (https://github.com/jleuschn/dival) for easy access of the dataset. The library includes specific functionalities for the interaction with the provided dataset.

Remark. Access to the dataset on Zenodo might be restricted or slow in some regions of the world. In this case please contact one of the corresponding authors to get an alternative download option.

DIV$\alpha\ell$ is also available through the package index PyPI (https://pypi.org/project/dival). With the library, the dataset is automatically downloaded, checked for corruption and ready for use within two lines of Python code:

```
from dival import get_standard_dataset
dataset = get_standard_dataset('lodopab').
```

Remark. When loading the dataset using DIV$\alpha\ell$, an ODL[35] `RayTransform` implementing the forward operator is created. This requires a backend, the default being '`astra_cuda`', which requires both the astra toolbox[36] and CUDA to be available. If either is unavailable, a different backend ('`astra_cpu`' or '`skimage`') must be selected by keyword argument `impl`.

In addition, DIV$\alpha\ell$ offers multiple options to work with the LoDoPaB-CT dataset:

- Access the train, validation and test subset and draw a specific number of samples.
- Sort the data by the patient ids.
- Use the pre-log or post-log data (cf. projection data generation in the "Methods" section).
- Evaluate the reconstruction performance.
- Compare with pre-trained standard reconstruction models.

Evaluation practice. Since ground truth data is provided in the dataset, we recommend using so-called full-reference methods for the evaluation. The peak signal-to-noise ratio (PSNR) and the structural similarity[45] (SSIM) are two standard image quality metrics often used in CT applications[42,47]. While the PSNR calculates pixel-wise intensity comparisons between ground truth and reconstruction, SSIM captures structural distortions.

Peak signal-to-noise ratio. The PSNR expresses the ratio between the maximum possible image intensity and the distorting noise, measured by the mean squared error (MSE),

$$\mathrm{PSNR}(\tilde{x},\, x) := 10\log_{10}\left(\frac{\max_x^2}{\mathrm{MSE}(\tilde{x},\, x)}\right), \qquad \mathrm{MSE}(\tilde{x},\, x) := \frac{1}{n}\sum_{i=1}^{n}|\tilde{x}_i - x_i|^2. \tag{12}$$

Here x is the ground truth image and $\tilde{x}$ the reconstruction. Higher PSNR values are an indication of a better reconstruction. We recommend choosing $\max_x = \max(x) - \min(x)$, i.e. the difference between the highest and lowest entry in x, instead of the maximum possible intensity, since the reference value of 3071HU is far from the most common values. Otherwise, the results can often be too optimistic.

Structural similarity. Based on assumptions about the human visual perception, SSIM compares the overall image structure of ground truth and reconstruction. Results lie in the range [0, 1], with higher values being better. The SSIM is computed through a sliding window at M locations

$$\mathrm{SSIM}(\tilde{x},\, x) := \frac{1}{M}\sum_{j=1}^{M}\frac{(2\tilde{\mu}_j\mu_j + C_1)\,(2\Sigma_j + C_2)}{(\tilde{\mu}_j^2 + \mu_j^2 + C_1)\,(\tilde{\sigma}_j^2 + \sigma_j^2 + C_2)}, \tag{13}$$

where $\widetilde{\mu_j}$ and μ_j are the average pixel intensities, $\widetilde{\sigma_j}$ and σ_j the variances and Σ_j the covariance of $\widetilde{x}$ and x at the j-th local window. Constants $C_1 = (K_1 L)^2$ and $C_2 = (K_2 L)^2$ stabilise the division. Following Wang *et al.*[45] we choose $K_1 = 0.01$ and $K_2 = 0.03$ for the technical validation in this paper. The window size is 7×7 and $L = \max(x) - \min(x)$.

Test & challenge set. The test data is the advised subset for offline model evaluation. To guarantee a fair comparison, the data should be in no way involved in the training process or hyperparameter selection of the model. We recommend using the whole test set and select the above-mentioned parameters for PSNR and SSIM. Deviations from this setting should be mentioned.

In addition, a challenge set without ground truth images is provided. We encourage users to submit their challenge reconstructions to the evaluation website (https://lodopab.grand-challenge.org/). All methods are assessed under the same conditions and with the same metrics. The performance can be directly compared with other methods on a public leaderboard. Therefore, we recommend to report performance measures on the challenge set for publications that use the LoDoPaB-CT dataset without modifications, in addition to any evaluations on the test set. In accordance with the Biomedical Image Analysis (BIAS) guidelines[48], more information about the challenge can be found on the aforementioned website.

Further usage. *Scan scenarios.* The provided measurements and simulation scripts can easily be modified to cover different scan scenarios:

- Limited and sparse-angle problems can be created by loading a subset of the projection data, e.g. a sparser setup with 200 angles was already used by Baguer *et al.*[25].
- Super-resolution experiments can be mimicked, by artificially binning the projection data into larger pixels.
- To study lower or higher photon counts, the dataset can be re-simulated with a different value of N_0 (e.g. using resimulate_observations.py[26] by changing the value of `PHOTONS_PER_PIXEL`).

The provided reconstructions can still be used as ground truth for all listed scenarios.

Transfer learning. Transfer learning is a popular approach to boost the performance of machine learning models on smaller datasets. The idea is to first train the model on a different, comprehensive data collection. Afterwards, the determined parameters are used as an initial guess for fine-tuning the model on the smaller one. In general, the goal is to learn to process low-level features, e.g. edges in images, from the comprehensive dataset. The adaption to specific high-level features is then performed on the smaller dataset. For imaging applications, the ImageNet database[49], with over 14 million natural images, is frequently used in this role. The applications range from image classification[50] to other domains like audio data[51].

Transfer learning has also been successfully applied to CT reconstruction tasks. This includes training on different scan scenarios[52,53], e.g. a different number of angles, as well as first training on 2D data and continuing on 3D data[54]. He *et al.*[55] simulated parallel beam measurements on some of the natural images contained in ImageNet. Subsequently, the training was continued on CT images from the Mayo Clinic[10]. LoDoPaB-CT, or parts of the dataset, can be used in similar roles for transfer learning. Additionally, the ground truth data from real thoracic CT scans may be advantageous for similar CT reconstruction tasks compared to random natural images from ImageNet[56].

Nonetheless, we advise the user to check the applicability for their specific use case and reconstruction model. Re-simulation or other changes to the LoDoPaB-CT dataset might be needed, especially for datasets with different scan geometries. Additionally, simulated data can not capture all aspects of real-world measurements and therefore cause reconstruction errors. For a comprehensive study on the benefits and challenges of transfer learning for medical imaging, we refer the reader to the publication by Raghu *et al.*[56].

Remark. An example for a simulation script with a fan beam geometry on the ground truth data can be found in the DIV$\alpha\ell$[46] library: `dival/examples/ct_simulate_fan_beam_from_lodopab_ground_truth.py`.

Limits of the dataset. The LoDoPaB-CT dataset is designed for a methodological comparison of CT reconstruction methods on a simulated low-dose parallel beam setting. The focus is on how a model deals with the challenges that arise from low photon count measurements to match the quality of normal-dose images. Of course, this represents only one aspect of many for the application in real-world scenarios. Therefore, results achieved on LoDoPaB-CT might not completely reflect the performance on real medical data. The following limits of the dataset should be considered when evaluating and comparing results:

- The simulation uses the Radon transform and Poisson noise. Real measurements can be influenced by additional physical effects, like scattering.
- Modern CT machines use advanced scanning geometries, like helical fan beam or cone beam. Specific challenges for the reconstruction can arise compared to parallel beam measurements (cf. Buzug[28]).
- In general, the goal is to reconstruct a whole 3D subject and not just a single 2D slice. Reconstruction methods might benefit from additional spacial information. On the other hand, requirements on memory and compute power can be higher for methods that reconstruct 3D volumes directly.

- Image metrics, e.g. PSNR and SSIM, cannot express and cover all aspects of high-quality CT reconstruction. An additional assessment by experts in the field can be beneficial.
- The ground truth images are based on reconstructions from normal-dose medical scans. As such, they can contain noise and artefacts. The measurements are created from this "noisy" ground truth. Therefore, a perfect reconstruction model would re-create the imperfections. Approaches that are designed to remove them can score lower PSNR and SSIM values, although their reconstruction quality might be higher.
- A crop to a region of interest is used for the ground truth images (cf. "Ground truth image extraction"). Hence, the results for full-subject measurements can be different.

Code availability

Python scripts[26] for the simulation setup and the creation of the dataset are publicly available on Github (https://github.com/jleuschn/lodopab_tech_ref). They make use of the ASTRA Toolbox[36] (version 1.8.3) and the Operator Discretization Library[35] (ODL, version $\geq$0.7.0). In addition, the ground truth reconstructions from the LIDC/IDRI database[21] are needed for the simulation process. A sample data split into training, validation, test and challenge part is also provided. It differs from the one used for the creation of this dataset in order to keep the ground truth data of the challenge set undisclosed. The random seeds used in the scripts are modified for the same reason. The authors acknowledge the National Cancer Institute and the Foundation for the National Institutes of Health, and their critical role in the creation of the free publicly available LIDC/IDRI database used in this study.

Received: 12 November 2020; Accepted: 4 March 2021;
Published online: 16 April 2021

References

1. Benning, M. & Burger, M. Modern regularization methods for inverse problems. *Acta Numerica* **27**, https://doi.org/10.1017/S0962492918000016 (2018).
2. Adler, J. & Öktem, O. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems* **33**, 124007, https://doi.org/10.1088/1361-6420/aa9581 (2017).
3. Chen, H. *et al.* Low-dose CT with a residual encoder-decoder convolutional neural network. *IEEE Transactions on Medical Imaging* **36**, 2524–2535, https://doi.org/10.1109/TMI.2017.2715284 (2017).
4. Jin, K. H., McCann, M. T., Froustey, E. & Unser, M. Deep convolutional neural network for inverse problems in imaging. *IEEE Transactions on Image Processing* **26**, 4509–4522, https://doi.org/10.1109/TIP.2017.2713099 (2017).
5. Li, H., Schwab, J., Antholzer, S. & Haltmeier, M. NETT: solving inverse problems with deep neural networks. *Inverse Problems* **36**, 065005, https://doi.org/10.1088/1361-6420/ab6d57 (2020).
6. Shan, H. *et al.* Competitive performance of a modularized deep neural network compared to commercial algorithms for low-dose CT image reconstruction. *Nature Machine Intelligence* **1**, 269–276, https://doi.org/10.1038/s42256-019-0057-9 (2019).
7. Wang, G., Ye, J. C., Mueller, K. & Fessler, J. A. Image reconstruction is a new frontier of machine learning. *IEEE Transactions on Medical Imaging* **37**, 1289–1296, https://doi.org/10.1109/TMI.2018.2833635 (2018).
8. Yang, Q. *et al.* Low-dose CT image denoising using a generative adversarial network with Wasserstein distance and perceptual loss. *IEEE Transactions on Medical Imaging* **37**, 1348–1357, https://doi.org/10.1109/TMI.2018.2827462 (2018).
9. McCollough, C. *et al.* Data from Low Dose CT Image and Projection Data. *The Cancer Imaging Archive* https://doi.org/10.7937/9npb-2637 (2020).
10. McCollough, C. TU-FG-207A-04: Overview of the Low Dose CT Grand Challenge. *Medical Physics* **43**, 3759–3760, https://doi.org/10.1118/1.4957556 (2016).
11. Hämäläinen, K. *et al.* Tomographic X-ray data of a walnut. Preprint at https://arxiv.org/abs/1502.04064 (2015).
12. Bubba, T. A., Hauptmann, A., Huotari, S., Rimpeläinen, J. & Siltanen, S. Tomographic X-ray data of a lotus root filled with attenuating objects. Preprint at https://arxiv.org/abs/1609.07299 (2016).
13. Der Sarkissian, H. *et al.* A cone-beam X-ray computed tomography data collection designed for machine learning. *Scientific Data* **6**, 215, https://doi.org/10.1038/s41597-019-0235-y (2019).
14. Knoll, F. *et al.* fastMRI: A publicly available raw k-space and DICOM dataset of knee images for accelerated MR image reconstruction using machine learning. *Radiology: Artificial Intelligence* **2**, e190007, https://doi.org/10.1148/ryai.2020190007 (2020).
15. Armato, S. G. III *et al.* The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): A completed reference database of lung nodules on CT scans. *Med. Phys.* **38**, 915–931, https://doi.org/10.1118/1.3528204 (2011).
16. Masoudi, M. *et al.* A new dataset of computed-tomography angiography images for computer-aided detection of pulmonary embolism. *Scientific Data* **5**, 180180 EP, https://doi.org/10.1038/sdata.2018.180 (2018).
17. Heller, N. *et al.* The KiTS19 Challenge data: 300 kidney tumor cases with clinical context, CT semantic segmentations, and surgical outcomes. Preprint at https://arxiv.org/abs/1904.00445 (2019).
18. Shiraishi, J. *et al.* Development of a digital image database for chest radiographs with and without a lung nodule. *American Journal of Roentgenology* **174**, 71–74, https://doi.org/10.2214/ajr.174.1.1740071 (2000).
19. Clark, K. W. *et al.* Creation of a CT image library for the lung screening study of the National Lung Screening Trial. *Journal of Digital Imaging* **20**, 23–31, https://doi.org/10.1007/s10278-006-0589-5 (2007).
20. Cody, D. D. *et al.* Normalized CT dose index of the CT scanners used in the national lung screening trial. *American Journal of Roentgenology* **194**, 1539–1546, https://doi.org/10.2214/AJR.09.3268 (2010).
21. Armato, S. G. III *et al.* Data from LIDC-IDRI. *The Cancer Imaging Archive* https://doi.org/10.7937/K9/TCIA.2015.LO9QL9SX (2015).
22. Clark, K. *et al.* The Cancer Imaging Archive (TCIA): Maintaining and operating a public information repository. *Journal of Digital Imaging* **26**, 1045–1057, https://doi.org/10.1007/s10278-013-9622-7 (2013).
23. Defrise, M., Noo, F. & Kudo, H. Rebinning-based algorithms for helical cone-beam CT. *Physics in Medicine and Biology* **46**, 2911–2937, https://doi.org/10.1088/0031-9155/46/11/311 (2001).
24. Etmann, C., Ke, R. & Schönlieb, C. iUNets: Learnable invertible up- and downsampling for large-scale inverse problems. In *30th IEEE International Workshop on Machine Learning for Signal Processing, MLSP 2020, Espoo, Finland, September 21–24, 2020*, 1–6, https://doi.org/10.1109/MLSP49062.2020.9231874 (IEEE, 2020).
25. Baguer, D. O., Leuschner, J. & Schmidt, M. Computed tomography reconstruction using deep image prior and learned reconstruction methods. *Inverse Problems* **36**, 094004, https://doi.org/10.1088/1361-6420/aba415 (2020).
26. Leuschner, J., Schmidt, M. & Baguer, D. O. LoDoPaB-CT Generation Technical Reference ($\geq$v1.2). *Zenodo* https://doi.org/10.5281/zenodo.3957743 (2020).

27. Radon, J. On the determination of functions from their integral values along certain manifolds. *IEEE Transactions on Medical Imaging* **5**, 170–176, https://doi.org/10.1109/TMI.1986.4307775 (1986).
28. Buzug, T. *Computed Tomography: From Photon Statistics to Modern Cone-Beam CT* (Springer Berlin Heidelberg, 2008).
29. Fu, L. *et al*. Comparison between pre-log and post-log statistical models in ultra-low-dose CT reconstruction. *IEEE Transactions on Medical Imaging* **36**, 707–720, https://doi.org/10.1109/TMI.2016.2627004 (2017).
30. Nashed, M. A new approach to classification and regularization of ill-posed operator equations. In Engl, H. W. & Groetsch, C. (eds.) *Inverse and Ill-Posed Problems*, 53–75, https://doi.org/10.1016/B978-0-12-239040-1.50009-0 (Academic Press, 1987).
31. Natterer, F. *The mathematics of computerized tomography*. No. 32 in Classics in applied mathematics (Society for Industrial and Applied Mathematics, Philadelphia, 2001).
32. Theis, L., van den Oord, A. & Bethge, M. A note on the evaluation of generative models. In Bengio, Y. & LeCun, Y. (eds.) *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings* (2016).
33. Ho, J., Chen, X., Srinivas, A., Duan, Y. & Abbeel, P. Flow++: Improving flow-based generative models with variational dequantization and architecture design. vol. 97 of *Proceedings of Machine Learning Research*, 2722–2730 (PMLR, Long Beach, California, USA, 2019).
34. Hubbell, J. & Seltzer, S. Tables of X-ray mass attenuation coefficients and mass energy-absorption coefficients 1 keV to 20 meV for elements z = 1 to 92 and 48 additional substances of dosimetric interest. Tech. Rep. PB-95-220539/XAB; NISTIR-5632; TRN: 51812148, National Inst. of Standards and Technology - PL, Gaithersburg, MD (United States). Ionizing Radiation Div. https://doi.org/10.18434/T4D01F (1995).
35. Adler, J. *et al*. odlgroup/odl: ODL 0.7.0. *Zenodo* https://doi.org/10.5281/zenodo.592765 (2018).
36. van Aarle, W. *et al*. The ASTRA Toolbox: A platform for advanced algorithm development in electron tomography. *Ultramicroscopy* **157**, 35–47, https://doi.org/10.1016/j.ultramic.2015.05.002 (2015).
37. Wirgin, A. The inverse crime. Preprint at https://arxiv.org/abs/math-ph/0401050 (2004).
38. Wang, G., Zhou, J., Yu, Z., Wang, W. & Qi, J. Hybrid pre-log and post-log image reconstruction for computed tomography. *IEEE Transactions on Medical Imaging* **36**, 2457–2465, https://doi.org/10.1109/TMI.2017.2751679 (2017).
39. Leuschner, J., Schmidt, M. & Baguer, D. O. LoDoPaB-CT dataset (v1.0.0). *Zenodo* https://doi.org/10.5281/zenodo.3384092 (2019).
40. The HDF Group. Hierarchical Data Format, version 5 (1997). https://www.hdfgroup.org/HDF5/.
41. Leuschner, J., Schmidt, M. & Baguer, D. O. LoDoPaB-CT challenge set (v1.0.0). *Zenodo* https://doi.org/10.5281/zenodo.3874937 (2020).
42. Joemai, R. M. S. & Geleijns, J. Assessment of structural similarity in CT using filtered backprojection and iterative reconstruction: a phantom study with 3D printed lung vessels. *The British Journal of Radiology* **90**, 20160519, https://doi.org/10.1259/bjr.20160519 (2017).
43. Ronneberger, O., Fischer, P. & Brox, T. U-Net: Convolutional networks for biomedical image segmentation. In Navab, N., Hornegger, J., Wells, W. M. & Frangi, A. F. (eds.) *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*, 234–241, https://doi.org/10.1007/978-3-319-24574-4_28 (Springer International Publishing, Cham, 2015).
44. Kingma, D. P. & Ba, J. Adam: A method for stochastic optimization. Preprint at https://arxiv.org/abs/1412.6980 (2014).
45. Wang, Z., Bovik, A. C., Sheikh, H. R. & Simoncelli, E. P. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* **13**, 600–612, https://doi.org/10.1109/TIP.2003.819861 (2004).
46. Leuschner, J., Schmidt, M., Baguer, D. O. & Erzmann, D. DIVal library. *Zenodo* https://doi.org/10.5281/zenodo.3970516 (2021).
47. Adler, J. & Öktem, O. Learned primal-dual reconstruction. *IEEE Transactions on Medical Imaging* **37**, 1322–1332, https://doi.org/10.1109/TMI.2018.2799231 (2018).
48. Maier-Hein, L. *et al*. BIAS: Transparent reporting of biomedical image analysis challenges. *Medical Image Analysis* **66**, 101796, https://doi.org/10.1016/j.media.2020.101796 (2020).
49. Deng, J. *et al*. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255, https://doi.org/10.1109/CVPR.2009.5206848 (2009).
50. Shermin, T. *et al*. Enhanced transfer learning with ImageNet trained classification layer. In Lee, C., Su, Z. & Sugimoto, A. (eds.) *Image and Video Technology*, 142–155, https://doi.org/10.1007/978-3-030-34879-3_12 (Springer International Publishing, Cham, 2019).
51. Grzywczak, D. & Gwardys, G. Deep image features in music information retrieval. *International Journal of Electronics and Telecommunications* **60**, 187–199, https://doi.org/10.1007/978-3-319-09912-5_16 (2014).
52. Wu, Z., Yang, T., Li, L. & Zhu, Y. Hierarchical convolutional network for sparse-view X-ray CT reconstruction. In Mahalanobis, A., Tian, L. & Petruccelli, J. C. (eds.) *Computational Imaging IV*, vol. 10990, 141–146, https://doi.org/10.1117/12.2521239. International Society for Optics and Photonics (SPIE, 2019).
53. Kalare, K. W. & Bajpai, M. K. RecDNN: Deep neural network for image reconstruction from limited view projection data. *Soft Comput.* **24**, 17205–17220, https://doi.org/10.1007/s00500-020-05013-4 (2020).
54. Shan, H. *et al*. 3-d convolutional encoder-decoder network for low-dose CT via transfer learning from a 2-d trained network. *IEEE Transactions on Medical Imaging* **37**, 1522–1534, https://doi.org/10.1109/TMI.2018.2832217 (2018).
55. He, J., Wang, Y. & Ma, J. Radon inversion via deep learning. *IEEE Transactions on Medical Imaging* **39**, 2076–2087, https://doi.org/10.1109/TMI.2020.2964266 (2020).
56. Raghu, M., Zhang, C., Kleinberg, J. & Bengio, S. Transfusion: Understanding transfer learning for medical imaging. In Wallach, H. *et al*. (eds.) *Advances in Neural Information Processing Systems*, vol. 32, 3347–3357 (Curran Associates, Inc., 2019).

Acknowledgements

Johannes Leuschner, Maximilian Schmidt and Daniel Otero Baguer acknowledge the support by the Deutsche Forschungsgemeinschaft (DFG) within the framework of GRK 2224/1 "π^3: Parameter Identification – Analysis, Algorithms, Applications". We thank Simon Arridge, Ozan Öktem, Carola-Bibiane Schönlieb and Christian Etmann for the fruitful discussion about the procedure, and Felix Lucka and Jonas Adler for their ideas and helpful feedback on the simulation setup. Open Access funding enabled and organized by Projekt DEAL.

Author contributions

All authors worked on the concept and simulation setup of the dataset. J.L. and M.S. wrote the simulation scripts and the main parts of the manuscript. D.O. performed and documented the model-based technical validation of the dataset. All authors reviewed, finalised and approved the manuscript.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to J.L. or M.S.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Paper 2

Computed tomography reconstruction using deep image prior and learned reconstruction methods

Inverse Problems

Computed tomography reconstruction using deep image prior and learned reconstruction methods

To cite this article: Daniel Otero Baguer *et al* 2020 *Inverse Problems* **36** 094004

OPEN ACCESS
IOP Publishing
Inverse Problems
Inverse Problems **36** (2020) 094004 (24pp)
https://doi.org/10.1088/1361-6420/aba415

Computed tomography reconstruction using deep image prior and learned reconstruction methods

Daniel Otero Baguer, Johannes Leuschner and Maximilian Schmidt

Center for Industrial Mathematics (ZeTeM), University of Bremen, Bibliothekstraße 5, 28359 Bremen, Germany

E-mail: {otero,jleuschn,schmidt4}@uni-bremen.de

Received 12 March 2020, revised 2 July 2020
Accepted for publication 8 July 2020
Published 2 September 2020

Abstract

In this paper we describe an investigation into the application of deep learning methods for low-dose and sparse angle computed tomography using small training datasets. To motivate our work we review some of the existing approaches and obtain quantitative results after training them with different amounts of data. We find that the learned primal-dual method has an outstanding performance in terms of reconstruction quality and data efficiency. However, in general, end-to-end learned methods have two deficiencies: (a) a lack of classical guarantees in inverse problems and (b) the lack of generalization after training with insufficient data. To overcome these problems, we introduce the deep image prior approach in combination with classical regularization and an initial reconstruction. The proposed methods achieve the best results in the low-data regime in three challenging scenarios.

Keywords: inverse problems, deep learning, computed tomography, deep image prior, neural networks

(Some figures may appear in colour only in the online journal)

1. Introduction

Deep learning approaches to solving ill-posed inverse problems currently achieve state-of-the-art reconstruction quality. However, they require large amounts of training data, i.e., pairs of ground truths and measurements, and it is not clear how much is necessary to be able to

achieve good generalization. For ill-posed inverse problems arising in medical imaging, such as magnetic resonance imaging (MRI), guided positron emission tomography (PET), magnetic particle imaging, or computed tomography (CT), obtaining such high amounts of training data is challenging. In particular, ground truth data is difficult to obtain as it is impossible to take a photograph of the inside of the human body. What learned methods usually consider as ground truths are phantoms or high-dose reconstructions obtained with classical methods, such as filtered back-projection (FBP). These methods work well when using a large amount of low-noise measurements. In MRI, it is possible to obtain these reconstructions, but the data acquisition process requires a great deal of time. Therefore, one potential benefit of learned approaches in MRI is the reduction of data acquisition times [30]. In other applications such as CT, it would be necessary to expose patients to high doses of x-ray radiation to obtain the required training ground truths.

There is another approach called deep image prior (DIP) [31] that also uses deep neural networks, for example, a U-Net [45]. However, there is a remarkable difference: the DIP does not need any learning, i.e., the weights of the network are not trained. This approach seems to have low applicability because it requires a lot of time for image reconstruction, in contrast to learned methods. In the applications initially considered, for example, inpainting, denoising, and super-resolution, it is much easier to obtain or simulate data, which allows for the use of learned methods, and the DIP does not seem to have an advantage.

In this paper, we aim to explore the application of the DIP together with other deep learning methods for obtaining CT reconstructions when little training data is available. The structure of the paper and the main contributions are organized as follows. In section 2, we briefly describe the CT reconstruction problem. Section 3 provides a summary of related articles and approaches, together with some background and observations that we use as motivation for our work. In section 4, we introduce the combination of the DIP with classical regularization methods and discuss under which assumptions the classical regularization results still hold. In section 5, we propose a similar approach to the DIP but using an initial reconstruction given by any end-to-end learned method. Finally, in section 6, we present a benchmark of the different methods that we have analyzed using varying amounts of data from two standard datasets.

2. CT

CT is one of the most valuable technologies in modern medical imaging [9]. It allows for a non-invasive acquisition of the inside of the human body using x-rays. Since the introduction of CT in the 1970s, technical innovations such as new scan geometries have extended the limits on speed and resolution. Current research focuses on reducing the amount of potentially harmful radiation to which a patient is exposed during the scan [9]. These innovations include making measurements using lower intensity x-rays or at fewer angles. Both approaches introduce particular challenges for reconstruction methods that can severely reduce the image quality. In our work, we compare several reconstruction methods in these low-dose scenarios for a basic 2D parallel beam geometry (cf figure 1).

In this case, the forward operator is given by the 2D Radon transform [43] and models the attenuation of the x-ray when passing through a body. We can parameterize the path of an x-ray beam by the distance from the origin $s \in \mathbb{R}$ and angle $\varphi \in [0, \pi]$:

$$L_{s,\varphi}(t) = s\omega(\varphi) + t\omega^{\perp}(\varphi), \quad \omega(\varphi) := [\cos(\varphi), \sin(\varphi)]^{\mathrm{T}}. \tag{1}$$

The Radon transform then calculates the integral along the line for parameters s and φ:

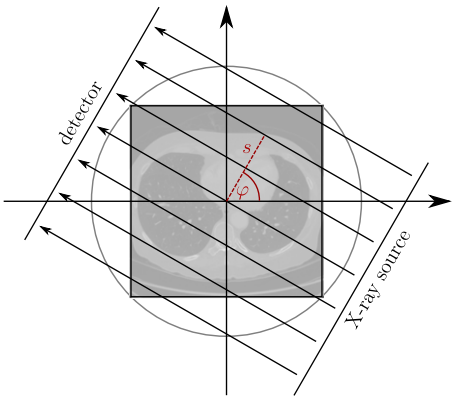

Figure 1. Parallel beam geometry.

$$Ax(s,\varphi) = \int_{\mathbb{R}} x\left(L_{s,\varphi}(t)\right)\ \mathrm{d}t. \tag{2}$$

According to Beer–Lambert's law, the result is the logarithm of the ratio of the intensity, I_0, at the x-ray source to the intensity, I_1, at the detector

$$Ax(s,\varphi) = -\ln\left(\frac{I_1(s,\varphi)}{I_0(s,\varphi)}\right) = y(s,\varphi)\,. \tag{3}$$

Calculating the transform for all pairs (s,φ) results in a so-called *sinogram*, which we also call an observation. To get a reconstruction $\hat{x}$ from the sinogram, we have to invert the forward model. Since the Radon transform is linear and compact, the inverse problem is *ill-posed* in the sense of Nashed [39, 40].

3. Related approaches and motivation

In this section, we first review and describe some of the existing data-driven and classical methods for solving ill-posed inverse problems, that have also been applied to obtain CT reconstructions. Following this, we review the DIP approach and related works.

In inverse problems one aims at obtaining an unknown quantity, in this case the image of the interior of the human body, from indirect measurements that frequently contain noise [16, 36, 44]. The problem is modeled by an operator $A : X \to Y$ between Banach or Hilbert spaces X and Y and the measured noisy data or observation:

$$y^\delta = Ax^\dagger + \tau. \tag{4}$$

The aim is to obtain an approximation $\hat{x}$ for $x^\dagger$ (the true solution), where τ, with $\|\tau\| \leqslant \delta$, describes the noise in the measurement.

Classical approaches to solving inverse problems include linear pseudo inverses given by filter functions [36] or non-linear regularized inverses given by the variational approach

$$\mathcal{T}_\alpha(y^\delta) \in \arg\min_{x\in\mathcal{D}} \mathcal{S}(Ax,\, y^\delta) + \alpha J(x), \tag{5}$$

where $\mathcal{S} : Y \times Y \to \mathbb{R}$ is the data discrepancy, $J : X \to \mathbb{R} \cup \{\infty\}$ is the regularizer, $\mathcal{D} := \mathcal{D}(A) \cap \mathcal{D}(J)$ and $\mathcal{D}(A)$, $\mathcal{D}(J)$ are the domains of A and J respectively. Examples of hand-crafted

regularizers/priors are $\|x\|^2$, $\|x\|_1$ and total variation (TV). The value of the regularization parameter α should be carefully selected. One way to do that, in the presence of a validation dataset with some ground truth and observation pairs, is to do a line-search and select the α that yields the best performance on average, assuming there is a uniform noise level. Given validation data $\{x_i^\dagger,\ y_i^\delta\}_{i=1}^N$, the data-driven parameter choice would be

$$\hat{\alpha} := \arg\max_{\alpha \in \mathbb{R}_+} \sum_{i=1}^{N} \ell(\mathcal{T}_\alpha(y_i^\delta),\ x_i^\dagger), \tag{6}$$

where $\ell : X \times X \to \mathbb{R}$ is some similarity measure, such as peak signal-to-noise ratio (PSNR) or structural self-similarity (SSIM).

Data-driven regularized inversion methods for solving inverse problems in imaging have recently had great success in terms of reconstruction quality [6]. Three main classes of methods are: end-to-end learned methods [1, 3, 8, 21, 28, 46], learned regularizers [34, 37] and generative networks [2, 7, 13]. For the study described in this paper, we only focus on the end-to-end learned methods.

3.1. End-to-end learned methods

In this section, we briefly review some of the most successful end-to-end learned methods. Most of them were implemented and included in our benchmark.

3.1.1. Post-processing. This method aims at improving the quality of the FBP reconstructions from noisy or few measurements by applying learned post-processing. Recent works [11, 28, 42, 48] have successfully used a convolutional neural network (CNN), such as the U-Net [45], to remove artifacts from FBP reconstructions. In mathematical terms, given a possibly regularized FBP operator $\mathcal{T}_{\text{FBP}}$, the reconstruction is computed using a network $D_\theta : X \to X$ as

$$\hat{x} := [D_\theta \circ \mathcal{T}_{\text{FBP}}](y^\delta) \tag{7}$$

with parameters θ of the network that are learned from data.

3.1.2. Fully learned. Methods of this type aim at directly learning the inversion process from data while keeping the network architecture as general as possible. This idea was successfully applied in MRI by the AUTOMAP architecture [49]. The main building blocks consist of fully connected layers. Depending on the problem, the number of parameters can grow quickly with the data dimension. For mapping from sinogram to reconstruction in the LoDoPaB-CT dataset [32] (see section 6.1), such a layer would have over $1000 \times 513 \times 362^2 \approx 67 \times 10^9$ parameters. This makes the naive approach infeasible for large CT data.

He *et al* [22] introduced an adapted two-part network, called iRadonMap. The first part reproduces the structure of the FBP. A fully connected layer is applied along s and shared over the rotation angle dimension φ, playing the role of the filtering. For each reconstruction pixel (i, j) only sinogram values on the sinusoid $s = i\cos(\varphi) + j\sin(\varphi)$ have to be considered and are multiplied by learned weights. For the example above, the number of parameters in this layer reduces to $513^2 + 362^2 \times 1000 \approx 13 \times 10^7$. The second part consists of a post-processing network. We choose the U-Net architecture for our experiments, which allows for a direct comparison with the FBP + U-Net approach.

3.1.3. Learned iterative schemes. Another series of works [1, 3, 20, 21] use CNNs to improve iterative schemes commonly used in inverse problems for solving (5), such as gradient descent, proximal gradient descent or hybrid primal-dual algorithms. For example, the proximal gradient descent is given by the iteration

$$x^{(k+1)} = \phi_{J,\,\alpha,\,\lambda_k}(x^{(k)} - \lambda_k A^*(Ax^{(k)} - y^\delta)), \tag{8}$$

for $k = 0, \ldots, L-1$, where $\phi_{J,\alpha,\lambda} : X \to X$ is the proximal operator or projector. In [20], the authors replace the projector by a CNN that is trained to project perturbed reconstructions to the set of clean reconstructions. However, this approach is not end-to-end because the network is first trained to do the projection and then inserted into the iterative scheme.

The idea behind end-to-end learned iterative methods is to unroll these schemes with a small number of iterations, and replace some operators by CNNs with parameters that are trained using ground truth and observation data pairs. Each iteration is performed by a convolutional network ψ_{θ_k} that includes the gradients of the data discrepancy and of the regularizer as input in each iteration. Moreover, the number of iterations is fixed and small, e.g., $L = 10$. The reconstruction operator is given by $\mathcal{T}_\theta : Y \to X$ with $\mathcal{T}_\theta(y^\delta) = x^{(L)}$ and

$$\begin{aligned} x^{(k+1)} &= \psi_{\theta_k}(x^{(k)},\ A^*(Ax^{(k)} - y^\delta), \nabla J(x^{(k)})) \\ x^{(0)} &= A^+(y^\delta) \end{aligned}$$

for any pseudo inverse A^+ of the operator A and $\theta = (\theta_0, \ldots, \theta_{L-1})$. Alternatively, $x^{(0)}$ could be just randomly initialized.

Similarly, more sophisticated algorithms, such as hybrid primal-dual algorithms, can be unrolled and trained in the same fashion. In this work, we used an implementation of the learned gradient descent [1] and the learned primal-dual method [3].

The above mentioned approaches all rely on a parameterized operator $\mathcal{T}_\theta : Y \to X$, whose parameters θ are optimized using a training set of N ground truth samples $x_i^\dagger$ and their corresponding noisy observations y_i^δ. Usually, the empirical mean squared error is minimized, i.e.,

$$\hat{\theta} \in \arg\min_{\theta \in \Theta} \frac{1}{N} \sum_{i=1}^{N} \|\mathcal{T}_\theta(y_i^\delta) - x_i^\dagger\|^2. \tag{9}$$

After training, the reconstruction $\hat{x} \in X$ from a noisy observation $y^\delta \in Y$ is given by $\hat{x} = \mathcal{T}_{\hat{\theta}}(y^\delta)$. The main disadvantage of most of these approaches is that they do not enforce data consistency. As a consequence, some information in the observation could be ignored, yielding a result that might lack important features of the image. In medical imaging, this is critical since it might remove an indication of a lesion. Recent works [4, 19] also show that some methods, such as those which are fully learned or follow the post-processing approach, are unstable, which means that tiny perturbations in the ground truth or the measurements may result in severe artifacts in the reconstructions. These are the main motivations for the approach we introduce in section 5. Nevertheless, there exist other methods [46] that do enforce data consistency and may not suffer from these instabilities.

3.2. DIP

The DIP is similar to the generative networks approach and the variational method. However, instead of having a regularization term $J(x)$, the regularization is incorporated by the reparametrization $x = \varphi(\theta, z)$, where φ is a deep generative network, for example a U-Net,

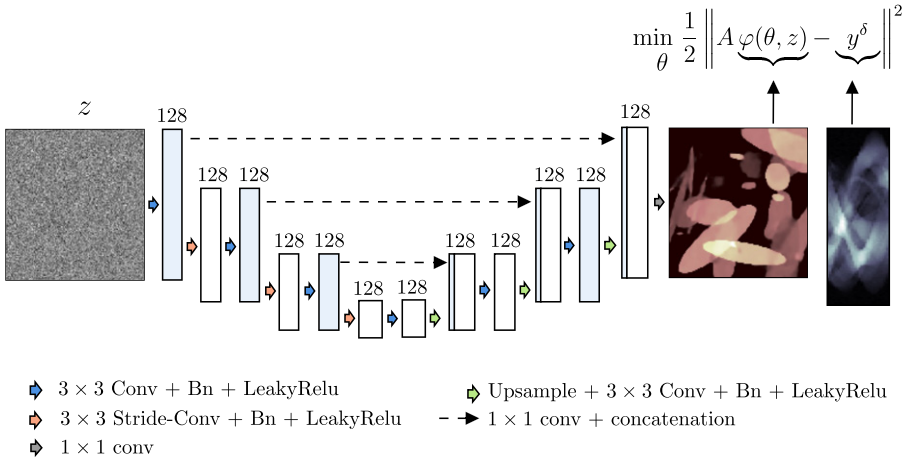

Figure 2. The figure illustrates the DIP approach. We use a U-Net architecture with 128 channels at every layer. Some layers have additionally the skip channels (coming from the dashed arrows). We always use either 4 or 0 skip channels.

Figure 3. Intermediate reconstructions of the DIP approach for CT (ellipses dataset, see section 6.2). At the beginning the coefficients are randomly initialized from a prior distribution. The method starts reconstructing the image from global to local details.

with randomly initialized weights $\theta \in \Theta$, and z is a fixed input such as random white noise. The approach is depicted in figure 2 and consists in solving

$$\hat{\theta} \in \arg\min_{\theta \in \Theta} \|A\varphi(\theta, z) - y^\delta\|^2, \qquad \hat{x} := \varphi(\hat{\theta}, z). \tag{10}$$

The weights are optimized by a gradient descent method to minimize the data discrepancy of the output of the network. In the original method, the authors use gradient descent with early stopping to avoid reproducing noise. This is necessary due to the overparameterization of the network, which makes it able to reproduce the noise. The regularization is a combination of early stopping (similar to the Landweber iteration) and the architecture [14]. The drawback is that it is not clear how to choose when to stop. In the original work, the authors do this using a validation set and select the number of iterations that performs best on average in terms of PSNR.

The prior is related to the implicit structural bias of this kind of deep convolutional networks. In the original DIP paper [31] and more recently in [10, 24], it is shown that convolutional image generators, optimized with gradient descent, fit *natural* images faster than noise and learn to construct them from low to high frequencies. This effect is illustrated in figure 3.

3.2.1. Related work. The DIP approach has inspired many other researchers to improve it by combining it with other methods [35, 38, 47], to use it for a wide range of applications [17, 18, 26, 27] and to offer different perspectives and explanations of why it works [10, 12, 14]. In [38], the concept of regularization by denoising (RED) is introduced and it is shown how the two (DIP and RED) can be merged into a highly effective unsupervised recovery process. Another series of works also adds explicit priors but on the weights of the network. In [47], this is done in the form of a multi-variate Gaussian but learning the covariance matrix and the mean using a small dataset. In [12], a Bayesian perspective on the DIP is introduced by also incorporating a prior on the weights θ and conducting the posterior inference using stochastic gradient Langevin dynamics.

So far, the DIP has been used for denoising, inpainting, super-resolution, image decomposition [17], compressed sensing [47], PET [18], MRI [27] among other applications. A similar idea [26] was also used for structural optimization, which is a popular method for designing objects such as bridge trusses, airplane wings, and optical devices. Rather than directly optimizing densities on a grid, they instead optimize the parameters of a neural network which outputs those densities.

3.2.2. Network architecture. In the paper by Ulyanov *et al* [31], several architectures were considered, for example, ResNet [23], encoder–decoder (autoencoder) and a U-Net [45]. For inpainting large regions, the Autoencoder with depth $= 6$ performed best, whereas for denoising a modified U-Net achieved the best results. The regularization happens mainly due to the architecture of the network, which reduces the search space but also influences the optimization process to find more *natural* images. Therefore, for each application, it is crucial to choose the appropriate architecture and to tune hyper-parameters, such as the network's depth and the number of channels per layer. Optimizing the hyper-parameters is the most time-consuming part. In figure 4 we show some reconstructions from the ellipses dataset (see section 6.2) with different hyper-parameter choices. In this case, it seems that the U-Net without skip connections and depth 5 (encoder–decoder) achieves the best performance. One can see that when the number of channels is too low, the network does not have enough representation power. Also, if there are no skip channels, the higher the number of scales (equivalent to the depth), the more the regularization effect. The extraordinary success of this approach demonstrates that the architecture of the network has a significant influence on the performance of deep learning approaches that use similar kinds of networks.

3.2.3. Early-stopping. As mentioned earlier, in [31], it is shown that early stopping has a positive impact on the reconstruction results. It was observed that in some applications, such as denoising, the loss decreases rapidly toward *natural* images, but takes much more time to go toward noisy images. This empirical observation helps to determine when to stop. In figure 5, one can observe how the similarity with respect to the ground truth (measured by the PSNR and the SSIM metrics) reaches a maximum and then deteriorates during the optimization process.

4. DIP and classical regularization

In this section we analyze the DIP in combination with classical regularization, i.e., we include a regularization term $J : X \to \mathbb{R} \cup \{\infty\}$, such as TV. We give necessary assumptions under which we are able to obtain standard guarantees in inverse problems, such as existence of a solution, convergence, and convergence rates.

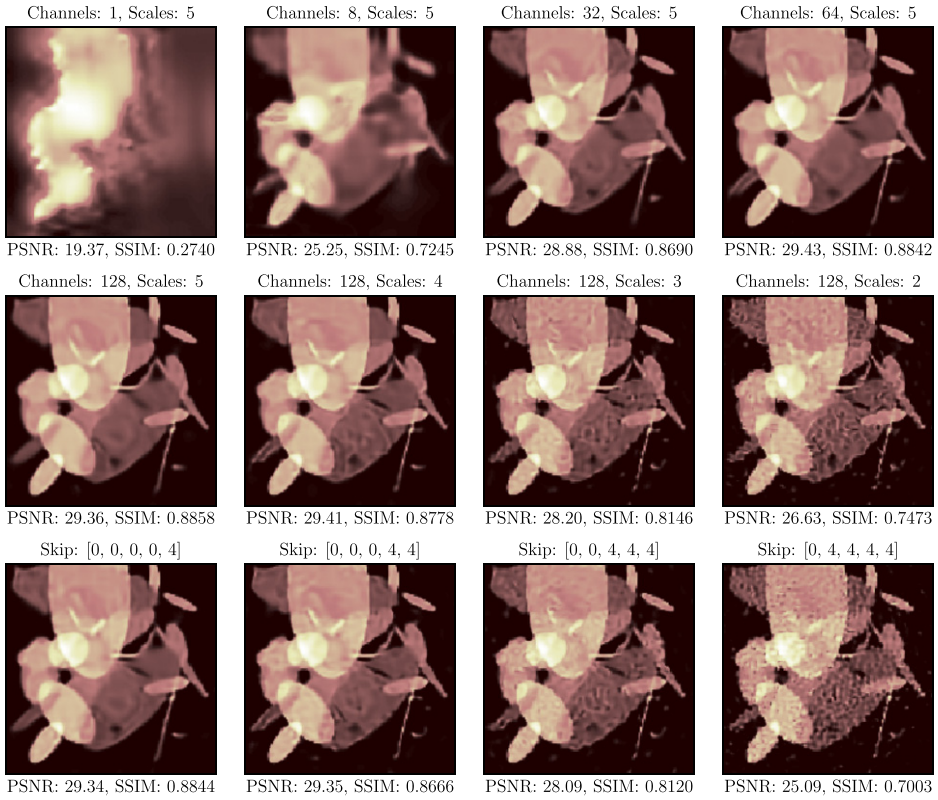

Figure 4. CT reconstructions after 5000 iterations using the DIP with a U-Net architecture and different scales (depths), channels per layer (the network has the same number of channels at every layer) and number of skip connections (the first two rows do not use skip connections, i.e., skip: $[0, 0, 0, 0, 0]$). In the last row all reconstructions use 5 scales and 128 channels.

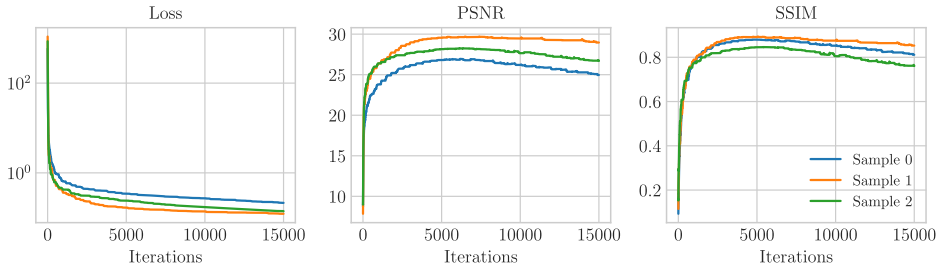

Figure 5. Training loss and true similarity (PSNR and SSIM) of CT reconstructions using the DIP approach. The training was done over 15 000 iterations and the architecture is an encoder–decoder (no skip channels) with 5 scales and 128 channels per layer.

In the general case, we consider X and Y to be Banach spaces, and $A : X \to Y$ a continuous linear operator. To simplify notation, we use $\varphi(\cdot)$ instead of $\varphi(\cdot, z)$, since the input to the network is fixed. Additionally, we assume that Θ is a Banach space, and $\varphi : \Theta \to X$ is a continuous mapping.

The proposed method aims at finding

$$\theta_\alpha^\delta \in \arg\min_{\theta\in\Theta} \mathcal{S}(A\varphi(\theta), y^\delta) + \alpha J(\varphi(\theta)) \quad \text{for} \quad \alpha > 0 \tag{11}$$

to obtain

$$\mathcal{T}_\alpha(y^\delta) := \varphi(\theta_\alpha^\delta). \tag{12}$$

With this approach, we eliminate the need for early stopping, i.e., the need to find an optimal number of iterations. However, we introduce the problem of finding an optimal α, which is a classical issue in inverse problems. These problems are similar since both choices depend on the noise level of the observation data. The higher the noise, the higher the value of α or the smaller the number of iterations for obtaining optimal results.

If the range of φ is $\Omega := \mathrm{rg}(\varphi) = X$, i.e.,

$$\forall\ x \in X:\ \exists\ \theta \in \Theta \text{ s.t. } \varphi(\theta) = x; \tag{13}$$

this is equivalent to the standard variational approach in equation (5). However, although the network can fit some noise, it cannot fit, in general, any arbitrary $x \in X$. This depends on the chosen architecture, and it is mainly because we do not use any fully connected layers. Nevertheless, the minimization in (11) is similar to the setting in equation (5) if we restrict the domain of A to be $\widetilde{\mathcal{D}}(A) := \mathcal{D}(A) \cap \Omega$

$$\mathcal{T}_\alpha(y^\delta) \in \arg\min_{x\in\widetilde{\mathcal{D}}} \mathcal{S}(Ax,\, y^\delta) + \alpha J(x), \tag{14}$$

where $\widetilde{\mathcal{D}} := \widetilde{\mathcal{D}}(A) \cap \mathcal{D}(J)$. If the following assumptions are satisfied, then all the classical theorems, namely well-posedness, stability, convergence, and convergence rates, still hold, see [25].

Assumption 1. The range of φ with respect to θ (parameters of the network), namely Ω, is closed, i.e., if there is a convergent sequence $\{x_k\} \subset \Omega$ with limit $\tilde{x}$, it holds $\tilde{x} \in \Omega$.

Definition 1. An element $x^\dagger \in \widetilde{\mathcal{D}}$ is called a J-minimizing solution if $Ax^\dagger = y^\dagger$ and $\forall\, x \in \widetilde{\mathcal{D}}:\ J(x^\dagger) \leqslant J(x)$, where $y^\dagger$ is the perfect noiseless data.

Assumption 2. There exists a J-minimizing solution $x^\dagger \in \widetilde{\mathcal{D}}$ and $J(x^\dagger) < \infty$.

Assumption 1 guarantees that the restricted domain of A is closed, whereas assumption 2 guarantees that there is a J-minimizing solution in the restricted domain. In appendix A, we analyze in which cases these conditions hold.

5. DIP with initial reconstruction

In this section, we propose a two-steps approach based on the method from the previous section. The idea is to take the result from any end-to-end learned method $\mathcal{T} : Y \to X$ as initial reconstruction (first step) and further enforce data consistency by optimizing over its deep-neural parameterization (second step).

Definition 2 (Deep-neural parameterization). Given an untrained network $\varphi : \Theta \times Z \to X$ and a fixed input $z \in Z$, the deep-neural parameterization of an element $x \in X$ with respect to φ and z is

$$\theta_x \in \arg\min_{\theta\in\Theta} \|\varphi(\theta,\, z) - x\|^2. \tag{15}$$

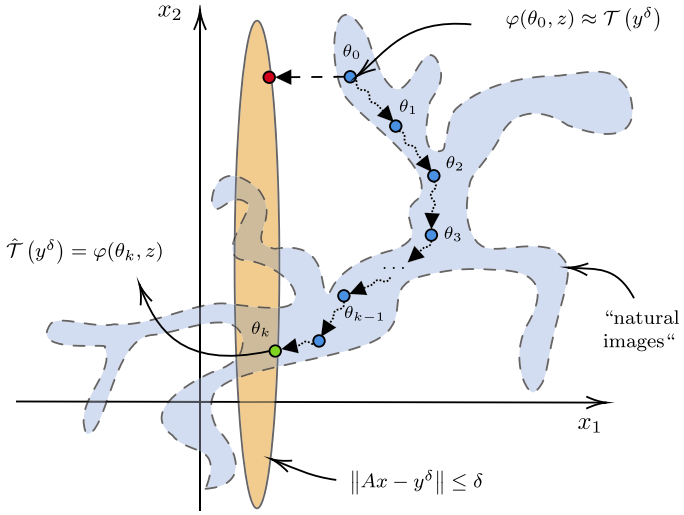

Figure 6. Graphical illustration of the DIP approach with initial reconstruction. The blue area refers to an approximation of some part of the space of *natural* images.

Algorithm 1. DIP with initial reconstruction.

1: $x_0 \leftarrow \mathcal{T}(y^\delta)$
2: $z \leftarrow$ noise
3: $\theta_0 \in \arg\min_\theta \|\varphi(\theta, z) - x_0\|^2$
4: **for** $k \leftarrow 0$ to $K - 1$ **do**
5: $\quad \omega \in \partial\mathcal{L}(\theta_k)$
6: $\quad \theta_{k+1} \leftarrow \theta_k - \eta\omega$
7: **end for**
8: $\hat{\mathcal{T}}(y^\delta) \leftarrow \varphi(\theta_k, z)$

The projection onto the range of the network is possible because of the result of assumption 1, i.e., the range is closed. If φ is a deep convolutional network, for example, a U-Net, the deep-neural parameterization has similarities with other signal representations, such as the Wavelets and Fourier transforms [26]. For image processing, such domains are usually more convenient than the classical pixel representation.

As shown in figure 6, one way to enforce data consistency is to project the initial reconstruction into the set where $\|Ax - y^\delta\| \leqslant \delta$. The puzzle is that due to the ill-posedness of the problem, the new solution (red point) will very likely have artifacts. The proposed approach first obtains the deep-neural parameterization θ_0 of the initial reconstruction $\mathcal{T}(y^\delta)$ and then use it as starting point to minimize

$$\mathcal{L}(\theta) := \|A\varphi(\theta, z) - y^\delta\|^2 + \alpha J(\varphi(\theta, z)), \tag{16}$$

over θ via gradient descent. The iterative process is continued until $\|A\varphi(\theta, z) - y^\delta\| \leqslant \delta$ or

for a given fixed number of iterations K determined by means of a validation dataset. This approach seems to force the reconstruction to stay close to the set of *natural* images because of the structural bias of the deep-neural parameterization. The procedure is listed in algorithm 1 and a graphical representation is shown in figure 6.

The new method $\hat{\mathcal{T}} : Y \to X$ is similar to other image enhancement approaches. For example, related methods [15] first compute the wavelet transform (parameterization), and then repeatedly perform smoothing or shrinking of the coefficients (further optimization).

6. Benchmark setup and results

For the benchmark, we implemented the end-to-end learned methods described in section 3.1. We trained them on different data sizes and compared them with classical methods, such as FBP and TV regularization, and with the proposed methods. The datasets we use were recently released to benchmark deep learning methods for CT reconstruction [32]. They are accessible through the DIV$\alpha\ell$ python library [33]. We also provide the code and the trained methods in the following GitHub repository: https://github.com/oterobaguer/dip-ct-benchmark.

6.1. The LoDoPaB-CT dataset

The low-dose parallel beam (LoDoPaB) CT dataset [32] consists of more than 40 000 two-dimensional CT images and corresponding simulated low-intensity measurements. Human chest CT reconstructions from the LIDC/IDRI database [5] are used as virtual ground truth. Each image has a resolution of 362×362 pixels. For the simulation setup, a simple parallel beam geometry with 1000 angles and 513 projection beams is used. To simulate low intensity, Poisson noise is applied to the projection data. The noise amount corresponds to an x-ray source that on average emits 4096 photons per detector pixel. We use the standard dataset split defining in total 35 820 training pairs, 3522 validation pairs and 3553 test pairs. In addition, we analyze another dataset, LoDoPaB (200), obtained by uniformly sampling 200 angles from the original 1000 without any further modification.

6.2. Ellipses dataset

As a synthetic dataset for imaging problems, random phantoms of combined ellipses are commonly used. We use the `'ellipses'` standard dataset from the DIV$\alpha\ell$ python library (as provided in version 0.4) [33]. The images have a resolution of 128×128 pixels. Measurements are simulated with a parallel beam geometry with only 30 angles and 183 projection beams. In addition to the sparse-angle setup, moderate Gaussian noise with a standard deviation of 2.5% of the mean absolute value of the projection data is added to the projection data. In total, the training set contains 32 000 pairs, while the validation and test set consist of 3200 pairs each.

6.3. Implementation details

For the DIP with initial reconstruction, we used the learned primal-dual, which has the best performance among the compared methods (see the results in figure 7). For each data size, we

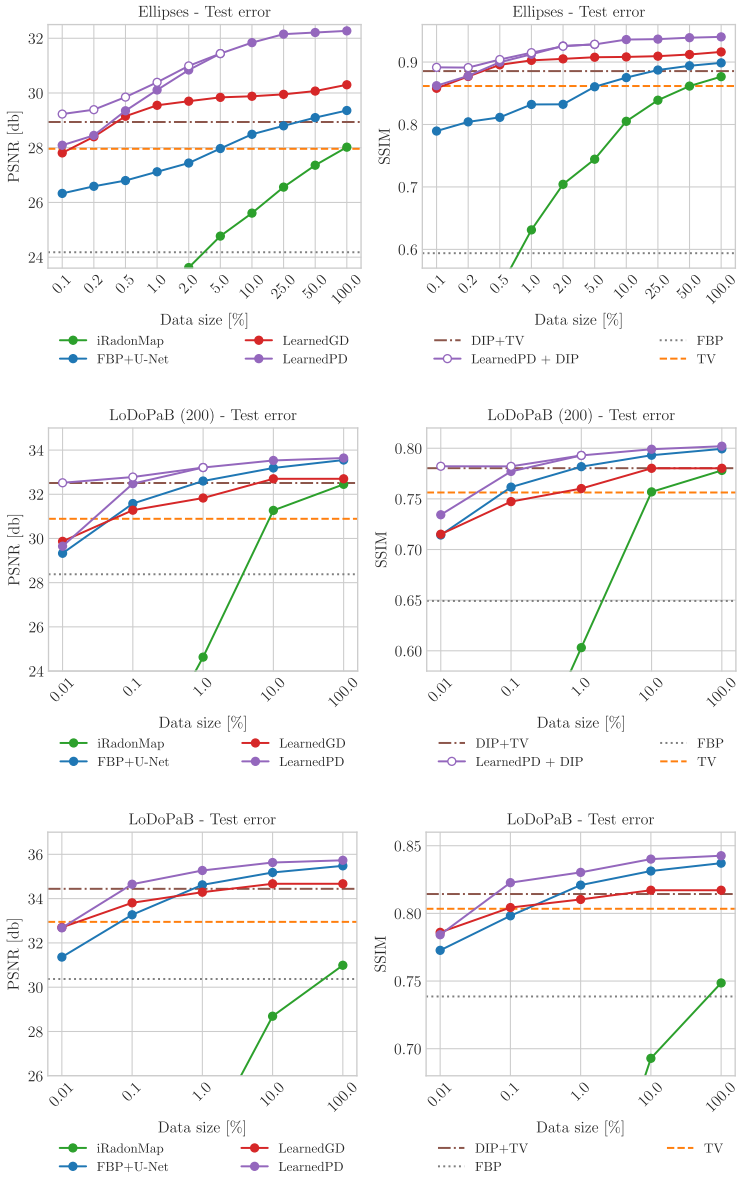

Figure 7. Benchmark results of several existing methods and the proposed approaches (DIP + TV, learned primal-dual + DIP) on the Ellipses, LoDoPaB (200) and LoDoPaB datasets. The horizontal lines indicate the performance of data-free methods.

chose different hyper-parameters, namely the step-size η, the TV regularization parameter α, and the number of iterations K, based on the available validation dataset.

Minimizing $\mathcal{L}(\theta)$ in (16) is not trivial because TV is not differentiable. In our implementation we use the PyTorch automatic differentiation framework [41] and the ADAM [29] optimizer. For the Ellipses dataset we use the ℓ_2-discrepancy term, whereas for LoDoPaB we use the Poisson loss.

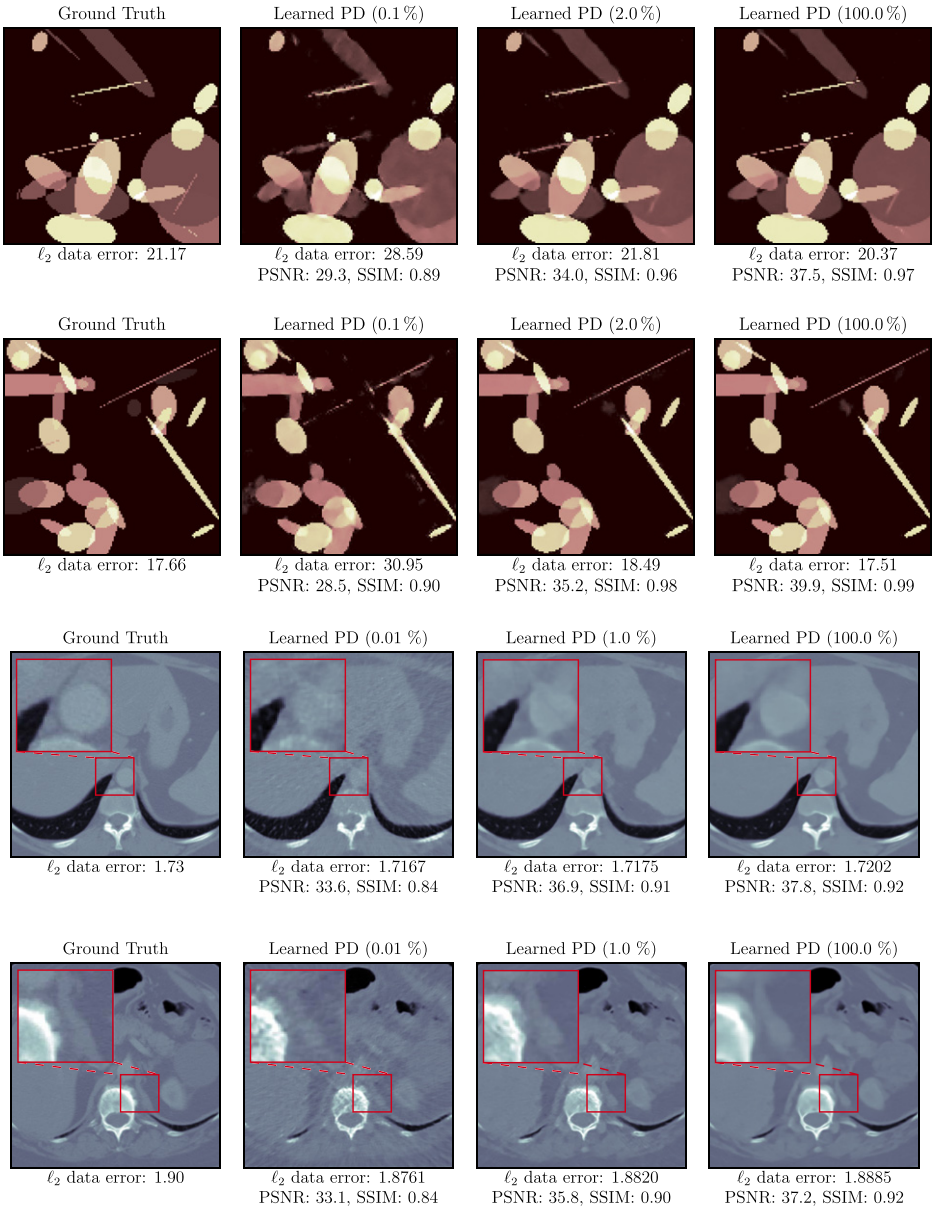

Figure 8. Reconstructions of test samples using the learned primal-dual method trained with different amounts of data from the ellipses and LoDoPaB datasets. The ℓ_2 data error measures the discrepancy between the noisy observation and the noise-free projection of the (reconstructed) image.

6.4. Numerical results

We trained all the methods with different dataset sizes. For example, 0.1% on the ellipses dataset means we trained the model with 0.1% (32 data-pairs) of the available training data and 0.1% (3 data-pairs) of the validation data. Afterward, we tested the performance of the method

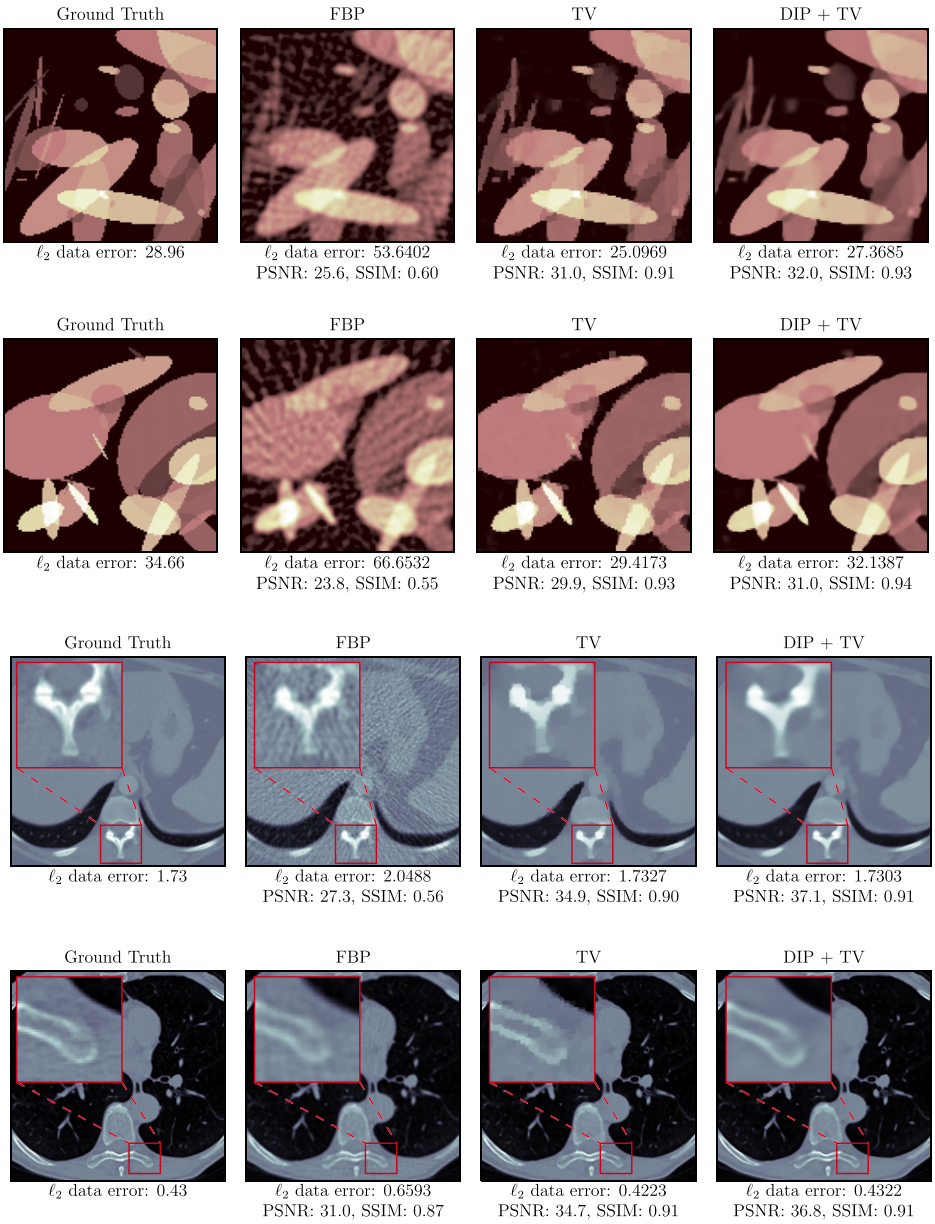

Figure 9. Reconstruction obtained with the FBP method, isotropic TV regularization and the DIP approach combined with TV, for test samples from the ellipses and LoDoPaB datasets. The ℓ_2 data error measures the discrepancy between the noisy observation and the noise-free projection of the (reconstructed) image.

on the first 100 samples of the test dataset (in the original order, i.e., not sorted by patient). This reduced test dataset was used because some of the methods require a lot of time for reconstruction, and the mean performance on 100 samples already allows for accurate benchmarking. The results are depicted in figure 7 and more details can be found in appendix B.

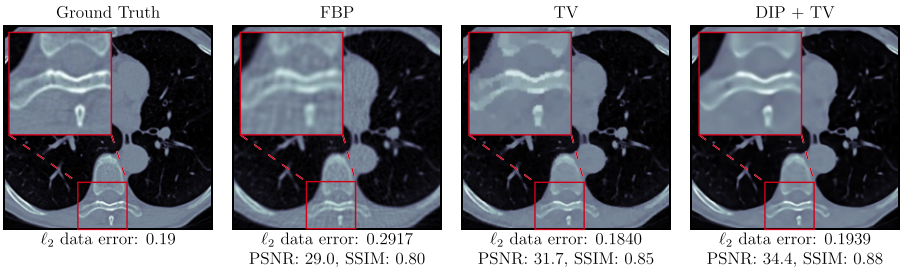

Figure 10. Reconstruction obtained with the FBP method, isotropic TV regularization and the DIP approach combined with TV, for test samples from the LoDoPaB (200) dataset. The ℓ_2 data error measures the discrepancy between the noisy observation and the noise-free projection of the (reconstructed) image.

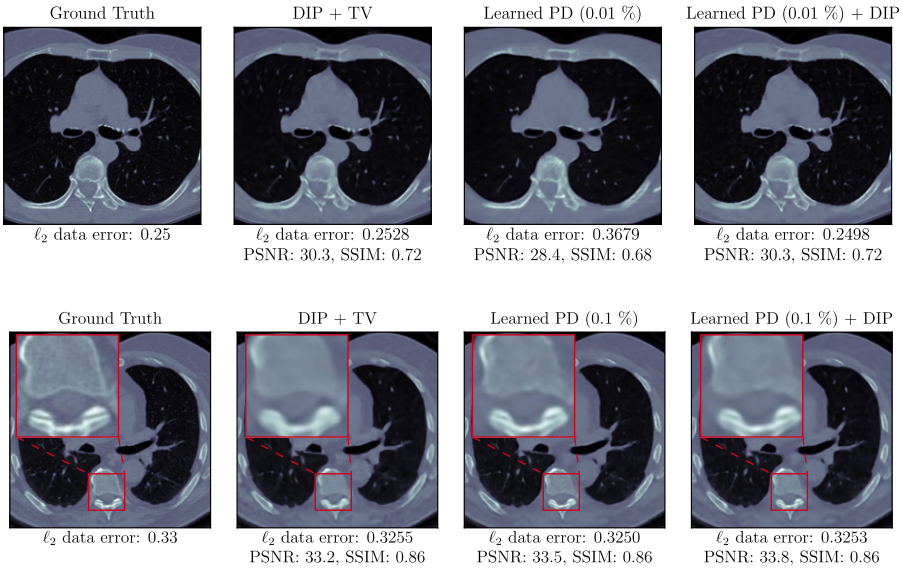

Figure 11. Examples of reconstructions obtained with the DIP + TV approach, the learned primal-dual method trained with 0.01% and 0.1% of the LoDoPaB (200) dataset and the DIP + TV approach with initial reconstruction. The ℓ_2 data error measures the discrepancy between the noisy observation and the noise-free projection of the (reconstructed) image.

As expected, the fully learned method (iRadonMap) requires a large amount of data to achieve acceptable performance. On the ellipses and LoDoPaB (200) dataset, it outperformed TV using 100% of the data, whereas on the LoDoPaB dataset, it performed just slightly better than the FBP. The learned post-processing (FBP + U-Net) required much less data. It outperformed TV with only 10% of the ellipses dataset and 0.1% of the LoDoPaB dataset. On the other hand, we find that the learned primal-dual is very data efficient and achieved

Figure 12. Examples of reconstructions obtained with the DIP approach combined with TV, the learned primal-dual method trained with 0.1% and 0.2% of the Ellipses dataset (32 and 64 resp. data-pairs) and the DIP approach with initial reconstruction. The ℓ_2 data error measures the discrepancy between the noisy observation and the noise-free projection of the (reconstructed) image.

the best performance. In figure 8, we show some results from the test set for different data sizes.

The DIP + TV approach achieved the best results among the data-free methods. On average, it outperforms TV by 1 dB on all the analyzed datasets. In figures 9 and 10, it can be observed that TV tends to produce flat regions but also produces high staircase effects on the edges. On the other hand, the combination with DIP produces more realistic edges. For the first two smaller data sizes of the ellipses and LoDoPaB (200) datasets, it performs better than all the end-to-end learned methods.

The DIP + TV with initial reconstruction improved the results on the low-data regime for the ellipses and LoDoPaB (200) datasets. For the higher data sizes and the LoDoPaB dataset, it did not yield reconstructions with higher quality than those already obtained by the DIP + TV or learned primal-dual methods. We believe that this approach is more useful in the case of having sparse measurements and little training data.

In figures 11 and 12, we show some reconstructions obtained using this method for the LoDoPab (200) and ellipses datasets. The reconstructions have a better data consistency with respect to the observed data (ℓ_2-discrepancy) and higher quality both visually and in terms of the PSNR and SSIM measures. Moreover, this approach is in general much faster, even if we also consider the iterations required to obtain the deep-prior/neural parameterization of the first reconstruction. These initial iterations are much faster because they only use the identity operator instead of the Radon transform. For example, for the Ellipses dataset, the DIP + TV approach needs 8000 iterations to obtain optimal performance in a validation dataset (five ground truth and observation pairs). On the other hand, by using the initial reconstruction, it needs 4000 iterations with the identity operator and only 1000 with the Radon transform operator, which results in a $2\times$ speed factor.

7. Conclusions

In this work, we study the combination of classical regularization, deep-neural parameterization, and deep learning approaches for CT reconstruction. We benchmark the investigated methods and evaluate how they behave in low-data regimes. Among the data-free approaches, the DIP + TV method achieves the best results. However, it is considerably slow and does not benefit from having a small dataset with reference reconstructions. On the other hand, the learned primal-dual is very data efficient. However, it lacks data consistency when not trained with enough data. These issues motivate us to adjust the reconstruction obtained with the learned primal-dual to match the observed data. We solved the puzzle without introducing artifacts through a combination of classical regularization and the DIP.

The results presented in this paper offer several baselines for future comparisons with other approaches. Moreover, the proposed methods could be applied to other imaging modalities.

Acknowledgments

The authors acknowledge the support by the Deutsche Forschungsgemeinschaft (DFG) within the framework of GRK 2224/1 'π^3: Parameter Identification—Analysis, Algorithms, Applications'. The authors also thank Jonas Adler, Jens Behrmann, Sören Dittmer, Peter Maass and Michael Pidcock for useful comments and discussions.

Appendix A. DIP and classical regularization

The mapping $\varphi : \Theta \to X$ has a neural network structure, with a fixed input $z \in \mathbb{R}^{n_0}$, and can be expressed as a composition of affine mappings and activation functions:

$$\varphi = \sigma^{(L)} \circ \mathcal{K}^{(L)} \circ \cdots \circ \sigma^{(2)} \circ \mathcal{K}^{(2)} \circ \sigma^{(1)} \circ \mathcal{K}^{(1)}, \tag{A.1}$$

where $\mathcal{K}^{(i)}(x) := W^{(i)}x + b^{(i)}$, $W^{(i)} \in G^{(i)} \subseteq \mathbb{R}^{n_i \times n_{i-1}}$, $b^{(i)} \in B^{(i)} \subseteq \mathbb{R}^{n_i}$, $\sigma^{(i)} : \mathbb{R} \to \mathbb{R}$ (applied component-wise), and $\theta = (W^{(L)}, b^{(L)}, \ldots, W^{(1)}, b^{(1)}) \in G^{(L)} \times B^{(L)} \cdots \times G^{(1)} \times B^{(1)} = \Theta$. In the following we analyze under which conditions we can guarantee that the range of φ (with respect to Θ) is closed.

Definition 3. An activation function $\sigma : \mathbb{R} \to \mathbb{R}$ is valid, if it is continuous, monotone, and bounded, in the sense there exist $c > 0$ such that $\forall x \in X : |\sigma(x)| \leqslant c|x|$.

Lemma 1. *Let φ be a neural network $\varphi : \Theta \to X$ with L layers. If Θ is a compact set, and the activation functions $\sigma^{(i)}$ are* valid*, then the range of φ is closed.*

Proof. In order to prove the result, we show that the range after each layer of the network is compact.

(a) Let the set $V = \{Wu : \ W \in G \subset \mathbb{R}^{m \times n}, u \in U \subset \mathbb{R}^n\}$, where G and U are compact sets, i.e., bounded and closed. Since G and U are bounded, it follows that V is bounded.

Let the sequence $\{W^{(k)}u^{(k)}\}$, with $W^{(k)} \in G$ and $u^{(k)} \in U$, converge to v. Since $\{W^{(k)}\}$ and $\{u^{(k)}\}$ are bounded, there is a subsequence $\{\overline{W}^{(k)}\bar{u}^{(k)}\}$, where both $\{\overline{W}^{(k)}\}$ and $\{\bar{u}^{(k)}\}$ converge to $\overline{W} \in G$ and $\bar{u} \in U$ respectively. It follows that $\{\overline{W}^{(k)}\bar{u}^{(k)}\}$ converges to $\overline{W}\bar{u}$, therefore, $v = \overline{W}\bar{u} \in V$, which shows that V is closed. Thus, V is compact.

(b) From (a), the fact that $G^{(i)}$, $B^{(i)}$ are compact sets, and assuming $U^{(i)} \subset \mathbb{R}^{n_i - 1}$ is also compact, it follows that $V^{(i)} = \{Wu + b : \ W \in G^{(i)}, u \in U^{(i)}, b \in B^{(i)} \subset \mathbb{R}^{n_i}\}$ is compact.

(c) It is easy to show that if the pre-image of a *valid* activation σ is compact, then its image is also compact.

In the first layer, $U_0 = \{z\}$, which is compact; thus, using (a), (b), and (c) it can be shown by induction that the range of $\varphi : \Theta \to \Omega$ is closed. □

All activation functions commonly used in the literature, for example, sigmoid, hyperbolic tangent, and piece-wise linear activations, are *valid*. The bounds on the weights of the network can be ensured by clipping the weights after each gradient update.

Remark 1. An alternative condition to the bound on the weights is to use only *valid* activation functions with closed range, for example, ReLU or leaky ReLU. However, it wouldn't be possible to use sigmoid or hyperbolic tangent. In our experiments, we observed that having a sigmoid activation in the last layer of the DIP network performs better than having a ReLU.

Appendix B. Dataset details, hyper-parameters and results

In this appendix, we present all the hyper-parameters tables B3–B10 that were selected for the method using a validation set. The first two tables B1–B2 depict the number of samples used for training and validation in each case.

For the data-free baseline approaches, i.e. FBP and TV, we used 100 samples for selecting the optimal hyper-parameters. In the low-data regime this by far exceeds the number of samples

Table B1. Amounts of training and validation pairs from the ellipses dataset used for the benchmark in section 6.

%	0.1	0.2	0.5	1.0	2.0	5.0	10.0	25.0	50.0	100.0
#train	32	64	160	320	640	1600	3200	8000	16 000	32 000
#val	3	6	16	32	64	160	320	800	1600	3200

Table B2. Amounts of training and validation pairs from the LoDoPaB dataset used for the benchmark in section 6. The last two lines denote the numbers of patients of whom images are included.

%	0.01	0.1	1.0	10.0	100.0
#train	3	35	358	3582	35 820
#val	1	3	35	352	3522
#patients train	1	1	7	64	632
#patients val	1	1	1	6	60

Table B3. FBP hyper-parameters and results.

Dataset	Filter type	Low-pass cut-off	PSNR (dB)	SSIM
Ellipses	Hann	0.7051	24.18	0.5939
LoDoPaB (200)	Hann	0.5000	28.38	0.6492
LoDoPaB	Hann	0.6410	30.37	0.7386

Table B4. TV hyper-parameters and results. The step size is set to 10^{-3}.

Dataset	Loss function	α	PSNR (dB)	SSIM
Ellipses	ℓ_2	7.743×10^{-4}	27.84	0.8495
LoDoPaB (200)	Poisson	12.63	30.89	0.7563
LoDoPaB	Poisson	20.55	32.95	0.8034

Table B5. DIP + TV hyper-parameters and results. For all experiments the number of channels is set to 128 at every scale. For the output sigmoid activation is used.

Dataset	Loss func.	Scales	Skip channels	α	step size	PSNR (dB)	SSIM
Ellipses	ℓ_2	5	(0, 0, 0, 0, 0)	3.162×10^{-4}	1×10^{-3}	28.94	0.8855
LoDoPaB (200)	Poisson	6	(0, 0, 0, 0, 4, 4)	4.0	5×10^{-4}	32.51	0.7803
LoDoPaB	Poisson	6	(0, 0, 0, 0, 4, 4)	7.0	5×10^{-4}	34.44	0.8143

used by the learned approaches, leading to a slight bias of the comparison in favor of the data-free baseline approaches. For the DIP + TV we used at most 5 samples for validation and selection of hyper-parameters.

Table B6. DIP + TV (with initial reconstruction given by the learned primal-dual method). For all experiments the number of channels is set to 128 at every scale. For the output sigmoid activation is used.

Dataset	Data size (%)	Loss func.	Scales	Skip channels	α	PSNR (dB)	SSIM
Ellipses	0.1	ℓ_2	5	(0, 0, 0, 0, 0)	3.162×10^{-4}	29.23	0.8915
	0.2	ℓ_2	5	(0, 0, 0, 0, 0)	2.154×10^{-4}	29.39	0.8911
	0.5	ℓ_2	5	(0, 0, 0, 0, 0)	2.154×10^{-4}	29.85	0.904
	1.0	ℓ_2	5	(0, 0, 0, 0, 0)	2.154×10^{-4}	30.39	0.915
	2.0	ℓ_2	5	(0, 0, 0, 0, 0)	2.154×10^{-4}	30.99	0.9253
	5.0	ℓ_2	5	(0, 0, 0, 0, 0)	2.154×10^{-4}	31.44	0.9285
	10.0	ℓ_2	5	(0, 0, 0, 0, 0)	1.292×10^{-4}	31.78	0.9337
LoDoPaB (200)	0.01	Poisson	6	(0, 0, 0, 0, 4, 4)	4.0	32.52	0.7822
	0.1	Poisson	6	(0, 0, 0, 0, 4, 4)	3.0	32.78	0.7821

Table B7. FBP + U-Net. The input FBP reconstruction uses a Hann filter with no additional low-pass filter. Common hyperparameters: scales = 5, skip channels = 4, linear output (i.e. no sigmoid activation). The maximum learning rate is set to 10^{-2} or 10^{-3} and scheduled with either cosine annealing or one-cycle policy.

Dataset	Data size (%)	Channels	Batch size	Epochs	PSNR (dB)	SSIM
Ellipses	0.1	(32, 32, 64, 64, 128)	16	5000	26.33	0.7895
	0.2	(32, 32, 64, 64, 128)	16	5000	26.59	0.8042
	0.5	(32, 32, 64, 64, 128)	16	5000	26.80	0.8114
	1.0	(32, 32, 64, 64, 128)	16	5000	27.12	0.8321
	2.0	(32, 32, 64, 64, 128)	16	2500	27.44	0.8323
	5.0	(32, 32, 64, 64, 128)	16	1000	27.97	0.8604
	10.0	(64, 64, 128, 128, 256)	16	700	28.49	0.8751
	25.0	(64, 64, 128, 128, 256)	16	280	28.80	0.8872
	50.0	(64, 64, 128, 128, 256)	16	140	29.10	0.8940
	100.0	(64, 64, 128, 128, 256)	16	70	29.36	0.8987
LoDoPaB (200)	0.01	(32, 32, 64, 64, 128)	32	5000	29.33	0.7143
	0.1	(32, 32, 64, 64, 128)	32	5000	31.58	0.7616
	1.0	(32, 32, 64, 64, 128)	32	2000	32.60	0.7818
	10.0	(32, 32, 64, 64, 128)	32	500	33.19	0.7931
	100.0	(32, 32, 64, 64, 128)	32	250	33.55	0.7994
LoDoPaB	0.01	(32, 32, 64, 64, 128)	32	5000	31.36	0.7727
	0.1	(32, 32, 64, 64, 128)	32	5000	33.27	0.7982
	1.0	(32, 32, 64, 64, 128)	32	2000	34.62	0.8209
	10.0	(32, 32, 64, 64, 128)	32	500	35.18	0.8313
	100.0	(32, 32, 64, 64, 128)	32	250	35.48	0.8371

For the learned methods, the numbers of epochs listed in the tables denote the maximum numbers—the model with best mean PSNR on the validation set reached during training is selected. In some cases we used a learning rate scheduler that improved the training. More details can be found in https://github.com/oterobaguer/dip-ct-benchmark.

Table B8. Learned gradient descent. For all experiments the number of iterations is set to $L = 10$. The output of the network is linear, i.e. no sigmoid activation is used.

Dataset	Data size (%)	Channels	Batch size	Epochs	lr	PSNR (dB)	SSIM
Ellipses	0.1	32	32	5000	10^{-3}	27.81	0.8580
	0.2	32	32	5000	10^{-3}	28.40	0.8769
	0.5	32	32	5000	10^{-3}	29.15	0.8955
	1.0	32	32	5000	10^{-3}	29.55	0.9027
	2.0	32	32	2500	10^{-3}	29.70	0.9051
	5.0	32	32	1000	10^{-3}	29.84	0.9077
	10.0	32	32	500	10^{-3}	29.88	0.9082
	25.0	32	32	200	10^{-3}	29.95	0.9094
	50.0	32	32	100	10^{-3}	30.07	0.9121
	100.0	32	32	50	10^{-3}	30.30	0.9162
LoDoPaB (200)	0.01	32	20	5000	10^{-4}	29.87	0.7151
	0.1	32	20	5000	10^{-5}	31.28	0.7473
	1.0	32	20	500	10^{-5}	31.83	0.7602
	10.0	64	1	200	10^{-5}	32.41	0.7724
	100.0	64	1	20	10^{-5}	32.41	0.7724
LoDoPaB	0.01	32	1	5000	10^{-3}	32.70	0.7860
	0.1	32	1	5000	10^{-3}	33.81	0.8043
	1.0	32	1	500	10^{-3}	34.29	0.8103
	10.0	64	1	100	10^{-4}	34.34	0.8115
	100.0	64	1	10	10^{-4}	34.36	0.8122

Table B9. Learned primal-dual. For all experiments the number of iterations is set to $L = 10$. The output of the network is linear, i.e. no sigmoid activation is used.

Dataset	Data size [%]	Channels	Batch size	Epochs	lr	PSNR [dB]	SSIM
Ellipses	0.1	32	5	5000	10^{-3}	28.09	0.8621
	0.2	32	5	5000	10^{-3}	28.45	0.8778
	0.5	32	5	5000	10^{-3}	29.35	0.8997
	1.0	32	5	5000	10^{-3}	30.11	0.9124
	2.0	32	5	2500	10^{-3}	30.84	0.9258
	5.0	32	5	1000	10^{-3}	31.44	0.9282
	10.0	32	5	500	10^{-3}	31.84	0.9360
	25.0	32	5	200	10^{-3}	32.15	0.9367
	50.0	32	5	100	10^{-3}	32.21	0.9390
	100.0	32	5	50	10^{-3}	32.27	0.9403
LoDoPaB (200)	0.01	32	1	5000	10^{-3}	29.65	0.7343
	0.1	32	1	5000	10^{-3}	32.48	0.7771
	1.0	32	1	500	10^{-3}	33.21	0.7929
	10.0	64	1	100	10^{-4}	33.53	0.7990
	100.0	64	1	10	10^{-4}	33.64	0.8020
LoDoPaB	0.01	32	1	5000	10^{-3}	32.68	0.7842
	0.1	32	1	5000	10^{-3}	34.65	0.8227
	1.0	32	1	500	10^{-3}	35.27	0.8303
	10.0	64	1	100	10^{-4}	35.63	0.8401
	100.0	64	1	10	10^{-4}	35.73	0.8426

Table B10. iRadonMap. The U-Net part of the network has the same hyperparameters for all experiments: scales = 5, skip channels = 4, channels = (32, 32, 64, 64, 128). The learning rate is set to 10^{-2}. Selection of the sigmoid output is based on the validation performance; the difference on LoDoPaB with and without sigmoid is marginal.

Dataset	Data size (%)	Batch size	Epochs	Sigmoid output	PSNR (dB)	SSIM
Ellipses	0.1	64	1000	✓	17.83	0.2309
	0.2	64	1000	✓	18.35	0.2837
	0.5	64	1000	✓	21.41	0.5378
	1.0	64	1000	✓	22.64	0.6312
	2.0	64	1000	✓	23.62	0.7042
	5.0	64	1000	✓	24.77	0.7444
	10.0	64	1000	✓	25.61	0.8051
	25.0	64	400	✓	26.56	0.8389
	50.0	64	200	✓	27.36	0.8615
	100.0	64	100	✓	28.02	0.8766
LoDoPaB (200)	0.01	32	150	✓	14.61	0.3529
	0.1	32	150		18.77	0.4492
	1.0	32	150		24.63	0.6031
	10.0	32	150		31.27	0.7569
	100.0	32	30	✓	32.45	0.7781
LoDoPaB	0.01	2	150		14.82	0.3737
	0.1	2	150		17.67	0.4438
	1.0	2	150		22.73	0.5361
	10.0	2	150		28.69	0.6929
	100.0	2	15	✓	30.99	0.7486

ORCID iDs

Daniel Otero Baguer https://orcid.org/0000-0001-6550-6043
Johannes Leuschner https://orcid.org/0000-0001-7361-9523
Maximilian Schmidt https://orcid.org/0000-0001-8710-1389

References

[1] Adler J and Öktem O 2017 Solving ill-posed inverse problems using iterative deep neural networks *Inverse Problems* **33** 124007
[2] Adler J and Öktem O 2018 Deep Bayesian inversion (arXiv:1811.0591)
[3] Adler J and Öktem O 2018 Learned primal-dual reconstruction *IEEE Trans. Med. Imaging* **37** 1322–32
[4] Antun V, Renna F, Poon C, Adcock B and Hansen A C 2020 On instabilities of deep learning in image reconstruction and the potential costs of AI *Proc. Natl Acad. Sci.* (https://www.pnas.org/content/early/2020/05/08/1907377117/tab-article-info)
[5] Armato S G III *et al* 2011 The lung image database consortium (LIDC) and image database resource initiative (IDRI): a completed reference database of lung nodules on CT scans *Med. Phys.* **38** 915–31
[6] Arridge S, Maass P, Öktem O and Schönlieb C-B 2019 Solving inverse problems using data-driven models *Acta Numerica* **28** 1–174
[7] Bora A, Jalal A, Price E and Dimakis A G 2017 Compressed sensing using generative models *Proc. 34th Int. Conf. on Machine Learning, ICML 2017* (Sydney, NSW, Australia 6–11 August 2017) pp 537–46

[8] Bubba T A, Kutyniok G, Lassas M, März M, Samek W, Siltanen S and Srinivasan V 2019 Learning the invisible: a hybrid deep learning-shearlet framework for limited angle computed tomography *Inverse Problems* **35** 064002

[9] Buzug T M 2008 *Computed Tomography: From Photon Statistics to Modern Cone-Beam CT* (Berlin, Heidelberg: Springer)

[10] Chakrabarty P and Maji S 2019 The spectral bias of the deep image prior (arXiv:1912.08905)

[11] Chen H, Zhang Y, Kalra M K, Lin F, Chen Y, Liao P, Zhou J and Wang G 2017 Low-dose CT with a residual encoder-decoder convolutional neural network *IEEE Trans. Med. Imaging* **36** 2524–35

[12] Cheng Z, Gadelha M, Maji S and Sheldon D 2019 A bayesian perspective on the deep image prior *The IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*

[13] Denker A, Schmidt M, Leuschner J, Maass P and Behrmann J 2020 Conditional normalizing flows for low-dose computed tomography image reconstruction (arXiv:2006.06270)

[14] Dittmer S, Kluth T, Maass P and Otero Baguer D 2019 Regularization by architecture: a deep prior approach for inverse problems *J. Math. Imaging Vis.* **62** 456–70

[15] Donoho D L and Johnstone I M 1994 Ideal spatial adaptation by wavelet shrinkage *Biometrika* **81** 425–55

[16] Engl H W, Hanke M and Neubauer A 1996 *Regularization of Inverse Problems* (Mathematics and its Applications vol 375) (Dordrecht: Kluwer)

[17] Gandelsman Y, Shocher A and Irani M 2019 "Double-DIP": Unsupervised image decomposition via coupled deep-image-priors *2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 11018–27

[18] Gong K, Catana C, Qi J and Li Q 2019 PET image reconstruction using deep image prior *IEEE Trans. Med. Imaging* **38** 1655–65

[19] Gottschling N M, Antun V, Adcock B and Hansen A C 2020 The troublesome kernel: why deep learning for inverse problems is typically unstable (arXiv:2001.01258)

[20] Gupta H, Jin K H, Nguyen H Q, McCann M T and Unser M 2018 CNN-based projected gradient descent for consistent CT image reconstruction *IEEE Trans. Med. Imaging* **37** 1440–53

[21] Hauptmann A, Lucka F, Betcke M, Huynh N, Adler J, Cox B, Beard P, Ourselin S and Arridge S 2018 Model-based learning for accelerated, limited-view 3-D photoacoustic tomography *IEEE Trans. Med. Imaging* **37** 1382–93

[22] He J, Wang Y and Ma J 2020 Radon inversion via deep learning *IEEE Trans. Med. Imaging* **39** 2076–87

[23] He K, Zhang X, Ren S and Sun J 2016 Deep residual learning for image recognition *2016 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 770–8

[24] Heckel R and Soltanolkotabi M 2020 Denoising and regularization via exploiting the structural bias of convolutional generators *Int. Conf. on Learning Representations*

[25] Hofmann B, Kaltenbacher B, Pöschl C and Scherzer O 2007 A convergence rates result for tikhonov regularization in banach spaces with non-smooth operators *Inverse Problems* **23** 987–1010

[26] Hoyer S, Sohl-Dickstein J and Greydanus S 2019 Neural reparameterization improves structural optimization (arXiv:1909.04240)

[27] Jin K H, Gupta H, Yerly J, Stuber M and Unser M 2019 Time-dependent deep image prior for dynamic MRI (arXiv:1910.01684)

[28] Jin K H, McCann M T, Froustey E and Unser M 2017 Deep convolutional neural network for inverse problems in imaging *IEEE Trans. Image Process.* **26** 4509–22

[29] Kingma D P and Ba J 2015 Adam: a method for stochastic optimization *3rd Int. Conf. on Learning Representations, ICLR 2015* eds Y Bengio and Y LeCun (San Diego, CA, USA May 7–9, 2015)

[30] Knoll F *et al* 2020 fastMRI: a publicly available raw k-space and DICOM dataset of knee images for accelerated MR image reconstruction using machine learning *Radiology. Artificial intelligence* **2** e190007

[31] Lempitsky V, Vedaldi A and Ulyanov D 2018 Deep image prior *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* pp 9446–54

[32] Leuschner J, Schmidt M, Otero Baguer D and Maass P 2019 The LoDoPaB-CT dataset: a benchmark dataset for low-dose CT reconstruction methods (arXiv:1910.01113)

[33] Leuschner J, Schmidt M and Erzmann D 2019 Deep inversion validation library https://github.com/jleuschn/dival

[34] Li H, Schwab J, Antholzer S and Haltmeier M 2020 NETT: Solving inverse problems with deep neural networks *Inverse Problems* (accepted manuscript)

[35] Liu J, Sun Y, Xu X and Kamilov U S 2019 Image restoration using total variation regularized deep image prior *ICASSP 2019—2019 IEEE Int. Conf. on Acoustics Speech and Signal Processing (ICASSP)* pp 7715–9
[36] Louis A K 1989 *Inverse und schlecht gestellte Probleme* (Wiesbaden: Vieweg$^+$Teubner Verlag)
[37] Lunz S, Öktem O and Schönlieb C-B 2018 Adversarial regularizers in inverse problems *Proc. 32nd Int. Conf. on Neural Information Processing Systems, NIPS'18* (*Red Hook, NY, USA*) pp 8516–25
[38] Mataev G, Elad M and Milanfar P 2019 DeepRED: deep image prior powered by RED (arXiv:1903.10176)
[39] Zuhair Nashed M 1987 A new approach to classification and regularization of ill-posed operator equations *Inverse and Ill-Posed Problems* eds H W Engl and C W Groetsch (New York: Academic) pp 53–75
[40] Natterer F 2001 The mathematics of computerized tomography *Classics in Applied Mathematics* (Philadelphia: Society for Industrial and Applied Mathematics)
[41] Paszke A *et al* 2017 Automatic differentiation in PyTorch *NIPS 2017 Workshop on Autodiff*
[42] Pelt D, Batenburg K and Sethian J 2018 Improving tomographic reconstruction from limited data using mixed-scale dense convolutional neural networks *J. Imaging* **4** 128
[43] Radon J 1986 On the determination of functions from their integral values along certain manifolds *IEEE Trans. Med. Imaging* **5** 170–6
[44] Rieder A 2003 *Keine Probleme mit inversen Problemen: eine Einführung in ihre stabile Lösung* (Braunschweig: Vieweg)
[45] Ronneberger O, Fischer P and Brox T 2015 U-Net: convolutional networks for biomedical image segmentation *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015* eds N Navab, J Hornegger, W M Wells and A F Frangi (Berlin: Springer) pp 234–41
[46] Schwab J, Antholzer S and Haltmeier M 2019 Deep null space learning for inverse problems: convergence analysis and rates *Inverse Problems* **35** 025008
[47] Van Veen D, Jalal A, Soltanolkotabi M, Price E, Vishwanath S and Dimakis A G 2018 Compressed sensing with deep image prior and learned regularization (arXiv:1806.06438)
[48] Yang Q *et al* 2018 Low-dose CT image denoising using a generative adversarial network with wasserstein distance and perceptual loss *IEEE Trans. Med. Imaging* **37** 1348–57
[49] Zhu B, Liu J Z, Cauley S F, Rosen B R and Rosen M S 2018 Image reconstruction by domain-transform manifold learning *Nature* **555** 487–92

Paper 3

Quantitative comparison of deep learning-based image reconstruction methods for low-dose and sparse-angle CT applications

Journal of *Imaging*

MDPI

Article

Quantitative Comparison of Deep Learning-Based Image Reconstruction Methods for Low-Dose and Sparse-Angle CT Applications

Johannes Leuschner [1,*,†], Maximilian Schmidt [1,†], Poulami Somanya Ganguly [2,3], Vladyslav Andriiashen [2], Sophia Bethany Coban [2], Alexander Denker [1], Dominik Bauer [4], Amir Hadjifaradji [5], Kees Joost Batenburg [2,6], Peter Maass [1] and Maureen van Eijnatten [2,7,*]

1 Center for Industrial Mathematics, University of Bremen, Bibliothekstr. 5, 28359 Bremen, Germany; maximilian.schmidt@uni-bremen.de (M.S.); adenker@uni-bremen.de (A.D.); pmaass@uni-bremen.de (P.M.)
2 Centrum Wiskunde & Informatica, Science Park 123, 1098 XG Amsterdam, The Netherlands; poulami.ganguly@cwi.nl (P.S.G.); vladyslav.andriiashen@cwi.nl (V.A.); sophia.coban@cwi.nl (S.B.C.); k.j.batenburg@cwi.nl (K.J.B.)
3 The Mathematical Institute, Leiden University, Niels Bohrweg 1, 2333 CA Leiden, The Netherlands
4 Computer Assisted Clinical Medicine, Heidelberg University, Theodor-Kutzer-Ufer 1-3, 68167 Mannheim, Germany; dominik.bauer@medma.uni-heidelberg.de
5 School of Biomedical Engineering, University of British Columbia, 2222 Health Sciences Mall, Vancouver, BC V6T 1Z3, Canada; ahadji@student.ubc.ca
6 Leiden Institute of Advanced Computer Science, Niels Bohrweg 1, 2333 CA Leiden, The Netherlands
7 Department of Biomedical Engineering, Eindhoven University of Technology, Groene Loper 3, 5612 AE Eindhoven, The Netherlands
* Correspondence: jleuschn@uni-bremen.de (J.L.); m.a.j.m.v.eijnatten@tue.nl (M.v.E.)
† These authors contributed equally to this work.

check for updates

Citation: Leuschner, J.; Schmidt, M.; Ganguly, P.S.; Andriiashen, V.; Coban, S.B.; Denker, A.; Bauer, D.; Hadjifaradji, A.; Batenburg, K.J.; Maass, P.; et al. Quantitative Comparison of Deep Learning-Based Image Reconstruction Methods for Low-Dose and Sparse-Angle CT Applications. *J. Imaging* **2021**, *7*, 44. https://doi.org/10.3390/jimaging7030044

Academic Editors: Yudong Zhang, Juan Manuel Gorriz and Zhengchao Dong

Received: 29 January 2021
Accepted: 22 February 2021
Published: 2 March 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Abstract: The reconstruction of computed tomography (CT) images is an active area of research. Following the rise of deep learning methods, many data-driven models have been proposed in recent years. In this work, we present the results of a *data challenge* that we organized, bringing together algorithm experts from different institutes to jointly work on quantitative evaluation of several data-driven methods on two large, public datasets during a ten day sprint. We focus on two applications of CT, namely, low-dose CT and sparse-angle CT. This enables us to fairly compare different methods using standardized settings. As a general result, we observe that the deep learning-based methods are able to improve the reconstruction quality metrics in both CT applications while the top performing methods show only minor differences in terms of peak signal-to-noise ratio (PSNR) and structural similarity (SSIM). We further discuss a number of other important criteria that should be taken into account when selecting a method, such as the availability of training data, the knowledge of the physical measurement model and the reconstruction speed.

Keywords: computed tomography (CT); image reconstruction; low-dose; sparse-angle; deep learning; quantitative comparison

1. Introduction

Computed tomography (CT) is a widely used (bio)medical imaging modality, with various applications in clinical settings, such as diagnostics [1], screening [2] and virtual treatment planning [3,4], as well as in industrial [5] and scientific [6–8] settings. One of the fundamental aspects of this modality is the reconstruction of images from multiple X-ray measurements taken from different angles. Because each X-ray measurement exposes the sample or patient to harmful ionizing radiation, minimizing this exposure remains an active area of research [9]. The challenge is to either minimize the dose per measurement or the total number of measurements while maintaining sufficient image quality to perform subsequent diagnostic or analytic tasks.

To date, the most common classical methods used for CT image reconstruction are filtered back-projection (FBP) and iterative reconstruction (IR) techniques. FBP is a stabilized and discretized version of the inverse Radon transform, in which 1D projections are filtered by the 1D Radon kernel (back-projected) in order to obtain a 2D signal [10,11]. FBP is very fast, but is not suitable for limited-data or sparse-angle setups, resulting in various imaging artifacts, such as streaking, stretching, blurring, partial volume effects, or noise [12]. Iterative reconstruction methods, on the other hand, are computationally intensive but are able to incorporate *a priori* information about the system during reconstruction. Many iterative techniques are based on statistical methods such as Markov random fields or regularization methods where the regularizers are designed and incorporated into the problem of reconstruction mathematically [13]. A popular choice for the regularizer is total variation (TV) [14,15]. Another well-known iterative method suitable for large-scale tomography problems is the conjugate gradient method applied to solve the least squares problem (CGLS) [16].

When classical techniques such as FBP or IR are used to reconstruct low-dose CT images, the image quality often deteriorates significantly in the presence of increased noise. Therefore, the focus is shifting towards developing reconstruction methods in which a single or multiple component(s), or even the entire reconstruction process is performed using deep learning [17]. Generally data-driven approaches promise fast and/or accurate image reconstruction by taking advantage of a large number of examples, that is, training data.

The methods that learn parts of the reconstruction process can be roughly divided into learned regularizers, unrolled iterative schemes, and post-processing of reconstructed CT images. Methods based on learned regularizers work on the basis of learning convolutional filters from the training data that can subsequently be used to regularize the reconstruction problem by plugging into a classical iterative optimization scheme [18]. Unrolled iterative schemes go a step further in the sense that they "unroll" the steps of the iterative scheme into a sequence of operations where the operators are replaced with convolutional neural networks (CNNs). A recent example is the learned primal-dual algorithm proposed by Adler et al. [19]. Finally, various post-processing methods have been proposed that correct noisy images or those with severe artifacts in the image domain [20]. Examples are improving tomographic reconstruction from limited data using a mixed-scale dense (MS-D) CNN [21], U-Net [22] or residual encoder-decoder CNN (RED-CNN) [23], as well as CT image denoising techniques [24,25]. Somewhat similar are the methods that can be trained in a supervised manner to improve the measurement data in the sinogram domain [26].

The first fully end-to-end learned reconstruction method was the automated transform by the manifold approximation (AUTOMAP) algorithm [27] developed for magnetic resonance (MR) image reconstruction. This method directly learns the (global) relation between the measurement data and the image, that is, it replaces the Radon or Fourier transform with a neural network. The disadvantages of this approach are the large memory requirements, as well as the fact that it might not be necessary to learn the entire transformation from scratch because an efficient analytical transform is already available. A similar approach for CT reconstruction was iRadonMAP proposed by He et al. [28], who developed an interpretable framework for Radon inversion in medical X-ray CT. In addition, Li et al. [29] proposed an end-to-end reconstruction framework for Radon inversion called iCT-Net, and demonstrated its advantages in solving sparse-view CT reconstruction problems.

The aforementioned deep learning-based CT image reconstruction methods differ greatly in terms of which component of the reconstruction task is learned and in which domain the method operates (image or sinogram domain), as well as the computational and data-related requirements. As a result, it remains difficult to compare the performance of deep learning-based reconstruction methods across different imaging domains and applications. Thorough comparisons between different reconstruction methods are further complicated by the lack of sufficiently large benchmarking datasets, including ground truth

reconstructions, for training, validation, and testing. CT manufacturers are typically very reluctant in making raw measurement data available for research purposes, and privacy regulations for making medical imaging data publicly available are becoming increasingly strict [30,31].

1.1. Goal of This Study

The aim of this study is to quantitatively compare the performance of classical and deep learning-based CT image reconstruction methods on two large, two-dimensional (2D) parallel-beam CT datasets that were specifically created for this purpose. We opted for a 2D parallel-beam CT setup to facilitate large-scale experiments with many example images, whereas the underlying operators in the algorithms have straightforward generalizations to other geometries. We focus on two reconstruction tasks with high relevance and impact—the first task is the reconstruction of low-dose medical CT images, and the second is the reconstruction of sparse-angle CT images.

1.1.1. Reconstruction of Low-Dose Medical CT Images

In order to compare (learned) reconstruction techniques in a low-dose CT setup, we use the low-dose parallel beam (LoDoPaB) CT dataset [32]. This dataset contains 42,895 two-dimensional CT images and corresponding simulated low-intensity measurements. The ground truth images of this dataset are human chest CT reconstructions taken from the LIDC/IDRI database [33]. These scans had been acquired with a wide range of scanners and models. The initial image reconstruction for creating the LIDC/IDRI database was performed with different convolution kernels, depending on the manufacturer. Poisson noise is applied to the simulated projection data to model the low intensity setup. A more detailed description can be found in Section 2.1.

1.1.2. Reconstruction of Sparse-Angle CT Images

When using X-ray tomography in high-throughput settings (i.e., scanning multiple objects per second) such as quality control, luggage scanning or inspection of products on conveyor belts, very few X-ray projections can be acquired for each object. In such settings, it is essential to incorporate *a priori* information about the object being scanned during image reconstruction. In order to compare (learned) reconstruction techniques for this application, we reconstruct parallel-beam CT images of apples with internal defects using as few measurements as possible. We experimented with three different noise settings: noise-free, Gaussian noise, and scattering noise. The generation of the datasets is described in Section 2.2.

2. Dataset Description

For both datasets, the simulation model uses a 2D parallel beam geometry for the creation of the measurements. The attenuation of the X-rays is simulated using the Radon transform [10]

$$\mathcal{A}x(s,\varphi) := \int_{\mathbb{R}} x\left(s\begin{bmatrix}\cos(\varphi)\\ \sin(\varphi)\end{bmatrix} + t\begin{bmatrix}-\sin(\varphi)\\ \cos(\varphi)\end{bmatrix}\right) \mathrm{d}t, \tag{1}$$

where $s \in \mathbb{R}$ is the distance from the origin and $\varphi \in [0, \pi)$ the angle of the beam (cf. Figure 1). Mathematically, the image is transformed into a function of (s, φ). For each fixed angle φ the 2D image x is projected onto a line parameterized by s, namely the X-ray detector.

A detailed description of both datasets is given below. Their basic properties are also summarized in Table 1.

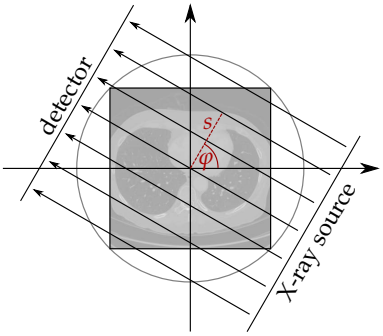

Figure 1. Parallel beam geometry. Adopted from [34].

Table 1. Settings of the low-dose parallel beam computed tomography (LoDoPaB-CT) and Apple CT datasets.

Property	LoDoPaB-CT	Apple CT
Subject	Human thorax	Apples
Scenario	low photon count	sparse-angle
Challenge	3678 reconstructions	100 reconstructions
Image size	362 px × 362 px	972 px × 972 px
Angles	1000	50, 10, 5, 2
Detector bins	513	1377
Sampling ratio	≈3.9	≈0.07–0.003

2.1. LoDoPaB-CT Dataset

The LoDoPaB-CT dataset [32] is a comprehensive collection of reference reconstructions and simulated low-dose measurements. It builds upon normal-dose thoracic CT scans from the LIDC/IDRI Database [33,35], whereby quality-assessed and processed 2D reconstructions are used as a ground truth. LoDoPaB features more than 40,000 scan slices from around 800 different patients. The dataset can be used for the training and evaluation of all kinds of reconstruction methods. LoDoPaB-CT has a predefined division into four parts, where each subset contains images from a distinct and randomly chosen set of patients. Three parts were used for training, validation and testing, respectively. It also contains a special challenge set with scans from 60 different patients. The ground truth images are undisclosed, and the patients are only included in this set. The challenge set is used for the evaluation of the model performance in this paper. Overall, the dataset contains 35,820 training images, 3522 validation images, 3553 test images and 3678 challenge images.

Low-intensity measurements suffer from an increased noise level. The main reason is so called quantum noise. It stems from the process of photon generation, attenuation and detection. The influence on the number of detected photons $\tilde{N}_1$ can be modeled, based on the mean photon count without attenuation N_0 and the Radon transform (1), by a Poisson distribution [36]

$$\tilde{\mathbb{N}}_1(s,\varphi) \sim \mathrm{Pois}(N_0 \exp(-\mathcal{A}x(s,\varphi))). \tag{2}$$

The model has to be discretized concerning s and φ for the simulation process. In this case, the Radon transform (1) becomes a finite-dimensional linear map $A : \mathbb{R}^n \to \mathbb{R}^m$, where n is the number of image pixels and m is the product of the number of detector pixels and the number of discrete angles. Together with the Poisson noise, the discrete simulation model is given by

$$Ax + \mathbb{e}(Ax) = \mathbb{y}_\delta, \qquad \mathbb{e}(Ax) = -Ax - \ln(\tilde{\mathbb{N}}_1/N_0), \qquad \tilde{\mathbb{N}}_1 \sim \mathrm{Pois}(N_0 \exp(-Ax)). \tag{3}$$

A single realization $y_\delta \in \mathbb{R}^m$ of $\mathbb{y}_\delta$ is observed for each ground truth image, $x = x^\dagger \in \mathbb{R}^n$. After the simulation according to (3), all data pairs $(y_\delta, x^\dagger)$ have been divided by

$\mu_{\max} = 81.35858$ to normalize the image values to the range $[0, 1]$. In the following sections, $\mathbb{y}_\theta$, y_δ and $x^\dagger$ denote the normalized values.

The LoDoPaB ground truth images have a resolution of 362 px × 362 px on a domain of size 26 cm × 26 cm. The scanning setup consists of 513 equidistant detector pixels s spanning the image diameter and 1000 equidistant angles φ between 0 and π. The mean photon count per detector pixel without attenuation is $N_0 = 4096$. The sampling ratio between the size of the measurements and the images is around 3.9 (oversampling case).

2.2. Apple CT Datasets

The Apple CT datasets [37] are a collection of ground truth reconstructions and simulated parallel beam data with various noise types and angular range sampling. The data is intended for benchmarking different algorithms and is particularly suited for use in deep learning settings due to the large number of slices available.

A total of 94 apples were scanned at the Flex-Ray Laboratory [8] using a point-source circular cone-beam acquisition setup. High quality ground truth reconstructions were obtained using a full rotation with an angular resolution of 0.005 rad and a spatial resolution of 54.2 µm. A collection of 1D parallel beam data for more than 70,000 slices were generated using the simulation model in Equation (1). A total of 50 projections were generated over an angular range of $[0, \pi)$, each of size 1×1377. The Apple CT ground truth images have a resolution of 972 px × 972 px. In order to make the angular sampling even sparser, we also reduced the data to include only 10, 5 and 2 angles. The angular sampling ranges are shown in Figure 2.

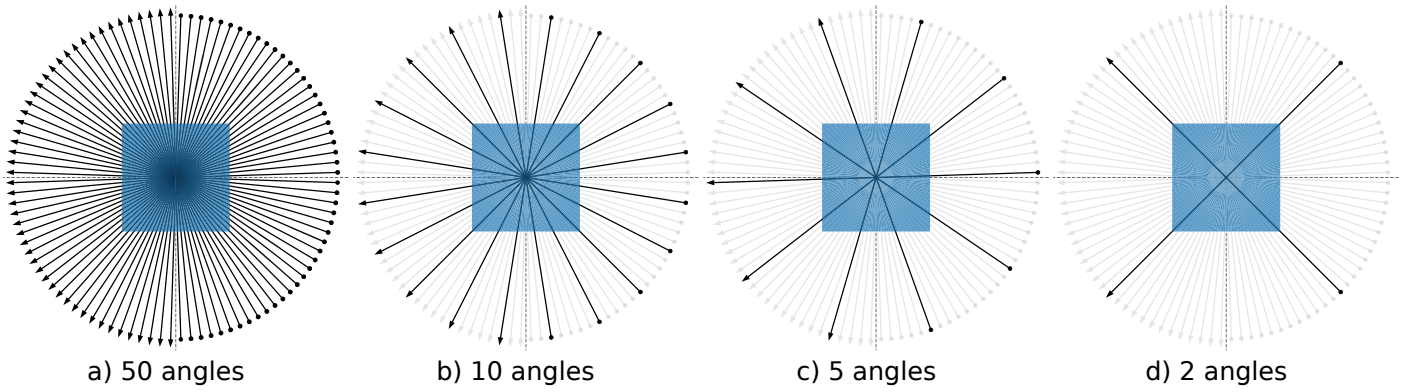

Figure 2. The angular sampling ranges employed for sparse image reconstructions for (**a**) 50 (full), (**b**) 10 (subset of 50 angles), (**c**) 5 (subset of 50 angles) and (**d**) 2 angles (subset of 10 angles). The black arrows show the position of the X-ray source (dot) and the position of the detector (arrowhead). For the sparse-angle scenario, the unused angles are shown in light gray.

The noise-free simulated data (henceforth Dataset A) were corrupted with 5% Gaussian noise to create Dataset B. Dataset C was generated by adding an imitation of scattering to Dataset A. Scattering intensity in a pixel u' is computed according to the formula

$$S(u') = \int_{u \in \mathbb{R}^2} G(u) \exp\left[-\frac{(u-u')^2}{2\sigma_1(u)^2}\right] + H(u) \exp\left[-\frac{(u-u')^2}{2\sigma_2(u)^2}\right], \tag{4}$$

where $|u - u'|$ is a distance between pixels, and scattering is approximated as a combination of Gaussian blurs with scaling factors G and H, standard deviations σ_1 and σ_2. Scattering noise in the target pixel u' contains contributions from all image pixels u as sources of scattering. Gaussian blur parameters depend on the X-ray absorption in the source pixel. To sample functions $G(u)$, $H(u)$, $\sigma_1(u)$ and $\sigma_2(u)$, a Monte Carlo simulation was performed for different thicknesses of water that was chosen as a material close to apple flesh. Furthermore, scaling factors $G(u)$ and $H(u)$ were increased to create a more challenging problem. We note that due to the computational complexity required, the

number of slices on which the scattering model is applied is limited to 7520 (80 slices per apple), meaning the scattering training subset is smaller.

The Apple CT datasets consist of apple slices with and without internal defects. Internal defects were observed to be of four main types: bitter pit, holes, rot and browning. A reconstruction of a healthy apple slice and one with bitter pit is shown in Figure 3 as examples. Each Apple CT dataset was divided into training and test subsets using an empirical bias elimination method to ensure that apples in both subsets had similar defect statistics. This process is detailed in [38].

For the network training, the noise-free and Gaussian noise training subsets are further split into 44,647 training and 5429 validation samples, and the scattering training subset is split into 5280 training and 640 validation samples.

From the test subsets, 100 test slices were extracted in a similar manner like for the split in training and test subsets. All evaluations in this paper refer to these 100 test slices in order to keep the reconstruction time and storage volume within reasonable limits. Five slices were extracted from each of the 20 test apples such that in total each defect type is occurring with a pixel count ratio similar to its ratio on the full test subset. Additionally, the extracted slices have a pairwise distance of at least 15 slices in order to improve the image diversity. The selected list of slices is specified in the supplementing repository [39] as file `supp_material/apples/test_samples_ids.csv`.

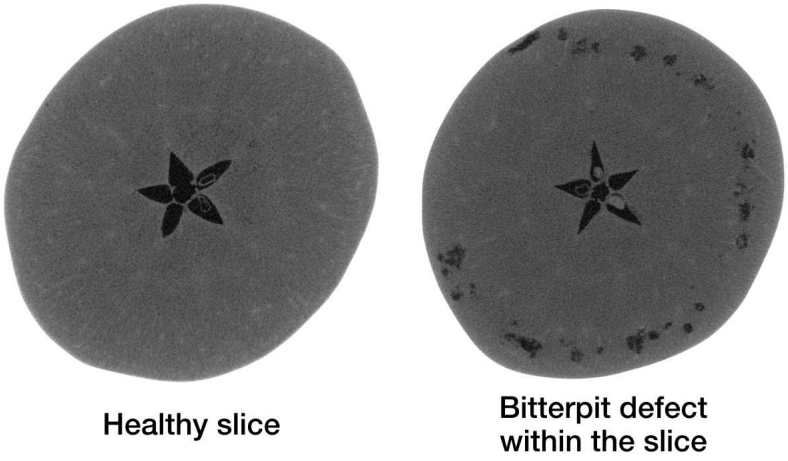

Figure 3. A horizontal cross-section of a healthy slice in an apple is shown on the **left**, and another cross-section with the bitter pit defects in the same apple on the **right**.

3. Algorithms

A variety of learned reconstruction methods were used to create a benchmark. The selection is based on methods submitted by participants for the data challenge on the LoDoPaB-CT and Apple CT datasets. The reconstruction methods include unrolled architectures, post-processing approaches, and fully-learned methods. Furthermore, classical methods such as FBP, TV regularization and CGLS were used as a baseline.

3.1. Learned Reconstruction Methods

In this section, the learned methods included in the benchmark are presented. An overview of the hyperparameters and pseudocode can be found in Appendix A. All methods utilize artificial neural networks F_Θ, each in different roles, for the reconstruction process.

Learning refers to the adaption of the parameters Θ for the reconstruction process in a data-driven manner. In general, one can divide this process into supervised and unsupervised learning. Almost all methods in this comparison are trained in a supervised way. This means that sample pairs $(y_\delta, x^\dagger)$ of noisy measurements and ground truth data are used for the optimization of the parameters, for example, by minimizing some

discrepancy $\mathcal{D}_X : X \times X \to \mathbb{R}$ between the output of the reconstruction model $\mathcal{T}_{F_\Theta}$ and the ground truth

$$\min_{\Theta} \mathcal{D}_X \left(\mathcal{T}_{F_\Theta}(y_\delta), x^\dagger \right). \tag{5}$$

Supervised methods often provide excellent results, but the number of required ground truth data can be high [34]. While the acquisition of ground truth images is infeasible in many applications, this is not a problem in the low-dose and sparse-angle case. Here, reconstructions of regular (normal-dose, full-angle) scans play the role of the reference.

3.1.1. Post-Processing

Post-processing approaches aim to improve the reconstruction quality of an existing method. When used in computed tomography, FBP (cf. Appendix B.1) is often used to obtain an initial reconstruction. Depending on the scan scenario, the FBP reconstruction can be noisy or contain artifacts. Therefore, it functions as an input for a learned post-processing method. This setting simplifies the task because the post-processing network $F_\Theta : X \to X$ maps directly from the target domain into the target domain

$$\hat{x} := [F_\Theta \circ \mathcal{T}_{\mathrm{FBP}}](y_\delta).$$

Convolutional neural networks (CNN) have successfully been used in recent works to remove artifacts and noise from FBP reconstructions. Four of these CNN post-processing approaches were used for the benchmark. The U-Net architecture [40] is a popular choice in many different applications and was also used for CT reconstruction [20]. The details of the network used in the comparison can be found in Appendix A.2. The U-Net++ [41] (cf. Appendix A.3) and ISTA U-Net [42] (cf. Appendix A.6) represent modifications of this approach. In addition, a mixed-scale dense (MS-D)-CNN [21] is included, which has a different architecture (cf. Appendix A.4). Like for the U-Net, one can consider to adapt other architectures originally used for segmentation, for example, the ENET [43], for the post-processing task.

3.1.2. Fully Learned

The goal of fully learned methods is to extract the structure of the inversion process from data. In this case, the neural network $F_\Theta : Y \to X$ directly maps from the measurement space Y to the target domain X. A prominent example is the AUTOMAP architecture [27], which was successfully used for reconstruction in magnetic resonance imaging (MRI). The main building blocks consist of fully-connected layers. This makes the network design very general, but the number of parameters can grow quickly with the data dimension. For example, a single fully-connected layer mapping from Y to X on the LoDoPaB-CT dataset (cf. Section 2.1) would require over $1000 \times 513 \times 362^2 \approx 67 \times 10^9$ parameters.

Adapted model designs exist for large CT data. They include knowledge about the inversion process in the structure of the network. He et al. [28] introduced an adapted two-part approach, called iRadonMap. The first part uses small fully-connected layers with parameter sharing to reproduce the structure of the FBP. This is followed by a post-processing network in the second part. Another approach is the iCT-Net [29], which uses convolutions in combination with fully-connected layers for the inversion. An extended version of the iCT-Net, called iCTU-Net, is part of our comparison and a detailed description can be found in Appendix A.8.

3.1.3. Learned Iterative Schemes

Similar to the fully learned approach, learned iterative methods also define a mapping directly from the measurement space Y to the target domain X. The idea in this case is that the network architecture is inspired by an analytic reconstruction operator $\mathcal{T} : Y \to X$ implicitly defined by an iterative scheme. The basic principle of unrolling can be explained

by the example of learned gradient descent (see e.g., [17]). Let $J(\cdot, y_\delta) : X \to \mathbb{R}$ be a smooth data discrepancy term and, possibly an additional regularization term. For an initial value $x^{[0]}$ the gradient descent is defined via the iteration

$$x^{[k+1]} = x^{[k]} - \omega_k \nabla_x J\left(x^{[k]}, y_\delta\right),$$

with a step size ω_k. Unrolling these iteration and stopping after K iterations, we can write the K-th iteration as

$$\mathcal{T}(y_\delta) := (\Lambda_{\omega_K} \circ \ldots \circ \Lambda_{\omega_1})(x^{[0]})$$

with $\Lambda_{\omega_k} := \mathrm{id} - \omega_k \nabla_x J(\cdot, y_\delta)$. In a learned iteration scheme, the operators Λ_{ω_k} are replaced by neural networks. As an example of a learned iterative procedure, learned primal-dual [19] was included in the comparison. A description of this method can be found in the Appendix A.1.

3.1.4. Generative Approach

The goal of the statistical approach to inverse problems is to determine the conditional distribution of the parameters given measured data. This statistical approach is often linked to Bayes' theorem [44]. In this Bayesian approach to inverse problems, the conditional distribution $p(x|y_\delta)$, called the posterior distribution, is supposed to be estimated. Based on this posterior distribution, different estimators, such as the maximum a posterior solution or the conditional mean, can be used as a reconstruction for the CT image. This theory provides a natural way to model the noise behavior and to integrate prior information into the reconstruction process. There are two different approaches that have been used for CT. Adler et al. [45] use a conditional variant of a generative adversarial network (GAN, [46]) to generate samples from the posterior. In contrast to this likelihood free approach, Ardizzone et al. [47] designed a conditional variant of invertible neural networks to directly estimate the posterior distribution. These conditional invertible neural networks (CINN) were also applied to the reconstruction of CT images [48]. The CINN was included for this benchmark. For a more detailed description, see Appendix A.5.

3.1.5. Unsupervised Methods

Unsupervised reconstruction methods just make use of the noisy measurements. They are favorable in applications where ground truth data is not available. The parameters of the model are chosen based on some discrepancy $\mathcal{D}_Y : Y \times Y \to \mathbb{R}$ between the output of the method and the measurements, for example,

$$\min_\Theta \mathcal{D}_Y\left(\mathcal{A}\mathcal{T}_{F_\Theta}(\cdot), y_\delta\right). \tag{6}$$

In this example, the output of $\mathcal{T}_{F_\Theta}$ plays the role of the reconstruction $\hat{x}$. However, comparing the distance just in the measurement domain can be problematic. This applies in particular to ill-posed reconstruction problems. For example, if the forward operator $\mathcal{A}$ is not bijective, no/multiple reconstruction(s) might match the measurement perfectly (ill-posed in the sense of Hadamard [49]). Another problem can occur for forward operators with an unstable inversion, where small differences in the measurement space, for example, due to noise, can result in arbitrary deviations in the reconstruction domain (ill-posed in the sense of Nashed [50]). In general, the minimization problem (6) is combined with some kind of regularization to mitigate these problems.

The optimization Formulation (6) is also used for the deep image prior (DIP) approach. DIP takes a special role among all neural network methods. The parameters are not determined on a dedicated training set, but during the reconstruction on the challenge data. This is done for each reconstruction separately. One could argue that the DIP approach is therefore not a learned method in the classical sense. The DIP approach, in combination with total variation regularization, was successfully used for CT reconstruction [34]. It is

part of the comparison on the LoDoPaB dataset in this paper. A detailed description is given in Appendix A.7.

3.2. Classical Reconstruction Methods

In addition to the learned methods, we implemented the popularly used direct and iterative reconstruction methods, henceforth referred to as classical methods. They can often be described as a variational approach

$$\mathcal{T}(y_\delta) \in \arg\min_x \mathcal{D}_Y(\mathcal{A}x, y_\delta) + \alpha\mathcal{R}(x),$$

where $\mathcal{D}_Y : Y \times Y \to \mathbb{R}$ is a data discrepancy and $\mathcal{R} : X \to \mathbb{R}$ is a regularizer. In this context $\mathcal{T} : Y \to X$ defines the reconstruction operator. The included methods in the benchmark are filtered back-projection (FBP) [10,51], conjugate gradient least squares (CGLS) [52,53] and anisotropic total variation minimization (TV) [54]. Detailed description of each classical method along with pseudocode are given in Appendix B.

4. Evaluation Methodology

4.1. Evaluation Metrics

Two widely used evaluation metrics were used to assess the performance of the methods.

4.1.1. Peak Signal-to-Noise Ratio

The peak signal-to-noise ratio (PSNR) is measured by a log-scaled version of the mean squared error (MSE) between the reconstruction $\hat{x}$ and the ground truth image $x^\dagger$. PSNR expresses the ratio between the maximum possible image intensity and the distorting noise

$$\mathrm{PSNR}\left(\hat{x}, x^\dagger\right) := 10\log_{10}\left(\frac{L^2}{\mathrm{MSE}(\hat{x}, x^\dagger)}\right), \quad \mathrm{MSE}\left(\hat{x}, x^\dagger\right) := \frac{1}{n}\sum_{i=1}^{n}\left|\hat{x}_i - x_i^\dagger\right|^2. \tag{7}$$

In general, higher PSNR values are an indication of a better reconstruction. The maximum image value L can be chosen in different ways. In our study, we report two different values that are commonly used:

- **PSNR**: In this case $L = \max(x^\dagger) - \min(x^\dagger)$, that is, the difference between the highest and lowest entry in $x^\dagger$. This allows for a PSNR value that is adapted to the range of the current ground truth image. The disadvantage is that the PSNR is image-dependent in this case.
- **PSNR-FR**: The same fixed L is chosen for all images. It is determined as the maximum entry computed over all training ground truth images, that is, $L = 1.0$ for LoDoPaB-CT and $L = 0.0129353$ for the Apple CT datasets. This can be seen as an (empirical) upper limit of the intensity range in the ground truth. In general, a fixed L is preferable because the scaling of the metric is image-independent in this case. This allows for a direct comparison of PSNR values calculated on different images. The downside for most CT applications is, that high values ($\hat{=}$ dense material) are not present in every scan. Therefore, the results can be too optimistic for these scans. However, based on Equation (7), all mean PSNR-FR values can be directly converted for another fixed choice of L.

4.1.2. Structural Similarity

The structural similarity (SSIM) [55] compares the overall image structure of ground truth and reconstruction. It is based on assumptions about the human visual perception.

Results lie in the range $[0,1]$, with higher values being better. The SSIM is computed through a sliding window at M locations

$$\mathrm{SSIM}\left(\hat{x}, x^{\dagger}\right) := \frac{1}{M}\sum_{j=1}^{M} \frac{\left(2\hat{\mu}_j\mu_j + C_1\right)\left(2\Sigma_j + C_2\right)}{\left(\hat{\mu}_j^2 + \mu_j^2 + C_1\right)\left(\hat{\sigma}_j^2 + \sigma_j^2 + C_2\right)}. \tag{8}$$

In the formula above $\hat{\mu}_j$ and μ_j are the average pixel intensities, $\hat{\sigma}_j$ and σ_j the variances and Σ_j the covariance of $\hat{x}$ and $x^{\dagger}$ at the j-th local window. Constants $C_1 = (K_1 L)^2$ and $C_2 = (K_2 L)^2$ stabilize the division. Following Wang et al. [55] we choose $K_1 = 0.01$ and $K_2 = 0.03$ and a window size of 7×7. In accordance with the PSNR metric, results for the two different choices for L are reported as SSIM and SSIM-FR (cf. Section 4.1.1).

4.1.3. Data Discrepancy

Checking data consistency, that is, the discrepancy $\mathcal{D}_Y(\mathcal{A}\hat{x}, y_\delta)$ between the forward-projected reconstruction and the measurement, can provide additional insight into the performance of the reconstruction methods. Since noisy data is used for the comparison, an ideal method would yield a data discrepancy that is close to the present noise level.

Poisson Regression Loss on LoDoPaB-CT Dataset

For the Poisson noise model used by LoDoPaB-CT, an equivalent to the negative log-likelihood is calculated to evaluate the data consistency. It is conventional to employ the negative log-likelihood for this task, since minimizing the data discrepancy is equivalent to determining a maximum likelihood (ML) estimate (cf. Section 5.5 in [56] or Section 2.4 in [17]). Each element $\mathbb{y}_{\delta,j}$, $j = 1, \ldots, m$, of a measurement $\mathbb{y}_\delta$, obtained according to (3) and subsequently normalized by $\mu_{\max}$, is associated with an independent Poisson model of a photon count $\tilde{\mathbb{N}}_{1,j}$ with

$$\mathbb{E}(\tilde{\mathbb{N}}_{1,j}) = \mathbb{E}\big(N_0 \exp(-\mathbb{y}_{\delta,j}\mu_{\max})\big) = N_0 \exp(-y_j\mu_{\max}),$$

where y_j is a parameter that should be estimated [36]. A Poisson regression loss for y is obtained by summing the negative log-likelihoods for all measurement elements and omitting constant parts,

$$-\ell_{\mathrm{Pois}}(y \,|\, y_\delta) = -\sum_{j=1}^{m} N_0 \exp(-y_{\delta,j}\mu_{\max})(-y_j\mu_{\max} + \ln(N_0)) - N_0 \exp(-y_j\mu_{\max}), \tag{9}$$

with each $y_{\delta,j}$ being the only available realization of $\mathbb{y}_{\delta,j}$. In order to evaluate the likelihood-based loss (9) for a reconstructed image $\hat{x}$ given y_δ, the forward projection $A\hat{x}$ is passed for y.

Mean Squared Error on Apple CT Data

On the Apple CT datasets we consider the mean squared error (MSE) data discrepancy,

$$\mathrm{MSE}_Y(y, y_\delta) = \frac{1}{m}\|y - y_\delta\|_2^2. \tag{10}$$

For an observation y_δ with Gaussian noise (Dataset B), this data discrepancy term is natural, as it is a scaled and shifted version of the negative log-likelihood of y given y_δ. In this noise setting, a good reconstruction usually should not achieve an MSE less than the variance of the Gaussian noise, that is, $\mathrm{MSE}_Y(A\hat{x}, y_\delta) \geq [0.05\frac{1}{m}\sum_{j=1}^{m}(Ax^{\dagger})_j]^2$. This can be motivated intuitively by the conception that a reconstruction that achieves a smaller MSE than the expected MSE of the ground truth probably fits the noise rather than the actual data of interest.

In the setting of y_δ being noise-free (Dataset A), the MSE of ideal reconstructions would be zero. On the other hand the MSE being zero does not imply that the reconstruction

matches the ground truth image because of the sparse-angle setting. Further, the MSE can not be used to judge reconstruction quality directly, as crucial differences in image domain may not be equally pronounced in the sinogram domain.

For the scattering observations (Dataset C), the MSE data discrepancy is considered, too, for simplicity.

4.2. Training Procedure

While the reconstruction process with learned methods usually is efficient, their training is more resource consuming. This limits the practicability of large hyperparameter searches. It can therefore be seen as a drawback of a learned reconstruction method if they require very specific hyperparameter choices for different tasks. As a result, it benefits a fair comparison to minimize the amount of hyperparameter searches. In general, default parameters, for example, from the original publications of the respective method, were used as a starting point. For some of the methods, good choices had been determined for the LoDoPaB-CT dataset first (cf. [34]) and were kept similar for the experiments on the Apple CT datasets. Further searches were only performed if required to obtain reasonable results. More details regarding the individual methods can be found in Appendix A. For the classical methods, hyperparameters were optimized individually for each setting of the Apple CT datasets (cf. Appendix B).

Most learned methods are trained using the mean squared error (MSE) loss. The exceptions are the U-Net++ using a loss combining MSE and SSIM, the iCTU-Net using an SSIM loss for the Apple CT datasets, and the CINN for which negative log-likelihood (NLL) and an MSE term are combined (see Appendix A for more details). Training curves for the trainings on the Apple CT datasets are shown in Appendix D. While we consider the convergence to be sufficient, continuing some of the trainings arguably would slightly improve the network. However, this mainly can be expected for those methods which are comparably time consuming to train (approximately 2 weeks for 20 epochs), in which case the limited number of epochs can be considered a fair regulation of resource usage.

Early stopping based on the validation performance is used for all trainings except for the ISTA U-Net on LoDoPaB-CT and for the iCTU-Net.

Source code is publicly available in a supplementing github repository [39]. Further records hosted by Zenodo provide the trained network parameters for the experiments on the Apple CT Datasets [57], as well as the submitted LoDoPaB-CT Challenge reconstructions [58] and the Apple CT test reconstructions of the 100 selected slices in all considered settings [59]. Source code and network parameters for some of the LoDoPaB-CT experiments are included in the DIV$\alpha\ell$ library [60], for others the original authors provide public repositories containing source code and/or parameters.

5. Results

5.1. LoDoPaB-CT Dataset

Ten different reconstruction methods were evaluated on the challenge set of the LoDoPaB-CT dataset. Reconstructions from these methods were either submitted as part of the CT Code Sprint 2020 (http://dival.math.uni-bremen.de/code_sprint_2020/, last accessed: 1 March 2021) (15 June–31 August 2020) or in the period after the event (1 September–31 December 2020).

5.1.1. Reconstruction Performance

In order to assess the quality of the reconstructions, the PSNR and the SSIM were calculated. The results from the official challenge website (https://lodopab.grand-challenge.org/, last accessed: 1 March 2021) are shown in Table 2. The differences between the learned methods are generally small. Notably, learned primal-dual yields the best performance with respect to both the PSNR and the SSIM. The following places are occupied by post-processing approaches, also with only minor differences in terms of the metrics. Of the other methods, DIP + TV stands out, with relatively good results for an unsuper-

vised method. DIP + TV is able to beat the supervised method iCTU-Net. The classical reconstruction models perform the worst of all methods. In particular, the performance of FBP shows a clear gap with the other methods. While learned primal-dual performs slightly better than the post-processing methods, the difference is not as significant as one could expect, considering that it incorporates the forward operator directly in the network. This could be explained by the beneficial combination of the convolutional architectures used for the post-processing, which are observed to perform well on a number of image processing tasks, and a sufficient number of available training samples. Otero et al. [34] investigated the influence of the size of the training dataset on the performance of different learned procedures on the LoDoPaB-CT dataset. Here, a significant difference is seen between learned primal-dual and other learned procedures when only a small subset of the training data is used.

Table 2. Results on the LoDoPaB-CT challenge set. Methods are ranked by their overall performance. The highest value for each metric is highlighted. All values are taken from the official challenge leaderboard https://lodopab.grand-challenge.org/evaluation/challenge/leaderboard/ (accessed on 4 January 2021).

Model	PSNR	PSNR-FR	SSIM	SSIM-FR	Number of Parameters
Learned P.-D.	36.25 ± 3.70	40.52 ± 3.64	0.866 ± 0.115	0.926 ± 0.076	874,980
ISTA U-Net	36.09 ± 3.69	40.36 ± 3.65	0.862 ± 0.120	0.924 ± 0.080	83,396,865
U-Net	36.00 ± 3.63	40.28 ± 3.59	0.862 ± 0.119	0.923 ± 0.079	613,322
MS-D-CNN	35.85 ± 3.60	40.12 ± 3.56	0.858 ± 0.122	0.921 ± 0.082	181,306
U-Net++	35.37 ± 3.36	39.64 ± 3.40	0.861 ± 0.119	0.923 ± 0.080	9,170,079
CINN	35.54 ± 3.51	39.81 ± 3.48	0.854 ± 0.122	0.919 ± 0.081	6,438,332
DIP + TV	34.41 ± 3.29	38.68 ± 3.29	0.845 ± 0.121	0.913 ± 0.082	hyperp.
iCTU-Net	33.70 ± 2.82	37.97 ± 2.79	0.844 ± 0.120	0.911 ± 0.081	147,116,792
TV	33.36 ± 2.74	37.63 ± 2.70	0.830 ± 0.121	0.903 ± 0.082	(hyperp.)
FBP	30.19 ± 2.55	34.46 ± 2.18	0.727 ± 0.127	0.836 ± 0.085	(hyperp.)

5.1.2. Visual Comparison

A representative reconstruction of all learned methods and the classical baseline is shown in Figure 4 to enable a qualitative comparison of the methods. An area of interest around the spine is magnified to compare the reproduction of small details and the sharpness of edges in the image. Some visual differences can be observed between the reconstructions. The learned methods produce somewhat smoother reconstructions in comparison to the ground truth. A possible explanations for the smoothness is the minimization of the empirical risk with respect to some variant of the $L2$-loss during the training of most learned methods, which has an averaging effect. The convolutional architecture of the networks can also have an impact. Adequate regularization during training and/or inference can be beneficial in this case (cf. Section 6.2.2 for a suitable class of regularizers). Additionally, the DIP + TV reconstruction appears blurry, which can be explained by the fact that it is the only unsupervised method in this comparison and thus has no access to ground truth data. The U-Net and the two modifications, U-Net++ and ISTA U-Net, show only slight visual differences on this example image.

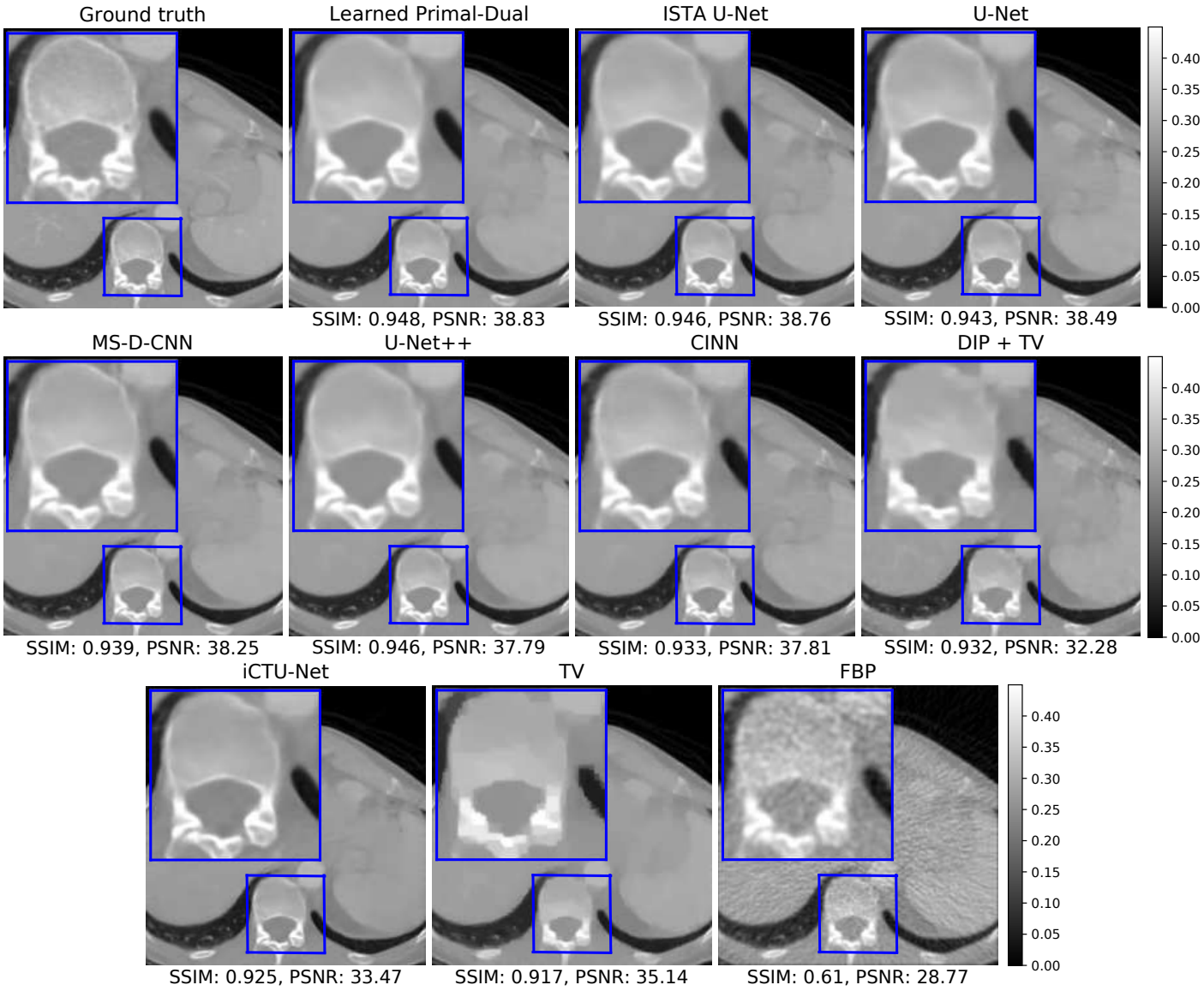

Figure 4. Reconstructions on the challenge set from the LoDoPaB-CT dataset. The window $[0, 0.45]$ corresponds to a HU range of $\approx[-1001, 831]$.

5.1.3. Data Consistency

The mean data discrepancy of all methods is shown in Figure 5, plotted against their reconstruction performance. The mean difference between the noise-free and noisy measurements is included as a reference. Good-performing models should be close to this empirical noise level. Values above the mean can indicate a sub-optimal data consistency, while values below can be a sign of overfitting to the noise. A data consistency term is only explicitly used in the TV and DIP + TV model. Nevertheless, the mean data discrepancy for most of the methods is close to the empirical noise level. The only visible outliers are the FBP and the iCTU-Net. A list of all mean data discrepancy values, including standard deviations, can be found in Table 3.

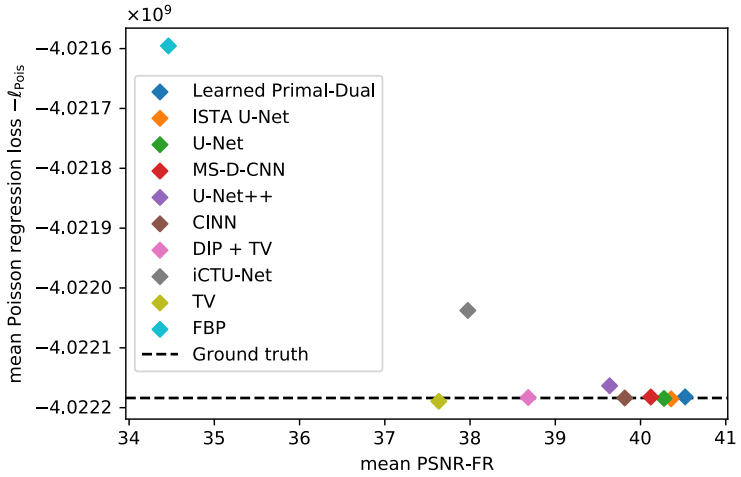

Figure 5. Mean data discrepancy $-\ell_{\text{Pois}}$ between the noisy measurements and the forward-projected reconstructions, respectively the noise-free measurements. Evaluation is done on the LoDoPaB challenge images.

Table 3. Mean and standard deviation of data discrepancy $-\ell_{\text{Pois}}$. Evaluation is done on the LoDoPaB challenge images.

Method	$-\ell_{\text{Pois}}(A\hat{x} \mid y_\delta)/10^9$
Learned Primal-Dual	-4.022182 ± 0.699460
ISTA U-Net	-4.022185 ± 0.699461
U-Net	-4.022185 ± 0.699460
MS-D-CNN	-4.022182 ± 0.699460
U-Net++	-4.022163 ± 0.699461
CINN	-4.022184 ± 0.699460
DIP + TV	-4.022183 ± 0.699466
iCTU-Net	-4.022038 ± 0.699430
TV	-4.022189 ± 0.699463
FBP	-4.021595 ± 0.699282
	$-\ell_{\text{Pois}}(Ax^\dagger \mid y_\delta)/10^9$
Ground truth	-4.022184 ± 0.699461

5.2. Apple CT Datasets

A total of 6 different learned methods were evaluated on the Apple CT data. This set included post-processing methods (MS-D-CNN, U-Net, ISTA U-Net), learned iterative methods (learned primal-dual), fully learned approaches (iCTU-Net), and generative models (CINN). As described in Section 2.2, different noise cases (noise-free, Gaussian noise and scattering noise) and different numbers of angles (50, 10, 5, 2) were used. In total, each model was trained on the 12 different settings of the Apple CT dataset. In addition to the learned methods, three classical techniques, namely CGLS, TV, and FBP, have been included as a baseline.

5.2.1. Reconstruction Performance

A subset of 100 data samples from the test set was selected for the evaluation (cf. Section 2.2). The mean PSNR and SSIM values for all experiments can be found in Table 4. Additionally, Tables A3–A5 in the appendix provide standard deviations and PSNR-FR and SSIM-FR values.

Table 4. Peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) (adapted to the data range of each ground truth image) for the different noise settings on the Apple CT datasets. Best results are highlighted in gray. See Figures A7 and A8 for a visualization.

Noise-Free	**PSNR**				**SSIM**			
Number of Angles	**50**	**10**	**5**	**2**	**50**	**10**	**5**	**2**
Learned Primal-Dual	38.72	35.85	30.79	22.00	0.901	0.870	0.827	0.740
ISTA U-Net	38.86	34.54	28.31	20.48	0.897	0.854	0.797	0.686
U-Net	39.62	33.51	27.77	19.78	0.913	0.803	0.803	0.676
MS-D-CNN	39.85	34.38	28.45	20.55	0.913	0.837	0.776	0.646
CINN	39.59	34.84	27.81	19.46	0.913	0.871	0.762	0.674
iCTU-Net	36.07	29.95	25.63	19.28	0.878	0.847	0.824	0.741
TV	39.27	29.00	22.04	15.95	0.915	0.783	0.607	0.661
CGLS	33.05	21.81	12.60	15.25	0.780	0.619	0.537	0.615
FBP	30.39	17.09	15.51	13.97	0.714	0.584	0.480	0.438
Gaussian Noise	**PSNR**				**SSIM**			
Number of Angles	**50**	**10**	**5**	**2**	**50**	**10**	**5**	**2**
Learned Primal-Dual	36.62	33.76	29.92	21.41	0.878	0.850	0.821	0.674
ISTA U-Net	36.04	33.55	28.48	20.71	0.871	0.851	0.811	0.690
U-Net	36.48	32.83	27.80	19.86	0.882	0.818	0.789	0.706
MS-D-CNN	36.67	33.20	27.98	19.88	0.883	0.831	0.748	0.633
CINN	36.77	31.88	26.57	19.99	0.888	0.771	0.722	0.637
iCTU-Net	32.90	29.76	24.67	19.44	0.848	0.837	0.801	0.747
TV	32.36	27.12	21.83	16.08	0.833	0.752	0.622	0.637
CGLS	27.36	21.09	14.90	15.11	0.767	0.624	0.553	0.616
FBP	27.88	17.09	15.51	13.97	0.695	0.583	0.480	0.438
Scattering Noise	**PSNR**				**SSIM**			
Number of Angles	**50**	**10**	**5**	**2**	**50**	**10**	**5**	**2**
Learned Primal-Dual	37.80	34.19	27.08	20.98	0.892	0.866	0.796	0.540
ISTA U-Net	35.94	32.33	27.41	19.95	0.881	0.820	0.763	0.676
U-Net	34.96	32.91	26.93	18.94	0.830	0.784	0.736	0.688
MS-D-CNN	38.04	33.51	27.73	20.19	0.899	0.818	0.757	0.635
CINN	38.56	34.08	28.04	19.14	0.915	0.863	0.839	0.754
iCTU-Net	26.26	22.85	21.25	18.32	0.838	0.796	0.792	0.765
TV	21.09	20.14	17.86	14.53	0.789	0.649	0.531	0.611
CGLS	20.84	18.28	14.02	14.18	0.789	0.618	0.547	0.625
FBP	21.01	15.80	14.26	13.06	0.754	0.573	0.475	0.433

The biggest challenge with the noise-free dataset is that the measurements become increasingly undersampled as the number of angles decreases. As expected, the reconstruction quality in terms of PSNR and SSIM deteriorates significantly as the number of angles decreases. In comparison with LoDoPaB-CT, no model performs best in all scenarios. Furthermore, most methods were trained to minimize the MSE between the output image and ground truth. The MSE is directly related to the PSNR. However, minimizing the MSE does not necessarily translate into a high SSIM. In many cases, the best method in terms of PSNR does not result in the best SSIM. These observations are also evident in the two noisy datasets. Noteworthy is the performance of the classical TV method on the noise-free dataset for 50 angles. This result is comparable to the best-performing learned methods, while the other classical approaches show a clear gap.

Noisy measurements, in addition to undersampling, present an additional difficulty on the Gaussian and scattering datasets. Intuitively, one would therefore expect a worse performance compared to the noise-free case. In general, a decrease in performance can be observed. However, this effect depends on the method and the noise itself. For example, the negative impact on classical methods is much more substantial for the scattering

noise. In contrast, the learned methods often perform slightly worse on the Gaussian noise. There are also some outliers with higher values than on the noise-free set. Possible explanations are the hyperparameter choices and the stochastic nature of the model training. Overall, the learned approaches can reach similar performances on the noisy data, while the performance of classical methods drops significantly. An additional observation can be made when comparing the results between Gaussian and scattering noise. For Gaussian noise with 50 angles, all learned methods, except for the iCTU net, achieve a PSNR of at least 36 dB. In contrast, the variation on scattering noise with 50 angles is much larger. The CINN obtains a much higher PSNR of 38.56 dB than the post-processing U-Net with 34.96 dB.

As already observed on the LoDoPaB dataset, the post-processing methods (MS-D-CNN, U-Net and ISTA U-Net) show only minor differences in all noise cases. This could be explained by the fact that these methods are all trained with the same objective function and differ only in their architecture.

5.2.2. Visual Comparison

Figure 6 shows reconstructions from all learned methods for an apple slice with bitter pit. The decrease in quality with the decrease in the number of angles is clearly visible. For 2 angles, none of the methods are able to accurately recover the shape of the apple. The iCTU-Net reconstruction has sharp edges for the 2-angle case, while the other methods produce blurry reconstructions.

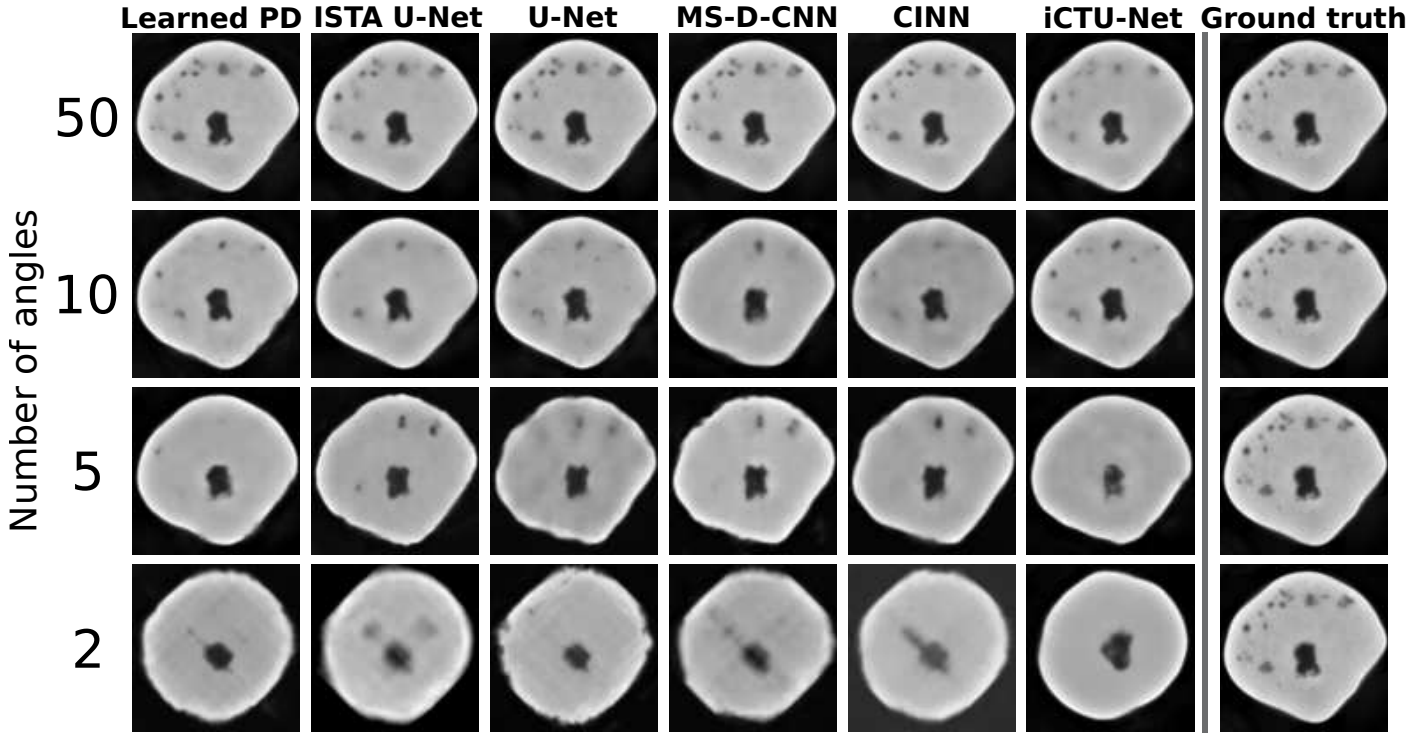

Figure 6. Visual overview of one apple slice with bitter pit for different learned methods. Evaluated on Gaussian noise. The quality of the reconstruction deteriorates very quickly for a reduced number of angles. For the 2-angle case, none of the methods can reconstruct the exact shape of the apple.

The inner structure, including the defects, is accurately reconstructed for 50 angles by all methods. The only exception is the iCTU-Net. Reconstructions from this network show a smooth interior of the apple. The other methods also result in the disappearance of smaller defects with fewer measurement angles. Nonetheless, a defect-detection system might still be able to sort out the apple based on the 5-angle reconstructions. The 2-angle case can be used to assess failure modes of the different approaches. The undersampling case is so severe that a lot of information is lost. However, the iCTU-Net is able to produce

a smooth image of an apple, but it has few similarities with the ground truth apple. It appears that the models have memorized the roundness of an apple and produce a round apple that has little in common with the real apple except for its size and core.

5.2.3. Data Consistency

The data consistency is evaluated for all three Apple CT datasets. The MSE is used to measure the discrepancy. It is the canonical choice for measurements with Gaussian noise (cf. Section 4.1.3). Table A6 in the appendix contains all MSE values and standard deviations. Figure 7 shows the results depending on the number of angles for the noise-free and Gaussian noise dataset.

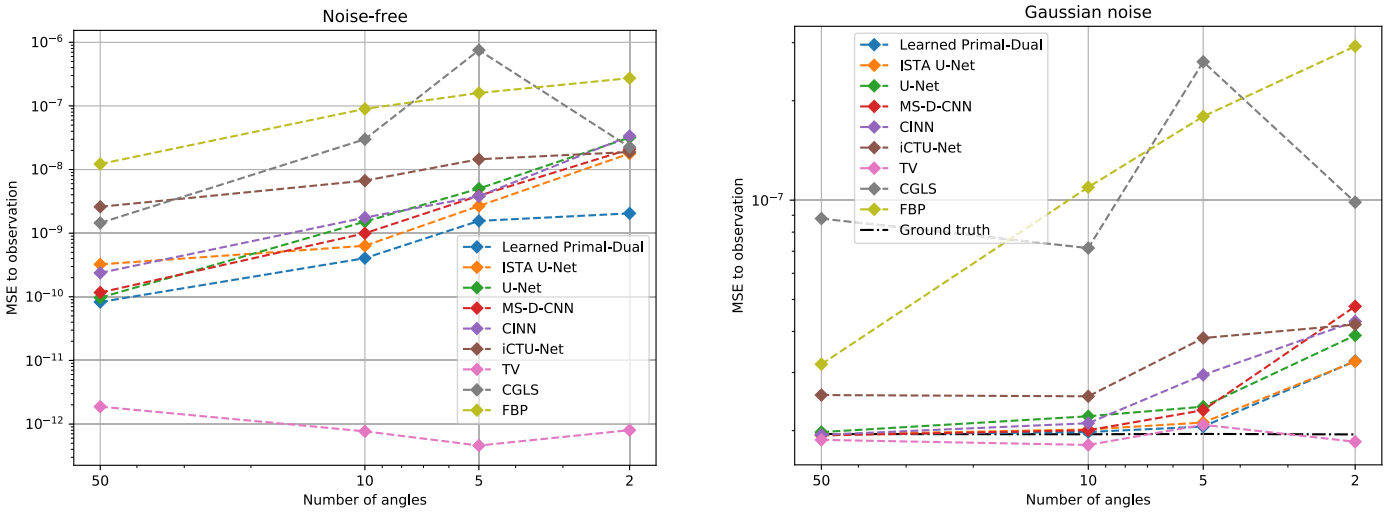

Figure 7. Mean squared error (MSE) data discrepancy between the measurements and the forward-projected reconstructions for the noise-free (**left**) and Gaussian noise (**right**) dataset. The MSE values are plotted against the number of angles used for the reconstruction. For the Gaussian dataset, the mean data discrepancy between noisy and noise-free measurements is given for reference. Evaluation is done on 100 Apple CT test images. See Table A6 for the exact values.

In the noise-free setup, the optimal MSE value is zero. Nonetheless, an optimal data consistency does not correspond to perfect reconstructions in this case. Due to the undersampling of the measurements, the discretized linear forward operator A has a non-trivial null space, that is, $\tilde{x} \in X$, apart from $\tilde{x} = 0$, for which $A\tilde{x} = 0$. Any element from the null space can be added to the true solution $x^\dagger$ without changing the data discrepancy

$$A\left(x^\dagger + \tilde{x}\right) = Ax^\dagger + A\tilde{x} = Ax^\dagger + 0 = Ax^\dagger = y.$$

In the Gaussian setup, the MSE between noise-free and noisy measurements is used as a reference for a good data discrepancy. The problem from the undersampling is also relevant in this setting.

Both setups show an increase in the data discrepancy with fewer measurement angles. The reason for the increase is presumably the growing number of deviations in the reconstructions. In the Gaussian noise setup, the high data discrepancy of all learned methods for 2 angles coincides with the poor reconstructions of the apple slice in Figure 6. Only the TV method, which enforces data consistency during the reconstruction, keeps a constant level. The main problem for this approach are the ambiguous solutions due to the undersampling. The TV method is not able to identify the correct solution given by the ground truth. Therefore, the PSNR and SSIM values are also decreasing.

Likewise, the data consistency was analyzed for the dataset with scattering noise. The MSE values of all learned methods are close to the empirical noise level. In contrast, FBP and TV have a much smaller discrepancy. Therefore, their reconstructions are most likely

influenced by the scattering noise. An effect that is also reflected in the PSNR and SSIM values in Table 4.

6. Discussion

Among all the methods we compared, there is no definite winner that is the best on both LoDoPaB-CT and Apple CT. Learned primal-dual, as an example of a learned iterative method, is the best method on LoDoPaB-CT, in terms of both PSNR and SSIM, and also gives promising results on Apple CT. However, it should be noted that the differences in performance between the learned methods are relatively small. The ISTA U-Net, second place in terms of PSNR on LoDoPaB-CT, scores only 0.14 dB less than learned primal-dual. The performance in terms of SSIM is even closer on LoDoPaB-CT. The best performing learned method resulted in an SSIM that was only 0.022 higher than the last placed learned method. The observation that the top scoring learned methods did not differ greatly in terms of performance has also been noted in the fastMRI challenge [61]. In addition to the performance of the learned methods, other characteristics are also of interest.

6.1. Computational Requirements and Reconstruction Speed

When discussing the computational requirements of deep learning methods, it is important to distinguish between training and inference. Training usually requires significantly more processing power and memory. All outputs of intermediate layers have to be stored for the determination of the gradients during backpropagation. Inference is much faster and less resource-intensive. In both cases, the requirements are directly influenced by image size, network architecture and batch size.

A key feature and advantage of the learned iterative methods, post-processing methods and fully-learned approaches is the speed of reconstruction. Once the network is trained, the reconstruction can be obtained by a simple forward pass of the model. Since the CINN, being a generative model, draws samples from the posterior distribution, many forward passes are necessary to well approximate the mean or other moments. Therefore, the quality of the reconstruction may depend on the number of forward passes [48]. The DIP + TV method requires a separate model to be trained to obtain a reconstruction. As a result, reconstruction is very time-consuming and resource-intensive, especially on the 972 px $\times$ 972 px images in the Apple CT datasets. However, DIP + TV does not rely on a large, well-curated dataset of ground truth images and measurements. As an unsupervised method, only measurement data is necessary. The large size of the Apple CT images is also an issue for the other methods. In comparison to LoDoPaB-CT, the batch size had to be reduced significantly in order to train the learned models. This small batch size can cause instability in the training process, especially for CINN (cf. Figure A14).

Transfer to 3D Reconstruction

The reconstruction methods included in this study were evaluated based on the reconstruction of individual 2D slices. In real applications, however, the goal is often to obtain a 3D reconstruction of the volume. This can be realized with separate reconstructions of 2D slices, but (learned) methods might benefit from additional spatial information. On the other hand, a direct 3D reconstruction can have a high demand on the required computing power. This is especially valid when training neural networks.

One way to significantly reduce the memory consumption of backpropagation is to use invertible neural networks (INN). Due to the invertibility, the intermediate activations can be calculated directly and do not have to be stored in memory. INNs were successfully applied to 3D reconstructions tasks in MRI [62] and CT [63]. The CINN approach from our comparison can be adapted in a similar way for 3D data. In most post-processing methods, the U-Net can be replaced by an invertible iUnet, as proposed by Etmann et al. [63].

Another option is the simultaneous reconstruction of only a part of the volume. The information from multiple neighboring slices is used in this case, which is also referred to as 2.5D reconstruction. Networks that operate on this scenario usually have a mixture

of 2D and 3D convolutional layers [64]. The goal is to strike a balance between the speed and memory advantage of the 2D scenario and the additional information from the third dimension. All deep learning methods included in this study would be suitable for 2.5D reconstruction with slight modifications to their network architecture.

Overall, 2.5D reconstruction can be seen as an intermediate step that can already be realized with many learned methods. The pure 3D case, on the other hand, requires specially adapted deep learning approaches. Technical innovations such as mixed floating point precision and increasing computing power may facilitate the transition in the coming years.

6.2. Impact of the Datasets

The type, composition and size of a dataset can have direct impact on the performance of the models. The observed effects can provide insight into how the models can be improved or how the results translate to other datasets.

6.2.1. Number of Training Samples

A large dataset is often required to successfully train deep learning methods. In order to assess the impact of the number of data pairs on the performance of the methods, we consider the Apple CT datasets. The scattering noise dataset (Dataset C), with 5280 training images, is only about 10% as large as the noise free dataset (Dataset A) and the Gaussian noise dataset (Dataset B). Here it can be noted that the iCTU net, as an example of a fully learned approach, performs significantly worse on this smaller dataset than on dataset A and dataset B (26.26 dB PSNR on Dataset C with 50 angles, 36.07 dB and 32.90 dB on Dataset A and Dataset B with 50 angles, respectively). This drop in performance could also be caused by the noise case. However, Baguer et al. [34] have already noted in their work that the performance of fully learned approaches heavily depends on the number of training images. This could be explained by the fact that fully learned methods need to infer most of the information about the inversion process purely from data. Unlike learned iterative methods, such as learned primal-dual, fully learned approaches do not incorporate the physical model. A drop in performance due to a smaller training set was not observed for the other learned methods. However, 5280 training images is still comprehensive. Baguer et al. [34] also investigated the low-data regime on LoDoPaB-CT, down to around 30 training samples. In their experiments, learned primal-dual worked well in this scenario, but was surpassed by the DIP + TV approach. The U-Net post-processing lined up between learned Primal-Dual and the fully learned method. Therefore, the amount of available training data should be considered when choosing a model. To enlarge the training set, the DIP + TV approach can also be used to generate pseudo ground truth data. Afterwards, a supervised method with a fast reconstruction speed can be trained to mimic the behavior of DIP + TV.

6.2.2. Observations on LoDoPaB-CT and Apple CT

The samples and CT setups differ greatly between the two datasets. The reconstructions obtained using the methods compared in this study reflect these differences to some extent, but there were also some effects that were observed for both datasets.

The sample reconstructions in Figures 4 and 6 show that most learned methods produce smooth images. The same observation can be made for TV, where smoothness is an integral part of the modeling. An extension by a suitable regularization can help to preserve edges in the reconstruction without the loss of small details, or the introduction of additional noise. One possibility is to use diffusion filtering [65], for example, variants of the Perona-Malik diffusion [66] in this role. Diffusion filtering was also successfully applied as a post-processing step for CT [67]. Whether smoothness of reconstructions is desired depends on the application and further use of the images, for example, visual or computer-aided diagnosis, screening, treatment planning, or abnormality detection. For the apple scans, a subsequent task could be the detection of internal defects for sorting them into different grades. The quality of the reconstructions deteriorates with the decreasing

number of measurement angles. Due to increasing undersampling, the methods have to interpolate more and more information to find an adequate solution. The model output is thereby influenced by the training dataset.

The effects of severe undersampling can be observed in the 2-angle setup in Figure 6. All reconstructions of the test sample show a prototypical apple with a round shape and a core in the center. The internal defects are not reproduced. One explanation is that supervised training aims to minimize the empirical risk on the ground truth images. Therefore, only memorizing and reconstructing common features in the dataset, like the roundness and the core, can be optimal in some ways to minimize the empirical risk on severely undersampled training data. Abnormalities in the data, such as internal defects, are not captured in this case. This effect is subsequently transferred to the reconstruction of test data. Hence, special attention should be paid to the composition of the training data. As shown in the next Section 6.2.3, this is particularly important when the specific features of interest are not well represented in the training set.

In the 5-angle setup, all methods are able to accurately reconstruct the shape of the apple. Internal defects are partially recovered only by the post-processing methods and the CINN. These approaches all use FBP reconstructions as a starting point. Therefore, they rely on the information that is extracted by the FBP. This can be useful in the case of defects but aggravating for artifacts in the FBP reconstruction. The CINN approach has the advantage of sampling from the space of possible solutions and the evaluability of the likelihood under the model. This information can help to decide whether objects in the reconstruction are really present.

In contrast, Learned Primal-Dual and the iCTU-Net work directly on the measurements. They are more flexible with respect to the extraction of information. However, this also means that the training objective strongly influences which aspects of the measurements are important for the model. Tweaking the objective or combining the training of a reconstruction and a detection model, that is, end-to-end learning or task-driven reconstruction, might be able to increase the model performance in certain applications [68,69].

6.2.3. Robustness to Changes in the Scanning Setup

A known attribute of learned methods is that they can often only be applied to data similar to the training data. It is often unclear how a method trained in one setting generalizes to a different setting. In CT, such a situation could for example arise due to altered scan acquisition settings or application to other body regions. Switching between CT devices from different manufacturers can also have an impact.

As an example, we evaluated the U-Net on a different number of angles than it was trained on. The results of this experiment are shown in Table 5. In most setups the PSNR drops by at least 10 dB when evaluated on a different setting. In practice, the angular sampling pattern may change and it would be cumbersome to train a separate model for each pattern.

Table 5. Performance of a U-Net trained on the Apple CT dataset (scattering noise) and evaluated on different angular samplings. In general, a U-Net trained on a specific number of angles fails to produce good results on a different number of angles. PSNR and SSIM are calculated with image-dependent data range.

Training \ Evaluation	50 Angles		10 Angles		5 Angles		2 Angles	
	PSNR	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**	**PSNR**	**SSIM**
50 angles	39.62	0.913	16.39	0.457	11.93	0.359	8.760	0.252
10 angles	27.59	0.689	33.51	0.803	18.44	0.607	9.220	0.394
5 angles	24.51	0.708	26.19	0.736	27.77	0.803	11.85	0.549
2 angles	15.57	0.487	14.59	0.440	15.94	0.514	19.78	0.676

6.2.4. Generalization to Other CT Setups

The LoDoPaB-CT and Apple CT datasets were acquired by simulating parallel-beam measurements, based on the Radon transform. This setup facilitates large-scale experiments with many example images, whereas the underlying operators in the algorithms have straightforward generalizations to other geometries. Real-world applications of CT are typically more complex. For example, the standard scanning geometries in medical applications are helical fan-beam or cone-beam [36]. In addition, the simulation model does not cover all physical effects that may occur during scanning. For this reason, the results can only be indicative of performance on real data.

However, learned methods are known to adapt well to other setups when retrained from scratch on new samples. It is often not necessary to adjust the architecture for this purpose, other than by replacing the forward operator and its adjoint where they are involved. For example, most learned methods show good performance on the scattering observations, whereas the classical methods perform worse compared to the Gaussian noise setup. This can be explained by the fact that the effect of scattering is structured, which, although adding to the instability of the reconstruction problem, can be learned to be (partially) compensated for. In contrast, classical methods require the reconstruction model to be manually adjusted in order to incorporate knowledge about the scattering. If scattering is treated like an unknown distortion (i.e., a kind of noise), such as in our comparison, the classical assumption of pixel-wise independence of the noise is violated by the non-local structure of the scattering. Convolutional neural networks are able to capture these non-local effects.

6.3. Conformance of Image Quality Scores and Requirements in Real Applications

The goal in tomographic imaging is to provide the expert with adequate information through a clearly interpretable reconstructed image. In a medical setting, this can be an accurate diagnosis or plan for an operation; and in an industrial setting, the image may be used for detection and identification of faults or defects as part of quality control.

PSNR and SSIM, among other image quality metrics, are commonly used in publications and data challenges [61] to evaluate the quality of reconstructed medical images [70]. However, there can be cases in which PSNR and SSIM are in a disagreement. Although not a huge difference, the results given in Table 4 are a good example of this. This often leads to the discussion of which metric is better suited for a certain application. The PSNR expresses a pixel-wise difference between the reconstructed image and its ground truth, whereas the SSIM checks for local structural similarities (cf. Section 4.1). A common issue with both metrics is that a local inaccuracy in the reconstructed image, such as a small artifact, would only have a minor influence on the final assessment. The effect of the artifact is further downplayed when the PSNR or SSIM values are averaged over the test samples. This is evident in some reconstructions from the DIP + TV approach, where an artifact was observed on multiple LoDoPaB-CT reconstructions whereas this is not reflected in the metrics. This artifact is highlighted with a red circle in the DIP + TV reconstruction in Figure A9.

An alternative or supporting metric to PSNR and SSIM is visual inspection of the reconstructions. A visual evaluation can be done, for example, through a blind study with assessments and rating of reconstructions by (medical) experts. However, due to the large amount of work involved, the scope of such an evaluation is often limited. The 2016 Low Dose CT Grand Challenge [9] based their comparison on the visibility of liver lesions, as evaluated by a group of physicians. Each physician had to rate 20 different cases. The fastMRI Challenge [61] employed radiologists to rank MRI reconstructions. The authors were able to draw parallels between the quantitative and blind study results, which revealed that, in their data challenge, SSIM was a reasonable estimate for the radiologists' ranking of the images. In contrast, Mason et al. [71] found differences in their study between several image metrics and experts' opinions on reconstructed MRI images.

In industrial settings, PSNR or related pixel-based image quality metrics fall short on assessing the accuracy or performance of a reconstruction method when physical and hardware-related factors in data acquisition play a role in the final reconstruction. These factors are not accurately reflected in the image quality metrics, and therefore the conclusions drawn may not always be applicable. An alternative practice is suggested in [72], in which reconstructions of a pack of glass beads are evaluated using pixel-based metrics, such as contrast-to-noise ratio (CNR), and pre-determined physical quantification techniques. The physical quantification is object-specific, and assessment is done by extracting a physical quality of the object and comparing this to a reference size or shape. In one of the case studies, the CNR values of iterated reconstructions suggest an earlier stopping for the best contrast in the image, whereas a visual inspection reveals the image with the "best contrast" to be too blurry and the bead un-segmentable. The Apple CT reconstructions can be assessed in a similar fashion, where we look at the overall shape of a healthy apple, as well as the shape and position of its pit.

6.4. Impact of Data Consistency

Checking the discrepancy between measurement and forward-projected reconstruction can provide additional insight into the quality of the reconstruction. Ground truth data is not needed in this case. However, an accurate model $\mathcal{A}$ of the measurement process must be known. Additionally, the evaluation must take into account the noise type and level, as well as the sampling ratio.

Out of all tested methods, only the TV, CGLS and DIP + TV approach use the discrepancy to the measurements as (part of) their minimization objective for the reconstruction process. Still, the experiments on LoDoPaB-CT and Apple CT showed data consistency on the test samples for most of the methods. Based on these observations, data consistency does not appear to be a problem with test samples coming from a comparable distribution to the training data. However, altering the scan setup can significantly reduce the reconstruction performance of learned methods (cf. Section 6.2.3). Verification of the data consistency can serve as an indicator without the need for ground truth data or continuous visual inspection.

Another problem can be the instability of some learned methods, which is also known under the generic term of adversarial attacks [73]. Recent works [74,75] show that some methods, for example, fully learned and post-processing approaches, can be unstable. Tiny perturbations in the measurements may result in severe artifacts in the reconstructions. Checking the data discrepancy may also help in this case. Nonetheless, severe artifacts were also found in some reconstructions from the DIP + TV method on LoDoPaB-CT.

All in all, including a data consistency objective in training (bi-directional loss), could further improve the results from learned approaches. Checking the discrepancy during the application of trained models can also provide additional confidence about the reconstructions' accuracy.

6.5. Recommendations and Future Work

As many learned methods demonstrated similar performance in both low-dose CT and sparse-angle CT setups, further attributes have to be considered when selecting a learned method for a specific application. As discussed above, consideration should also be given to reconstruction speed, availability of training data, knowledge of the physical process, data consistency, and subsequent image analysis tasks. An overview can be found in Table 6. From the results of our comparison, some recommendations for the choice and further investigation of deep learning methods for CT reconstruction emerge.

Table 6. Summary of selected reconstruction method features. The reconstruction error ratings reflect the average performance improvement in terms of the evaluated metrics PSNR and SSIM compared to filtered back-projection (FBP). Specifically, for LoDoPaB-CT improvement quotients are calculated for PSNR and SSIM, and the two are averaged; for the Apple CT experiments the quotients are determined by first averaging PSNR and SSIM values within each noise setting over the four angular sampling cases, next computing improvement quotients independently for all three noise settings and for PSNR and SSIM, and finally averaging over these six quotients. GPU memory values are compared for 1-sample batches.

Model	Reconstruction Error (Image Metrics)		Training Time	Reconstruction Time	GPU Memory	Learned Parameters	Uses $\mathcal{D}_Y$ Discrepancy	Operator Required
Learned P.-D.	⋆⋆	⋆	⋆⋆⋆⋆	⋆⋆	⋆⋆⋆⋆	⋆⋆	no	⋆⋆⋆
ISTA U-Net	⋆⋆	⋆	⋆⋆⋆	⋆⋆	⋆⋆⋆	⋆⋆⋆	no	⋆⋆
U-Net	⋆⋆	⋆	⋆⋆	⋆⋆	⋆⋆	⋆⋆	no	⋆⋆
MS-D-CNN	⋆⋆	⋆	⋆⋆⋆⋆	⋆⋆	⋆⋆	⋆	no	⋆⋆
U-Net++	⋆⋆	-	⋆⋆	⋆⋆	⋆⋆⋆	⋆⋆⋆	no	⋆⋆
CINN	⋆⋆	⋆	⋆⋆	⋆⋆⋆	⋆⋆⋆	⋆⋆⋆	no	⋆⋆
DIP + TV	⋆⋆⋆	-	-	⋆⋆⋆⋆	⋆⋆	3+	yes	⋆⋆⋆⋆
iCTU-Net	⋆⋆⋆	⋆⋆	⋆⋆	⋆⋆	⋆⋆⋆	⋆⋆⋆⋆	no	⋆
TV	⋆⋆⋆	⋆⋆⋆	-	⋆⋆⋆	⋆	3	yes	⋆⋆⋆⋆
CGLS	-	⋆⋆⋆⋆	-	⋆	⋆	1	yes	⋆⋆⋆⋆
FBP	⋆⋆⋆⋆	⋆⋆⋆⋆	-	⋆	⋆	2	no	⋆⋆⋆⋆
Legend	LoDoPaB Avg. improv. over FBP	Apple CT	Rough values for Apple CT Dataset B (varying for different setups and datasets)					
⋆⋆⋆⋆	0%	0–15%	>2 weeks	>10 min	>10 GiB	$>10^8$		Direct
⋆⋆⋆	12–16%	25–30%	>5 days	>30 s	>3 GiB	$>10^6$		In network
⋆⋆	17–20%	40–45%	>1 day	>0.1 s	>1.5 GiB	$>10^5$		For input
⋆		50–60%		≤0.02 s	≤1 GiB	$\leq 10^5$		Only concept

Overall, the learned primal-dual approach proved to be a solid choice on the tested low photon count and sparse-angle datasets. The applicability of the method depends on the availability and fast evaluation of the forward and the adjoint operators. Both requirements were met for the 2D parallel beam simulation setup considered. However, without adjustments to the architecture, more complicated measurement procedures and especially 3D reconstruction could prove challenging. In contrast, the post-processing methods are more flexible, as they only rely on some (fast) initial reconstruction method. The performance of the included post-processing models was comparable to learned primal-dual. A disadvantage is the dependence on the information provided by the initial reconstruction.

The other methods included in this study are best suited for specific applications due to their characteristics. Fully learned methods do not require knowledge about the forward operator, but the necessary amount of training data is not available in many cases. The DIP + TV approach is on the other side of the spectrum, as it does not need any ground truth data. One downside is the slow reconstruction speed. However, faster reconstruction methods can be trained based on pseudo ground truth data created by DIP + TV. The CINN method allows for the evaluation of the likelihood of a reconstruction and can provide additional statistics from the sampling process. The invertible network architecture also enables the model to be trained in a memory-efficient way. The observed performance for 1000 samples per reconstruction was comparable to the post-processing methods. For time-critical applications, the number of samples would need to be lowered considerably, which can deteriorate the image quality.

In addition to the choice of model, the composition and amount of the training data also plays a significant role for supervised deep learning methods. The general difficulty of application to data that deviate from the training scenario was also observed in our comparison. Therefore, the training set should either contain examples of all expected cases or the model must be modified to include guarantees to work in divergent scenarios,

such as different noise levels or number of angles. Special attention should also be directed to subsequent tasks. Adjusting the training objective or combining training with successive detection models can further increase the value of the reconstruction. Additionally, incorporating checks for the data consistency during training and/or reconstruction can help to detect and potentially prevent deviations in reconstruction quality. This potential is currently underutilized by many methods and could be a future improvement. Furthermore, the potential of additional regularization techniques to reduce the smoothness of reconstructions from learned methods should be investigated.

Our comparison lays the foundation for further research that is closer to real-world applications. Important points are the refinement of the simulation model, the use of real measurement data and the transition to fan-beam/cone-beam geometries. The move to 3D reconstruction techniques and the study of the influence of the additional spatial information is also an interesting aspect. Besides the refinement of the low photon count and sparse-angle setup, a future comparison should include limited-angle CT. A first application of this setting to Apple CT can be found in the dataset descriptor [38].

An important aspect of the comparison was the use of PSNR and SSIM image quality metrics to rate the produced reconstructions. In the future, this assessment should be supplemented by an additional evaluation of the reconstruction quality of some samples by (medical) professionals. A multi-stage blind study for the evaluation of unmarked reconstructions, including or excluding the (un)marked ground truth image, may provide additional insights.

Finally, a comparison is directly influenced by the selection of the included models. While we tested a broad range of different methods, there are still many missing types, for example, learned regularization [18] and null space networks [76]. We encourage readers to test additional reconstruction methods on the datasets from our comparison and submit reconstructions to the respective data challenge websites: (https://lodopab.grand-challenge.org/, last accessed: 1 March 2021) and (https://apples-ct.grand-challenge.org/, last accessed: 1 March 2021).

7. Conclusions

The goal of this work is to quantitatively compare learned, data-driven methods for image reconstruction. For this purpose, we organized two online data challenges, including a 10-day *kick-off event*, to give experts in this field the opportunity to benchmark their methods. In addition to this event, we evaluated some popular learned models independently. The appendix includes a thorough explanation and references to the methods used. We focused on two important applications of CT. With the LoDoPaB-CT dataset we simulated low-dose measurements and with the Apple CT datasets we included several sparse-angle setups. In order to ensure reproducibility, the source code of the methods, network parameters and the individual reconstruction are released. In comparison to the classical baseline (FBP and TV regularization) the data-driven methods are able to improve the quality of the CT reconstruction in both sparse-angle and low-dose settings. We observe that the top scoring methods, namely learned primal-dual and different post-processing approaches, perform similarly well in a variety of settings. Besides that, the applicability of deep learning-based models depends on the availability of training examples, prior knowledge about the physical system and requirements for the reconstruction speed.

Supplementary Materials: The following are available online at https://zenodo.org/record/4460055#.YD9IiIsRVPZ; https://zenodo.org/record/4459962#.YD9IqIsRVPZ; https://zenodo.org/record/4459250#.YD9GtU5xdPY.

Author Contributions: Conceptualization, J.L., M.S., P.S.G., P.M. and M.v.E.; Data curation, J.L., V.A. and S.B.C.; Formal analysis, J.L., M.S., P.S.G., V.A., S.B.C. and A.D.; Funding acquisition, K.J.B. and P.M.; Investigation, J.L., M.S., V.A., S.B.C. and A.D.; Project administration, M.S.; Software, J.L., M.S., P.S.G., V.A., S.B.C., A.D., D.B., A.H. and M.v.E.; Supervision, K.J.B., P.M. and M.v.E.; Validation, J.L. and M.S.; Visualization, J.L. and A.D.; Writing—original draft, J.L., M.S., P.S.G., V.A., S.B.C., A.D., D.B., A.H. and M.v.E.; Writing—review & editing, J.L., M.S., P.S.G., V.A., S.B.C., A.D., D.B., A.H., K.J.B., P.M. and M.v.E. All authors have read and agreed to the published version of the manuscript.

Funding: J.L., M.S., A.D. and P.M. were funded by the German Research Foundation (DFG; GRK 2224/1). J.L. and M.S. additionally acknowledge support by the project DELETO funded by the Federal Ministry of Education and Research (BMBF, project number 05M20LBB). A.D. further acknowledges support by the Klaus Tschira Stiftung via the project MALDISTAR (project number 00.010.2019). P.S.G. was funded by The Marie Skłodowska-Curie Innovative Training Network MUMMERING (Grant Agreement No. 765604). S.B.C. was funded by The Netherlands Organisation for Scientific Research (NWO; project number 639.073.506). M.v.E. and K.J.B. acknowledge the financial support by Holland High Tech through the PPP allowance for research and development in the HTSM topsector.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The two datasets used in this study are the LoDoPaB-CT dataset [32], and the AppleCT dataset [37], both publicly available on Zenodo. The reconstructions discussed in Section 5 are provided as supplementary materials to this submission. These are shared via Zenodo through [57–59].

Acknowledgments: We are grateful for the help of GREEFA b.v. and the FleX-ray Laboratory of CWI for making CT scans of apples with internal defects available for the Code Sprint. The Code Sprint was supported by the DFG and the European Commission's MUMMERING ITN. We would like to thank Jens Behrmann for fruitful discussions. Finally, we would like to thank all participants of the Code Sprint 2020 for contributing to the general ideas and algorithms discussed in this paper.

Conflicts of Interest: The authors declare no conflict of interest, financial or otherwise.

Appendix A. Learned Reconstruction Methods

Appendix A.1. Learned Primal-Dual

The *Learned Primal-Dual* algorithm is a learned iterative procedure to solve inverse problems [19]. A primal-dual scheme [77] is unrolled for a fixed number of steps and the proximal operators are replaced by neural networks (cf. Figure A1). This unrolled architecture is then trained using data pairs from measurements and ground truth reconstructions. The forward pass is given in Algorithm A1. In contrast to the regular primal-dual algorithm, the primal and the dual space are extended to allow memory between iterations:

$$x = [x_{(1)}, \dots, x_{(N_{\text{primal}})}] \in X^{N_{\text{primal}}},$$
$$h = [h_{(1)}, \dots, h_{(N_{\text{dual}})}] \in Y^{N_{\text{dual}}}.$$

For the benchmark $N_{\text{primal}} = 5$ and $N_{\text{dual}} = 5$ was used. Both the primal and dual operators were parameterized as convolutional neural networks with 3 layers and 64 intermediate convolution channels. The primal-dual algorithm was unrolled for $K = 10$ iterations. Training was performed by minimizing the mean squared error loss using the Adam optimizer [78] with a learning rate of 0.0001. The model was trained for 10 epochs on LoDoPaB-CT and for at most 50 epochs on the apple data, whereby the model with the highest PSNR on the validation set was selected. Batch size 1 was used. Given a learned primal-dual algorithm the reconstruction can be obtained using Algorithm A1.

Algorithm A1 Learned Primal-Dual.

Given learned proximal dual and primal operators $\Gamma_{\theta_k^d}, \Lambda_{\theta_k^p}$ *for* $k = 1, \dots, K$ *the reconstruction from noisy measurements* y_δ *is calculated as follows.*

1. Initialize $x^{[0]} \in X^{N_{\text{primal}}}, h^{[0]} \in Y^{N_{\text{dual}}}$
2. **for** $k = 1 : K$
3. $\quad h^{[k]} = \Gamma_{\theta_k^d}\left(h^{[k-1]}, \mathcal{A}(x_{(2)}^{[k-1]}), y_\delta\right)$
4. $\quad x^{[k]} = \Lambda_{\theta_k^p}\left(x^{[k-1]}, \left[\mathcal{A}(x_{(1)}^{[k-1]})\right]^*(h_{(1)}^{[m]})\right)$
5. **end**
6. **return** $\hat{x} = x_{(1)}^{[K]}$

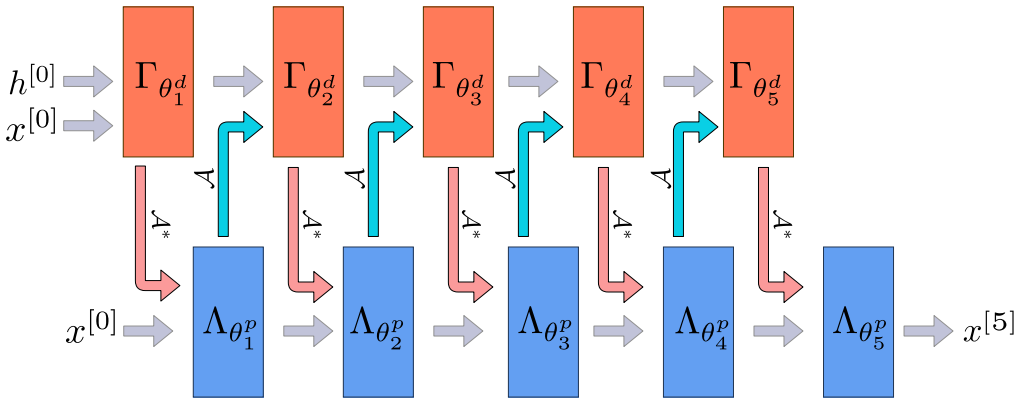

Figure A1. Architecture of the learned primal dual algorithm unrolled for $K = 5$ iterations. We used a zero initialization for $h^{[0]}$ and the FBP reconstruction for $x^{[0]}$. Adapted from [19].

Appendix A.2. U-Net

The goal of post-processing methods is to improve a pre-computed reconstruction. For CT, the FBP is used to obtain an initial reconstruction. This reconstruction is then used as an input to a post-processing network. For the enhancement of CT reconstructions, the post-processing network is implemented as a U-Net [20]. The U-Net architecture, as proposed by Ronneberger et al. [40], was originally designed for the task of semantic segmentation, but has many properties that are also beneficial for denoising. The general architecture is shown in Figure A2. In our implementation we used 5 scales (4 up- and downsampling blocks each) both for the LoDoPaB-CT and the Apple CT datasets. The skip connection between same scale levels mitigates the vanishing gradient problem so that deeper networks can be trained. In addition, the multi-scale architecture can be considered as a decomposition of the input image, in which an optimal filtering can be learned for each scale. There are many extensions to this basic architecture. For example, the U-Net++ (cf. Appendix A.3) extends the skip connections to different pathways.

The used numbers of channels at the different scales are 32, 32, 64, 64, and 128. For all skip connections 4 channels were used. The input FBPs were computed with Hann filtering and no frequency scaling. Linear activation (i.e., no sigmoid or ReLU activation) was used for the network output. Training was performed by minimizing the mean squared error loss using the Adam optimizer. For each training, the model with the highest PSNR on the validation set was selected. Due to the different memory requirements imposed by the image sizes of LoDoPaB-CT and the Apple CT data, different batch sizes were used. While for LoDoPaB-CT the batch size was 32 and standard batch normalization was applied, for the Apple CT data a batch size of 4 was used and layer normalization was applied instead of batch normalization. On LoDoPaB-CT, the model was trained for 250 epochs

with learning rate starting from 0.001, reduced to 0.0001 by cosine annealing. On the Apple CT datasets, the model was trained for at most 50 epochs with fixed learning rate 0.001.

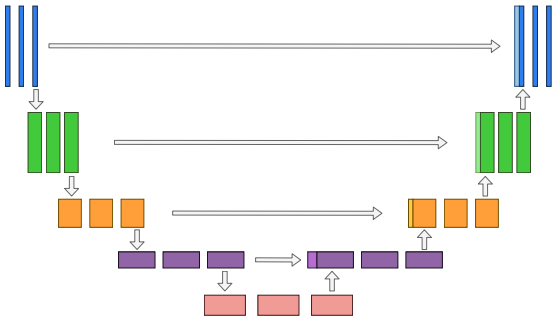

Figure A2. Architecture of the mutli-scale, post-processing U-Net. The general architecture of a U-Net consists on a downsampling path on the left and a upsampling path on the right with intermediate connection between similar scales. Adapted from [40].

Appendix A.3. U-Net++

The U-Net++ was introduced by Zhou et al. [41], the network improves on the original U-Net [40] architecture by incorporating nested and dense convolution blocks between skip connections. In U-Net, the down-sample block outputs of the encoder are directly input into the decoder's up-sample block at the same resolution. In U-Net++, the up-sampling block receives a concatenated input of a series of dense convolutional blocks at the same resolution. The input to these dense convolutional blocks is the concatenation of all previous dense convolutional blocks and the corresponding up-sample of a lower convolutional block.

The design is intended to convey similar semantic information across the skip-pathway. Zhou et al. suggest that U-Net's drawback is that the skip connections combine semantically dissimilar feature maps from the encoder and decoder. The results of these dissimilar semantic feature maps can limit the learning of the network. As a result, they proposed U-Net++ to address this drawback in the U-Net architecture. The purpose of the network is to progressively gain more fine-grained details from the nested dense convolutional blocks. Once these feature maps are combined with the decoder feature maps, it should, in theory, reduce the dissimilarity between the feature maps [41]. U-Net++ has shown to be successful in nodule segmentation of low-dose CT scans.

For our comparison on the LoDoPaB-CT dataset, we adopted a U-Net++ architecture with five levels, four down-samples reduced by a factor of 2 and four up-samples. The numbers of filters per convolutional block were 32, 64, 128, 256, 512 for the different levels, respectively. Each convolutional block contained two convolutional layers, each followed by batch normalization and ReLU activation. Input FBPs computed with Hann filtering and no frequency scaling were used. Linear activation (i.e., no sigmoid or ReLU activation) was used for the network output.

The loss function was chosen as a combination of MSE and SSIM,

$$\alpha\,\mathrm{MSE}(\hat{x}, x^{\dagger}) + (1-\alpha)(1-\mathrm{SSIM}(\hat{x}, x^{\dagger})).$$

Empirically, the mixed loss function with weighting of 0.35 and 0.65 for MSE and SSIM, respectively, provided the best results.

The optimizer used for this task was RMSprop [79] with a weight decay of 1×10^{-8} and momentum of 0.9. The model was trained for 8 epochs with a learning rate of 1×10^{-5} using a batch size of 4, and the model with the lowest loss on the validation set was selected.

Source code and model weights are publicly available in a github repository (https://github.com/amirfaraji/LowDoseCTPytorch, last accessed: 1 March 2021).

Appendix A.4. Mixed-Scale Dense Convolutional Neural Network

The Mixed-Scale Dense (MS-D) network architecture was introduced by Pelt & Sethian [21]. The main properties of the MS-D architecture are mixing scales in every layer and dense connection of all feature maps. Instead of downscaling and upscaling, features at different scales are captured with dilated convolutions, and multiple scales are used in each layer. All feature maps have the same size, and every layer can use all previously computed feature maps as an input. Thus, feature maps are maximally reused, and features do not have to be replicated in multiple layers to be used deeper in the network. The output image is computed based on all layers instead of only the last one.

The authors show that MS-D architecture can achieve results comparable to typical DCNN with fewer feature maps and trainable parameters. This enables training with smaller datasets, which is highly important for CT. Furthermore, accurate results can usually be achieved without fine-tuning hyperparameters, and the same network architecture can often be used for different problems. A small number of feature maps leads to less memory usage in comparison with typical DCNN and enables training with larger images.

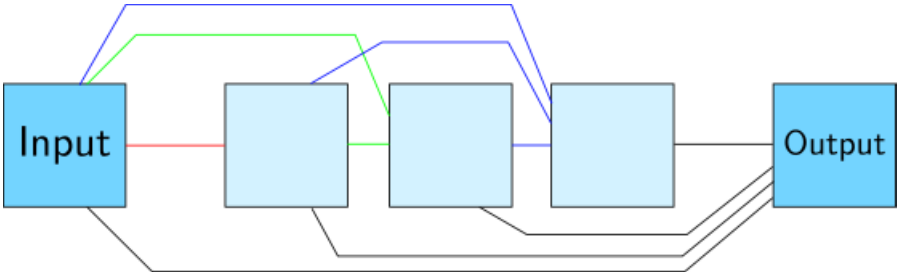

Figure A3. Architecture of the MS-D neural network for width of 1 and depth of 3, feature maps are drawn as light blue squares. Colored lines represent dilated convolutions, different colors correspond to different dilation values. Black lines represent 1×1 convolutions that connect the input and all feature maps to the output image. Adapted from [21].

The networks used equally distributed dilations with intervals from 1 to 10. The depth was 200 layers for the LoDoPaB-CT dataset and 100 layers for the Apple CT datasets. For the input FBPs, Hann filtering and no frequency scaling were used. The training was performed by minimizing MSE loss using the Adam optimizer with a learning rate of 0.001, using batch size 1. The model was trained for 15 epochs on LoDoPaB-CT and for at most 50 epochs on the apple data, whereby the model with the highest PSNR on the validation set was selected. Data augmentation consisting of rotations and flips was used for the apple data, but not for LoDoPaB-CT.

Appendix A.5. Conditional Invertible Neural Networks

Conditional invertible neural networks (CINN) are a relatively new approach for solving inverse problems [47,80]. Models of this type consist of two network parts (cf. Figure A4). An invertible network F represents a learned transformation between the (unknown) distribution $\mathcal{X}$ of the ground truth data and a standard probability distribution $\mathcal{Z}$, e.g., a Gaussian distribution. The second building block is a conditioning network C, which includes physical knowledge about the problem and encodes information from the measured data as an additional input to F.

A CINN was successfully applied to the task of low-dose CT reconstruction by Denker et al. [48]. Their model uses a multi-scale convolutional architecture as proposed in [81] and is built upon the FrEIA (https://github.com/VLL-HD/FrEIA, last accessed: 1 March 2021) python library. For the experiments in this paper, several improvements over the design in [48] are incorporated. The structure of the invertible network F and the conditioning network C are simplified. Using additive coupling layers [82] with Activation Normalization [83] improves stability of the training. Replacing downsampling operations with a learned version from Etmann et al. [63] prevents checkerboard artifacts and enhances

the overall reconstruction quality. In addition, the negative log-likelihood (NLL) loss is combined with a weighted mean-squared error (MSE) term

$$\min_{\Theta} \left[\log p_{\mathcal{Z}} \left(F_\Theta \left(x^\dagger, C_\Theta(y_\delta) \right) \right) + \log \left| \det \left(\mathrm{J}_{F_\Theta} \left(x^\dagger, C_\Theta(y_\delta) \right) \right) \right| + \alpha \, \mathrm{MSE} \left(F_\Theta^{-1}(z, C_\Theta(y_\delta)), x^\dagger \right) \right].$$

The applied network has 5 different downsampling scales, where both spatial dimensions are reduced by factor 2. Simultaneously, the number of channels increases by a factor of 4, making the operation invertible. After each downsampling step, half the channels are split of and send directly to the output layer. In total, the network has around 6.5 million parameters. It is trained with the Adam optimizer and a learning rate of 0.0005 for at most 200 epochs using batch size 4 (per GPU) on LoDoPaB-CT and for at most 32 epochs using batch size 3 on the apple data. The best model according to the validation loss is selected. A Gaussian distribution is chosen for $\mathcal{Z}$. The MSE weight is set to $\alpha = 1.0$. After training, the reconstructions are generated as a conditioned mean over $K = 1000$ sample reconstructions from the Gaussian distribution (cf. Algorithm A2).

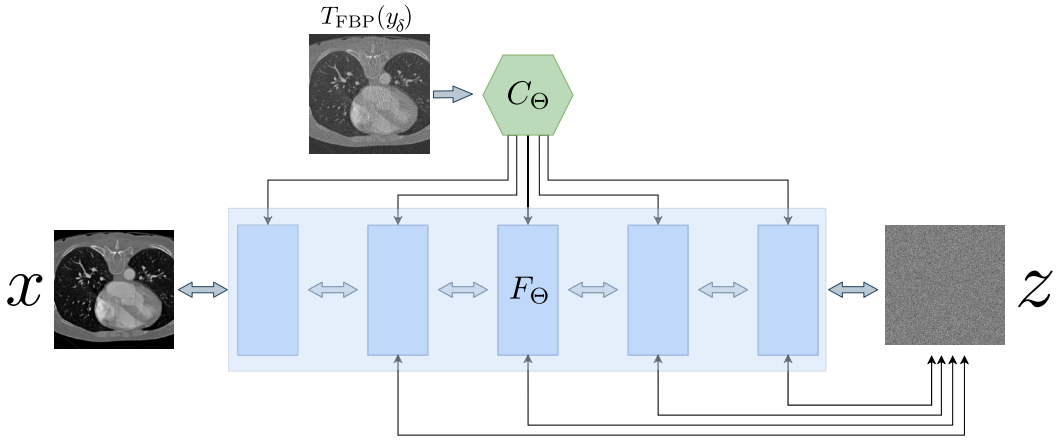

Figure A4. Architecture of the conditional invertible neural network. The ground truth image x is transformed by F_Θ to a Gaussian distributed z. Adapted from [48].

Algorithm A2 Conditional Invertible Neural Network (CINN).

Given a noisy measurement, y_δ, an invertible neural network F and a conditioning network C. Let $K \in \mathbb{N}$ be the number of random samples that should be drawn from a normal distribution $\mathcal{N}(0, \mathrm{I})$. The algorithm calculates the mean and variance of the conditioned reconstructions.

1. Calculate FBP: $c_0 = \mathcal{T}_{\mathrm{FBP}}(y_\delta)$.
2. Calculate outputs of the conditioning: $c = C_\Theta(c_0)$
3. **for** $k = 1 : K$
4. $\quad z^{[k]} \sim \mathcal{N}(0, \mathrm{I})$
5. $\quad \hat{x}^{[k]} = F^{-1}\left(z^{[k]}, c\right)$
6. **end**
7. Calculate mean: $\hat{x} = \frac{1}{K} \sum_k \hat{x}^{[k]}$
8. Calculate variance: $\hat{\sigma} = \frac{1}{K} \sum_k \left(\hat{x}^{[k]} - \hat{x} \right)^2$

Appendix A.6. ISTA U-Net

The ISTA U-Net [42] is a relatively new approach based on the encoder-decoder structure of the original U-Net. The authors draw parallels from the supervised training

of U-Nets to task-driven dictionary learning and sparse coding. For the ISTA U-Net the encoder is replaced by a sparse representation of the input vector and the decoder is linearized by removing all non-linearities, batch normalization and additive biases (cf. Figure A5). Given a data set of measurements and ground truth pairs $\{y_{\delta i}, x_i^\dagger\}_{i=1}^M$ the training problem can be formulated as a bi-level optimization problem

$$\min_{\{\theta,\gamma\},\lambda>0} \frac{1}{M}\sum_{i=1}^{M}\frac{1}{2}\|D_\gamma \alpha_{y_{\delta i},\theta} - x_i^\dagger\|_2^2$$
$$\text{where } \alpha_{y_{\delta i},\theta} = \arg\min_{\alpha\geq 0}\frac{1}{2}\|D_\theta \alpha - y_{\delta i}\|_2^2 + \|\lambda \odot \alpha\|_1,$$

where $\odot$ denotes the Hadamard product. Using an encoder dictionary D_θ the corresponding sparse code α_θ can be determined with the iterative thresholding algorithm (ISTA, [84]) with an additional non-negativity constraint for the sparse code. Liu et al. [42] use a learned variant of ISTA, called LISTA [85], to compute the sparse code. LISTA works by unrolling ISTA for a fixed number of K iterations

$$\alpha_{y_\delta,\theta}^{[k]} = \mathrm{ReLU}\left(\alpha_{y_\delta,\theta}^{[k-1]} + \eta D_\kappa^T\left(y_\delta - D_\theta \alpha_{y_\delta,\theta}^{[k-1]}\right) - \eta\lambda\right),$$

with $k = 1,\ldots,K$. In their framework they additionally untie the parameters for D_κ and D_θ, although both dictionaries have the same structure. The forward pass of the network is given in Algorithm A3.

For all experiments, $K = 5$ unrolled ISTA iterations were used. On LoDoPaB-CT, five scales with hidden layer widths 1024, 512, 256, 128, 64 were used and the lasso parameters λ were initialized with 10^{-3}. For the Apple CT datasets, the network appeared to be relatively sensitive with respect to the hyperparameter choices. For the noise-free data (Dataset A), five scales with hidden layer widths 512, 256, 128, 64, 32 were used and λ was initialized with 10^{-5}. For Datasets B and C, six scales, but less wide hidden layers, namely 512, 256, 128, 64, 32, 16, were used and λ was initialized with 10^{-4}. In all experiments, input FBPs computed with Hann filtering and no frequency scaling were used. A ReLU activation was applied to the network output. The network was trained by minimizing the mean squared error loss using the Adam optimizer. For LoDoPaB-CT, the network was trained for 20 epochs with a learning rate starting from 2×10^{-4}, reduced by cosine annealing to 1×10^{-5}, using batch size 2. For the Apple CT datasets, the network was trained for at most 80 epochs with a learning rate starting from 1×10^{-4}, reduced by cosine annealing to 1×10^{-5}, using batch size 1, whereby the model with the highest PSNR on the validation set was selected.

Source code is publicly available in a github repository (https://github.com/liutianlin0121/ISTA-U-Net, last accessed: 1 March 2021). A slightly modified copy of the code used for training on the Apple CT datasets is also contained in our github repository (https://github.com/jleuschn/learned_ct_reco_comparison_paper, last accessed: 1 March 2021).

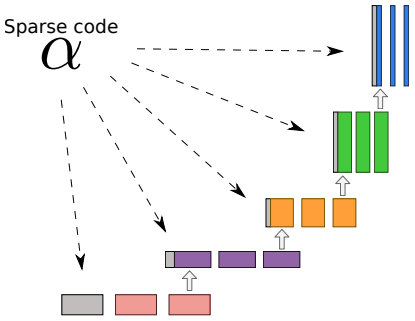

Figure A5. Architecture of the ISTA U-Net adapted from [42]. The sparse code α replaces the downsampling part in the standard U-Net (cf. Figure A2).

Algorithm A3 ISTA U-Net.

Given a noisy input y_δ, learned dictionaries $D_\kappa, D_\theta, D_\gamma$ and learned step sizes η and λ the reconstruction using the ISTA U-Net can be computed as follows.

1. Calculate FBP: $\hat{x} = \mathcal{T}_{\text{FBP}}(y_\delta)$
2. Initialize $\alpha_{y^\delta}^{[0]} = 0$
3. **for** $k = 1 : K$
4. $\quad \alpha_{y_\delta}^{[k]} = \text{ReLU}\left(\alpha_{y_\delta}^{[k-1]} + \eta D_\kappa^T \left(\hat{x} - D_\theta \alpha_{y_\delta}^{[i-1]}\right) - \eta\lambda\right)$
5. **end**
6. **return** $\hat{x} = D_\gamma \alpha_{y_\delta}^{[K]}$

Appendix A.7. Deep Image Prior with TV Denoising

The deep image prior (DIP) [86] takes a special role among the listed neural network approaches. In general, a DIP network F is not previously trained and, therefore, omits the problem of ground truth acquisition. Instead, the parameters Θ are adjusted iteratively during the reconstruction process by gradient descent steps (cf. Algorithm A4). The main objective is to minimize the data discrepancy of the output of the network for a fixed random input z

$$\min_\Theta \mathcal{D}_Y(\mathcal{A}F_\Theta(z), y_\delta). \tag{A1}$$

The number of iterations have a great influence on the reconstruction quality: While too few can result in an overall bad image, too many can cause overfitting to the noise of the measurement. The general regularization strategy for this problem is a combination of early stopping and the architecture itself [87], where the prior is related to the implicit structural bias of the network. Especially convolutional networks, in combination with gradient descent, fit natural images faster than noise and learn to construct them from low to high frequencies [86,88,89].

The loss function (A1) can also be combined with classical regularization. Baguer et al. [34] add a weighted anisotropic total variation (TV) term and apply their approach to low-dose CT measurements. The method DIP + TV is also used for this comparison. The network architecture is based on the same U-Net as for the FBP U-Net post-processing (cf. Appendix A.2). It has 6 different scales with 128 channels each and a skip-channel setup of $(0, 0, 0, 0, 4, 4)$. The data discrepancy $\mathcal{D}_Y$ was measured with a Poisson loss (see Equation (9)) and the weight for TV was chosen as $\alpha = 7.0$. Gradient descent was performed for $K = 17,000$ iterations with a stepsize of 5×10^{-4}.

Algorithm A4 Deep Image Prior + Total Variation (DIP + TV).

Given a noisy measurement y_δ, a neural network F_Θ with initial parameterization $\Theta^{[0]}$, forward operator $\mathcal{A}$ and a fixed random input z. The reconstruction $\hat{x}$ is calculated iteratively over a number of $K \in \mathbb{N}$ iterations:

1. **for** $k = 1 : K$
2. $\quad$ Evaluate loss: $L = \mathcal{D}\left(\mathcal{A}F_{\Theta^{[k-1]}}(z), y_\delta\right) + \alpha\,\text{TV}\left(F_{\Theta^{[k-1]}}(z)\right)$
3. $\quad$ Calculate gradients: $\nabla_{\Theta^{[k-1]}} = \nabla_\Theta L$
4. $\quad$ Update parameters: $\Theta^{[k]} = \text{Optimizer}\left(\Theta^{[k-1]}, \nabla_{\Theta^{[k-1]}}\right)$
5. $\quad$ Current reconstruction: $\hat{x}^{[k]} = F_{\Theta^{[k]}}(z)$
6. **end**

Appendix A.8. iCTU-Net

The iCTU-Net is based on the iCT-Net by Li et al. [29], which in turn is inspired by the common filtered back-projection. The reconstruction process is learned end-to-end, that is, the sinogram is the input of the network and the output is the reconstructed image. The full network architecture is shown in Figure A6.

First, disturbances in the raw measurement data, such as excessive noise, are suppressed as much as possible via 3×3 convolutions (refining layers). The corrected sinogram is then filtered using 10×1 convolutions (filtering layers). The filtered sinogram maintains the size of the input sinogram. Afterwards, the sinogram is back-projected into the image space. This is realized by a $d \times 1$ convolution with N^2 output channels without padding, where d is the number of detectors in the sinogram and N is the output image size. This convolution corresponds to a fully connected layer for each viewing angle, as it connects every detector element with every pixel in the image space. The results for each view are reshaped to $N \times N$ sized images and rotated according to the acquisition angle. A 1×1 convolution combines all views into the back projected image. Finally, a U-Net further refines the image output.

To significantly lower the GPU memory requirements, an initial convolutional layer with stride 1×2 was added, to downsample the LoDoPaB sinograms from 1000 to 500 projection angles. For the apple reconstruction the number of detector elements d and the output image size N were halved. After reconstruction the image size was doubled again using linear interpolation. Training was performed using the Adam optimizer with a learning rate of 0.001 and batch size 1. For LoDoPaB-CT the mean squared error loss and for Apple CT the SSIM loss function was used. The network was trained for 2 epochs on LoDoPaB-CT and for at most 60 epochs on the Apple CT datasets, without validation based model selection (i.e., no automated early stopping).

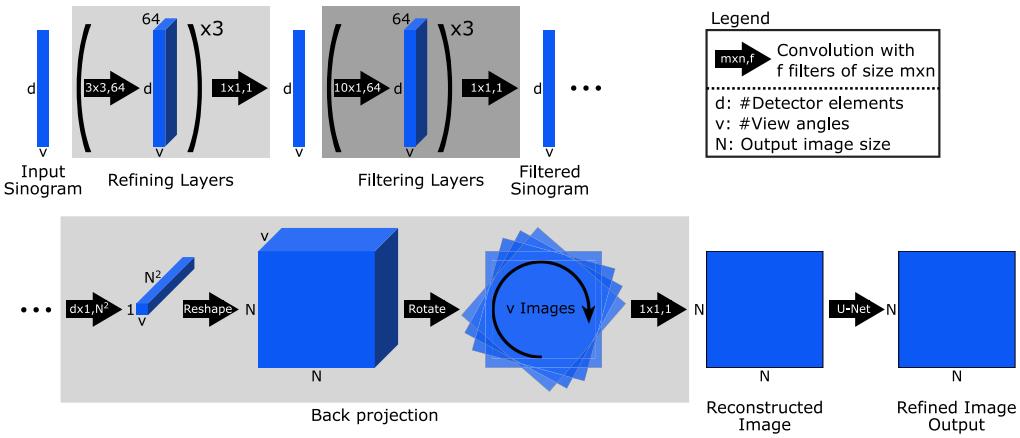

Figure A6. Architecture of the iCTU-Net.

Appendix B. Classical Reconstruction Methods

Appendix B.1. Filtered Back-Projection (FBP)

The Radon transform [10] maps (or projects) a function $x(u)$, $u = (u_1, u_2)$, defined on a two-dimensional plane to a function $\mathcal{A}x(s, \varphi)$ defined on a two-dimensional space of lines, which are parameterized by distance to the origin, s and the angle φ of the normal. The Radon transform is given by

$$\mathcal{A}x(s,\varphi) := \int_{\mathbb{R}} x\left(s\begin{bmatrix}\cos(\varphi)\\ \sin(\varphi)\end{bmatrix} + t\begin{bmatrix}-\sin(\varphi)\\ \cos(\varphi)\end{bmatrix}\right) \mathrm{d}t,$$

A simple inversion idea consists in back-projecting the intensities $\mathcal{A}x(s, \varphi)$ to those positions u in the image $x(u)$ that lie on the corresponding lines parameterized by s and φ, that is, those positions that contribute to the respective measured intensity. Mathematically,

the back-projection is described by the adjoint Radon transform $\mathcal{A}^*$, also provided in [10]. To obtain an inversion formula, the projections $\mathcal{A}x$ need to be filtered before the back-projection (see e.g., [36] for a derivation and an alternative formula applying a filter after obtaining the back-projection $\mathcal{A}^*\mathcal{A}x$). A generic FBP reconstruction formula reads

$$\hat{x} = \frac{1}{2}\mathcal{A}^*\mathcal{F}^{-1}|\cdot|W\mathcal{F}y_\delta,$$

where $\mathcal{F}$ denotes the one-dimensional Fourier transform along the detector pixel dimension s, $|\cdot|$ denotes the Ram-Lak filter, which multiplies each frequency component with the absolute value of the frequency, and W is a low-pass filter (applying a window function). While from perfect projections $\mathcal{A}x(s,\varphi)$ exact recovery of $x(u)$ is possible by choosing a rectangular window function for W, in practice W is also used to reduce high frequency components. This stabilizes the inversion by reducing the impact of noise present in higher frequencies. Typical choices for W are the Hann or the Cosine window. Sometimes the resulting weighting function is additionally shrunk along the frequency axis with a frequency scaling factor, which leads to removal of all frequency components above a threshold frequency.

For all experiments the implementation of ODL [90] was used in conjunction with the ASTRA toolbox [91]. Suitable hyperparameters have been determined based on the performance on validation samples and are listed in Table A1. The FBPs used for post-processing networks were computed with the Hann window and without frequency scaling. The Hann window thereby serves as a pre-processing step for the network and the frequency scaling was omitted in order to keep all information available.

Table A1. Hyperparameters for filtered back-projection (FBP).

		Window	**Frequency Scaling**
LoDoPaB-CT Dataset		Hann	0.641
Apple CT Dataset A (Noise-free)	50 angles	Cosine	0.11
	10 angles	Cosine	0.013
	5 angles	Hann	0.011
	2 angles	Hann	0.011
Apple CT Dataset B (Gaussian noise)	50 angles	Cosine	0.08
	10 angles	Cosine	0.013
	5 angles	Hann	0.011
	2 angles	Hann	0.011
Apple CT Dataset C (Scattering)	50 angles	Cosine	0.09
	10 angles	Hann	0.018
	5 angles	Hann	0.011
	2 angles	Hann	0.009

Appendix B.2. Conjugate Gradient Least Squares

The Conjugate Gradient Least Squares (CGLS) method is the modification of the well-known Conjugate Gradient [52] where the CG method is applied to solve the least squares problem $A^TA\hat{x} = A^Ty_\delta$. Here, $A \in \mathbb{R}^{m\times n}$ is the geometry matrix, $y_\delta \in \mathbb{R}^{m\times 1}$ is the measured data and $\hat{x} \in \mathbb{R}^{n\times 1}$ is the reconstruction. CGLS is a popular method in signal and image processing for its simple and computationally inexpensive implementation and fast convergence. The method is given in Algorithm A5, codes from [92].

Our implementation also includes a non-negativity step (negative pixel values equal to zero), applied to the final iterated solution. There is no parameter-tuning done for this implementation since the only user-defined parameter is the maximum number of iterations, K.

Algorithm A5 Conjugate Gradient Least Squares (CGLS).

Given a geometry matrix, A, a data vector y_δ and a zero solution vector $\hat{x}^{[0]} = 0$ (a black image) as the starting point, the algorithm below gives the solution at k^{th} iteration.

1. Initialise the direction vector as $d^{[0]} = A^T y_\delta$.
2. **for** $k = 1 : K$
3. $\quad q^{[k-1]} = Ad^{[k-1]}$, $\alpha = \|d^{[k-1]}\|_2^2 / \|q^{[k-1]}\|_2^2$
4. $\quad$ Update: $\hat{x}^{[k]} = \hat{x}^{[k-1]} + \alpha d^{[k-1]}$, $b^{[k]} = b^{[k-1]} - \alpha q^{[k-1]}$
5. $\quad$ Reinitialise: $q^{[k]} = A^T q^{[k-1]}$, $\beta = \|q^{[k]}\|_2^2 / \|d^{[k-1]}\|_2^2$, $d^{[k]} = q^{[k]} + \beta d^{[k-1]}$
6. **end**

Appendix B.3. Total Variation Regularization

Regularizing the reconstruction process with anisotropic total variation (TV) is a common approach for CT [93]. In addition to the data discrepancy $\mathcal{D}$, a weighted regularization term is added to the minimization problem

$$\mathcal{T}_{\mathrm{TV}}(y_\delta) \in \arg\min_x \mathcal{D}(\mathcal{A}x, y_\delta) + \alpha(\|\nabla_h x\|_1 + \|\nabla_v x\|_1), \tag{A2}$$

where ∇_h and ∇_v denote gradients in horizontal and vertical image direction, respectively, and can be approximated by finite differences in the discrete setting. TV penalizes variations in the image, e.g., from noise. Therefore, it is often applied in a denoising role. A number of optimization algorithms exist for minimizing (A2) [54]. The choice and exact formulation depend on the properties of the data discrepancy term.

For our comparison, we use the standard DIV$\alpha\ell$ implementation of TV. Adam gradient descent minimizes (A2), whereby the gradients are calculated by automatic differentiation in PyTorch [94] (cf. Algorithm A6).

Algorithm A6 Total Variation Regularization (TV).

Given a noisy measurement y_δ, an initial reconstruction $\hat{x}^{[0]}$, a weight $\alpha > 0$ and a maximum number of iterations K.

1. **for** $k = 1 : K$
2. $\quad$ Evaluate loss: $L = \mathcal{D}\left(\mathcal{A}\hat{x}^{[k-1]}, y_\delta\right) + \alpha\left(\left\|\nabla_h \hat{x}^{[k-1]}\right\|_1 + \left\|\nabla_v \hat{x}^{[k-1]}\right\|_1\right)$
3. $\quad$ Calculate gradients: $\nabla_{\hat{x}^{[k-1]}} = \nabla_x L$
4. $\quad$ Update: $\hat{x}^{[k]} = \mathrm{Optimizer}\left(\hat{x}^{[k-1]}, \nabla_{\hat{x}^{[k-1]}}\right)$
5. **end**

For the data discrepancy $\mathcal{D}$, a Poisson loss (see (9)) was used for LoDoPaB-CT, while the MSE was used for the Apple CT datasets. Suitable hyperparameters have been determined based on the performance on validation samples and are listed in Table A2. For lower numbers of angles, a very high number of iterations was found to be beneficial, leading to very slow reconstruction ($\approx$17 min per image for $K = 150{,}000$ iterations, which we chose to be the maximum). In all cases an FBP with Hann window and frequency scaling factor 0.1 was used as initial reconstruction.

Table A2. Hyperparameters for total variation regularization (TV).

		Discrepancy	Iterations	Step Size	α
LoDoPaB-CT Dataset		$-\ell_{\text{Pois}}$	5000	0.001	20.56
Apple CT Dataset A (Noise-free)	50 angles	MSE	600	3×10^{-2}	2×10^{-12}
	10 angles	MSE	75,000	3×10^{-3}	6×10^{-12}
	5 angles	MSE	146,000	1.5×10^{-3}	1×10^{-11}
	2 angles	MSE	150,000	1×10^{-3}	2×10^{-11}
Apple CT Dataset B (Gaussian noise)	50 angles	MSE	900	3×10^{-4}	2×10^{-10}
	10 angles	MSE	66,000	2×10^{-5}	6×10^{-10}
	5 angles	MSE	100,000	1×10^{-5}	3×10^{-9}
	2 angles	MSE	149,000	1×10^{-5}	4×10^{-9}
Apple CT Dataset C (Scattering)	50 angles	MSE	400	5×10^{-3}	1×10^{-11}
	10 angles	MSE	13,000	2×10^{-3}	4×10^{-11}
	5 angles	MSE	149,000	1×10^{-3}	4×10^{-11}
	2 angles	MSE	150,000	4×10^{-4}	6×10^{-11}

Appendix C. Further Results

Table A3. Standard deviation of PSNR and SSIM (adapted to the data range of each ground truth image) for the different noise settings on the 100 selected Apple CT test images.

Noise-Free	**Standard Deviation of PSNR**				**Standard Deviation of SSIM**			
Number of Angles	**50**	**10**	**5**	**2**	**50**	**10**	**5**	**2**
Learned Primal-Dual	1.51	1.63	1.97	2.58	0.022	0.016	0.014	0.022
ISTA U-Net	1.40	1.77	2.12	2.13	0.018	0.018	0.022	0.037
U-Net	1.56	1.61	2.28	1.63	0.021	0.019	0.025	0.031
MS-D-CNN	1.51	1.65	1.81	2.09	0.021	0.020	0.024	0.022
CINN	1.40	1.64	1.99	2.17	0.016	0.019	0.023	0.027
iCTU-Net	1.68	2.45	1.92	1.93	0.024	0.027	0.030	0.028
TV	1.60	1.29	1.21	1.49	0.022	0.041	0.029	0.023
CGLS	0.69	0.48	2.94	0.70	0.014	0.027	0.029	0.039
FBP	0.80	0.58	0.54	0.50	0.021	0.023	0.028	0.067
Gaussian Noise	**Standard Deviation of PSNR**				**Standard Deviation of SSIM**			
Number of Angles	**50**	**10**	**5**	**2**	**50**	**10**	**5**	**2**
Learned Primal-Dual	1.56	1.63	2.00	2.79	0.021	0.018	0.021	0.022
ISTA U-Net	1.70	1.76	2.27	2.12	0.025	0.021	0.022	0.038
U-Net	1.66	1.59	1.99	2.22	0.023	0.020	0.025	0.026
MS-D-CNN	1.66	1.75	1.79	1.79	0.025	0.024	0.019	0.022
CINN	1.53	1.51	1.62	2.06	0.023	0.017	0.017	0.020
iCTU-Net	1.98	2.06	1.89	1.91	0.031	0.032	0.039	0.027
TV	1.38	1.26	1.09	1.62	0.036	0.047	0.039	0.030
CGLS	0.78	0.49	1.76	0.68	0.014	0.026	0.029	0.037
FBP	0.91	0.58	0.54	0.50	0.028	0.023	0.028	0.067

Table A3. *Cont.*

Scattering Noise	**Standard Deviation of PSNR**				**Standard Deviation of SSIM**			
Number of Angles	**50**	**10**	**5**	**2**	**50**	**10**	**5**	**2**
Learned Primal-Dual	1.91	1.80	1.71	2.47	0.017	0.016	0.016	0.060
ISTA U-Net	1.48	1.59	2.05	1.81	0.023	0.019	0.019	0.038
U-Net	1.76	1.56	1.81	1.47	0.015	0.021	0.027	0.024
MS-D-CNN	2.04	1.78	1.85	2.03	0.023	0.022	0.015	0.020
CINN	1.82	1.92	2.32	2.25	0.019	0.024	0.029	0.030
iCTU-Net	1.91	2.09	1.78	2.29	0.030	0.031	0.033	0.040
TV	2.53	2.44	1.86	1.59	0.067	0.076	0.035	0.062
CGLS	2.38	1.32	1.71	0.95	0.020	0.020	0.026	0.032
FBP	2.23	0.97	0.80	0.68	0.044	0.025	0.023	0.058

Table A4. PSNR-FR and SSIM-FR (computed with fixed data range 0.0129353 for all images) for the different noise settings on the 100 selected Apple CT test images. Best results are highlighted in gray.

Noise-Free	**PSNR-FR**				**SSIM-FR**			
Number of Angles	**50**	**10**	**5**	**2**	**50**	**10**	**5**	**2**
Learned Primal-Dual	45.33	42.47	37.41	28.61	0.971	0.957	0.935	0.872
ISTA U-Net	45.48	41.15	34.93	27.10	0.967	0.944	0.907	0.823
U-Net	46.24	40.13	34.38	26.39	0.975	0.917	0.911	0.830
MS-D-CNN	46.47	41.00	35.06	27.17	0.975	0.936	0.898	0.808
CINN	46.20	41.46	34.43	26.07	0.975	0.958	0.896	0.838
iCTU-Net	42.69	36.57	32.24	25.90	0.957	0.938	0.920	0.861
TV	45.89	35.61	28.66	22.57	0.976	0.904	0.746	0.786
CGLS	39.66	28.43	19.22	21.87	0.901	0.744	0.654	0.733
FBP	37.01	23.71	22.12	20.58	0.856	0.711	0.596	0.538
Gaussian Noise	**PSNR-FR**				**SSIM-FR**			
Number of Angles	**50**	**10**	**5**	**2**	**50**	**10**	**5**	**2**
Learned Primal-Dual	43.24	40.38	36.54	28.03	0.961	0.944	0.927	0.823
ISTA U-Net	42.65	40.17	35.09	27.32	0.956	0.942	0.916	0.826
U-Net	43.09	39.45	34.42	26.47	0.961	0.924	0.904	0.843
MS-D-CNN	43.28	39.82	34.60	26.50	0.962	0.932	0.886	0.797
CINN	43.39	38.50	33.19	26.60	0.966	0.904	0.878	0.816
iCTU-Net	39.51	36.38	31.29	26.06	0.939	0.932	0.905	0.867
TV	38.98	33.73	28.45	22.70	0.939	0.883	0.770	0.772
CGLS	33.98	27.71	21.52	21.73	0.884	0.748	0.668	0.734
FBP	34.50	23.70	22.12	20.58	0.839	0.711	0.596	0.538
Scattering Noise	**PSNR-FR**				**SSIM-FR**			
Number of Angles	**50**	**10**	**5**	**2**	**50**	**10**	**5**	**2**
Learned Primal-Dual	44.42	40.80	33.69	27.60	0.967	0.954	0.912	0.760
ISTA U-Net	42.55	38.95	34.03	26.57	0.959	0.922	0.887	0.816
U-Net	41.58	39.52	33.55	25.56	0.932	0.910	0.877	0.828
MS-D-CNN	44.66	40.13	34.34	26.81	0.969	0.927	0.889	0.796
CINN	45.18	40.69	34.66	25.76	0.976	0.952	0.936	0.878
iCTU-Net	32.88	29.46	27.86	24.93	0.931	0.901	0.896	0.873
TV	27.71	26.76	24.48	21.15	0.903	0.799	0.674	0.743
CGLS	27.46	24.89	20.64	20.80	0.896	0.738	0.659	0.736
FBP	27.63	22.42	20.88	19.68	0.878	0.701	0.589	0.529

Table A5. Standard deviation of PSNR-FR and SSIM-FR (computed with fixed data range 0.0129353 for all images) for the different noise settings on the 100 selected Apple CT test images.

Noise-Free	**Standard Deviation of PSNR-FR**				**Standard Deviation of SSIM-FR**			
Number of Angles	**50**	**10**	**5**	**2**	**50**	**10**	**5**	**2**
Learned Primal-Dual	1.49	1.67	2.03	2.54	0.007	0.006	0.010	0.019
ISTA U-Net	1.37	1.82	2.21	2.21	0.005	0.010	0.020	0.034
U-Net	1.53	1.66	2.33	1.68	0.006	0.012	0.019	0.026
MS-D-CNN	1.46	1.71	1.90	2.15	0.006	0.011	0.021	0.015
CINN	1.35	1.65	2.09	2.21	0.004	0.007	0.023	0.025
iCTU-Net	1.82	2.54	2.03	1.91	0.014	0.017	0.020	0.023
TV	1.54	1.32	1.28	1.36	0.006	0.023	0.026	0.018
CGLS	0.71	0.51	2.96	0.56	0.009	0.029	0.033	0.045
FBP	0.77	0.46	0.38	0.41	0.011	0.015	0.029	0.088
Gaussian Noise	**Standard Deviation of PSNR-FR**				**Standard Deviation of SSIM-FR**			
Number of Angles	**50**	**10**	**5**	**2**	**50**	**10**	**5**	**2**
Learned Primal-Dual	1.52	1.68	2.04	2.83	0.006	0.008	0.013	0.016
ISTA U-Net	1.65	1.78	2.36	2.17	0.008	0.010	0.018	0.034
U-Net	1.61	1.62	2.05	2.24	0.007	0.012	0.019	0.024
MS-D-CNN	1.62	1.80	1.84	1.84	0.008	0.011	0.015	0.014
CINN	1.50	1.59	1.65	2.09	0.007	0.016	0.017	0.019
iCTU-Net	2.07	2.12	1.93	1.90	0.020	0.021	0.026	0.024
TV	1.30	1.26	1.15	1.50	0.014	0.027	0.030	0.019
CGLS	0.63	0.45	1.76	0.53	0.012	0.028	0.034	0.043
FBP	0.83	0.46	0.38	0.41	0.014	0.015	0.029	0.088
Scattering Noise	**Standard Deviation of PSNR-FR**				**Standard Deviation of SSIM-FR**			
Number of Angles	**50**	**10**	**5**	**2**	**50**	**10**	**5**	**2**
Learned Primal-Dual	1.92	1.85	1.81	2.51	0.005	0.007	0.014	0.038
ISTA U-Net	1.56	1.68	2.17	1.89	0.010	0.014	0.014	0.035
U-Net	1.72	1.63	1.91	1.59	0.010	0.012	0.024	0.024
MS-D-CNN	2.02	1.84	1.96	2.08	0.008	0.012	0.016	0.019
CINN	1.74	1.97	2.41	2.21	0.005	0.011	0.016	0.022
iCTU-Net	1.96	2.14	1.79	2.32	0.016	0.023	0.022	0.030
TV	2.43	2.35	1.80	1.49	0.048	0.074	0.040	0.051
CGLS	2.28	1.24	1.67	0.83	0.016	0.021	0.030	0.035
FBP	2.14	0.87	0.66	0.55	0.028	0.016	0.020	0.078

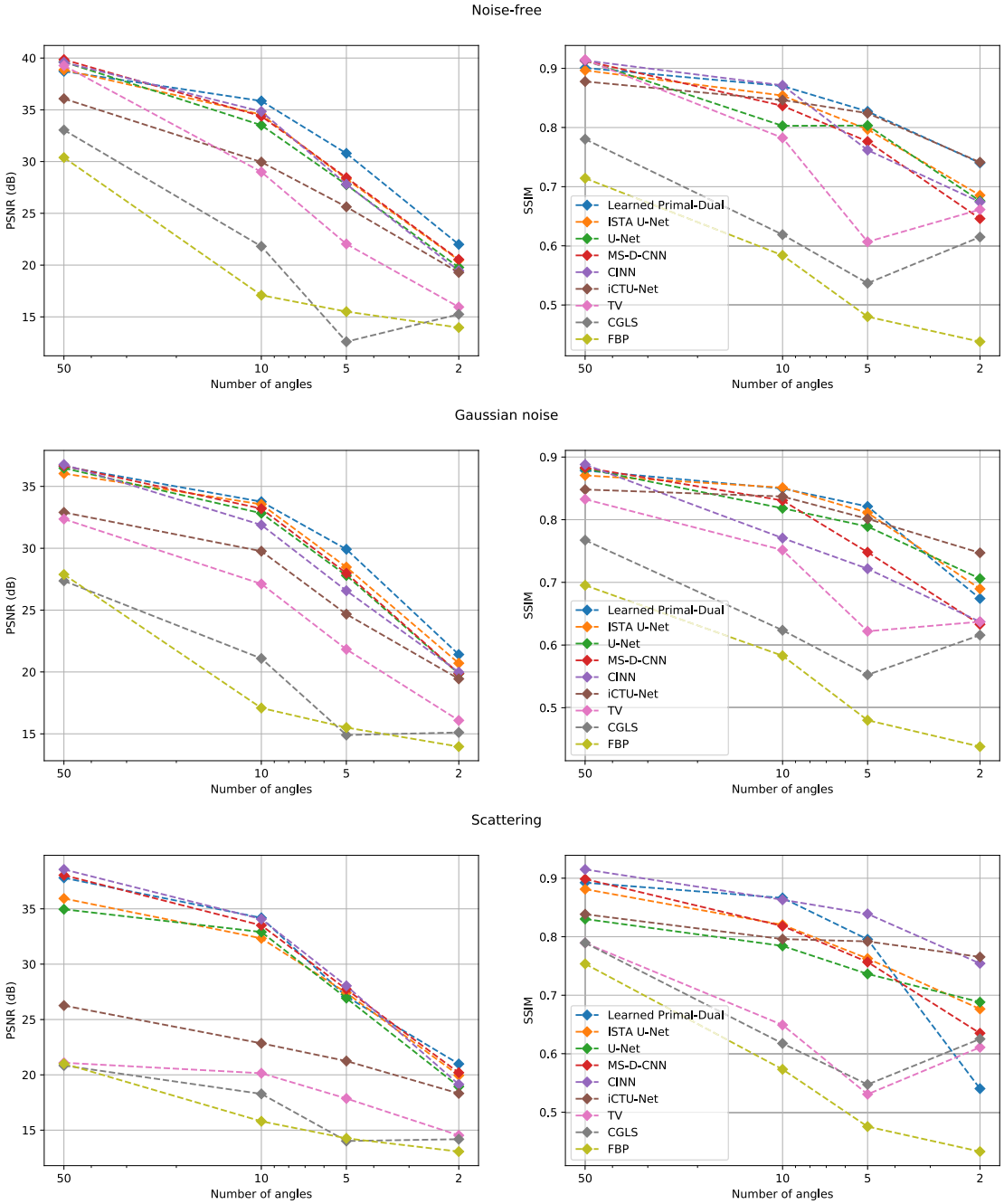

Figure A7. PSNR and SSIM depending on the number of angles on the Apple CT datasets.

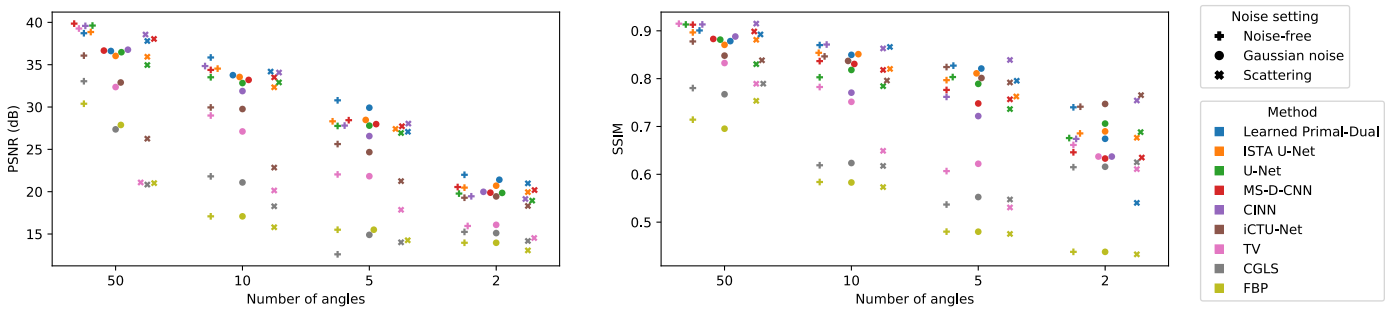

Figure A8. PSNR and SSIM compared for all noise settings and numbers of angles.

Table A6. Mean and standard deviation of the mean squared difference between the noisy measurements and the forward-projected reconstructions, respectively the noise-free measurements, on the 100 selected Apple CT test images.

Noise Free	**MSE $\times 10^9$**			
Number of Angles	**50**	**10**	**5**	**2**
Learned Primal-Dual	0.083 ± 0.027	0.405 ± 0.156	1.559 ± 0.543	2.044 ± 1.177
ISTA U-Net	0.323 ± 0.240	0.633 ± 0.339	2.672 ± 1.636	17.840 ± 12.125
U-Net	0.097 ± 0.093	1.518 ± 0.707	5.011 ± 3.218	31.885 ± 17.219
MS-D-CNN	0.117 ± 0.088	0.996 ± 0.595	3.874 ± 2.567	20.879 ± 12.038
CINN	0.237 ± 0.259	1.759 ± 0.348	3.798 ± 2.176	33.676 ± 16.747
iCTU-Net	2.599 ± 3.505	6.686 ± 8.469	14.508 ± 16.694	18.876 ± 12.553
TV	0.002 ± 0.000	0.001 ± 0.000	0.000 ± 0.000	0.001 ± 0.000
CGLS	1.449 ± 0.299	29.921 ± 6.173	752.997 ± 722.151	22.507 ± 13.748
FBP	12.229 ± 3.723	89.958 ± 9.295	159.746 ± 15.596	273.054 ± 114.552
Ground truth	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000	0.000 ± 0.000
Gaussian Noise	**MSE $\times 10^9$**			
Number of Angles	**50**	**10**	**5**	**2**
Learned Primal-Dual	19.488 ± 5.923	19.813 ± 5.851	20.582 ± 5.690	32.518 ± 4.286
ISTA U-Net	19.438 ± 5.943	20.178 ± 6.060	21.167 ± 6.052	32.435 ± 9.782
U-Net	19.802 ± 6.247	22.114 ± 6.364	23.645 ± 6.527	38.895 ± 17.211
MS-D-CNN	19.348 ± 5.921	20.056 ± 5.930	23.080 ± 5.959	47.625 ± 18.133
CINN	19.429 ± 5.891	21.069 ± 5.663	29.517 ± 7.296	42.876 ± 15.471
iCTU-Net	25.645 ± 9.602	25.421 ± 9.976	38.179 ± 22.887	41.956 ± 15.942
TV	18.760 ± 5.674	18.107 ± 5.395	20.837 ± 5.510	18.514 ± 5.688
CGLS	87.892 ± 23.312	71.526 ± 17.600	262.616 ± 151.655	98.520 ± 18.245
FBP	31.803 ± 9.558	109.430 ± 14.107	179.260 ± 19.744	292.692 ± 109.223
Ground truth	19.538 ± 6.029	19.505 ± 6.019	19.551 ± 6.028	19.483 ± 6.086
Scattering Noise	**MSE $\times 10^9$**			
Number of Angles	**50**	**10**	**5**	**2**
Learned Primal-Dual	541.30 ± 311.82	579.14 ± 317.59	549.30 ± 328.41	435.07 ± 260.02
ISTA U-Net	553.64 ± 355.14	557.03 ± 342.67	575.94 ± 338.82	522.33 ± 365.58
U-Net	629.62 ± 353.54	635.91 ± 343.31	550.54 ± 340.27	642.20 ± 295.46
MS-D-CNN	579.86 ± 332.39	585.18 ± 331.93	533.35 ± 331.21	606.55 ± 365.25
CINN	638.80 ± 355.24	619.47 ± 353.47	603.53 ± 362.96	649.30 ± 409.83
iCTU-Net	622.51 ± 348.32	622.63 ± 335.28	652.18 ± 359.00	573.46 ± 324.00
TV	3.35 ± 5.02	3.19 ± 4.83	2.96 ± 4.47	2.55 ± 6.33
CGLS	6.40 ± 6.39	34.71 ± 8.16	286.20 ± 205.42	19.92 ± 14.01
FBP	12.48 ± 6.88	73.53 ± 10.19	144.70 ± 15.82	221.79 ± 59.71
Ground truth	610.47 ± 355.25	610.40 ± 355.16	611.23 ± 354.51	620.11 ± 386.79

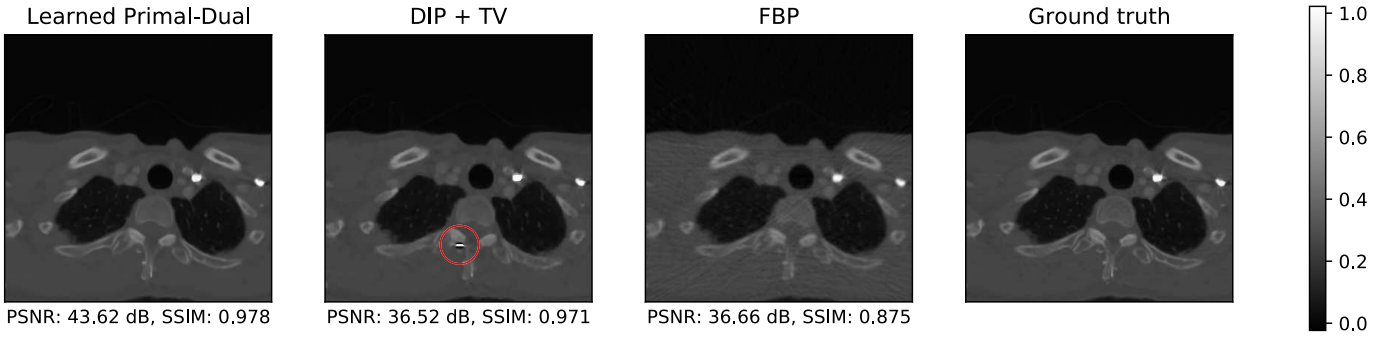

Figure A9. Example of an artifact produced by DIP + TV, which has only minor impact on the evaluated metrics (especially the SSIM). The area containing the artifact is marked with a red circle.

Appendix D. Training Curves

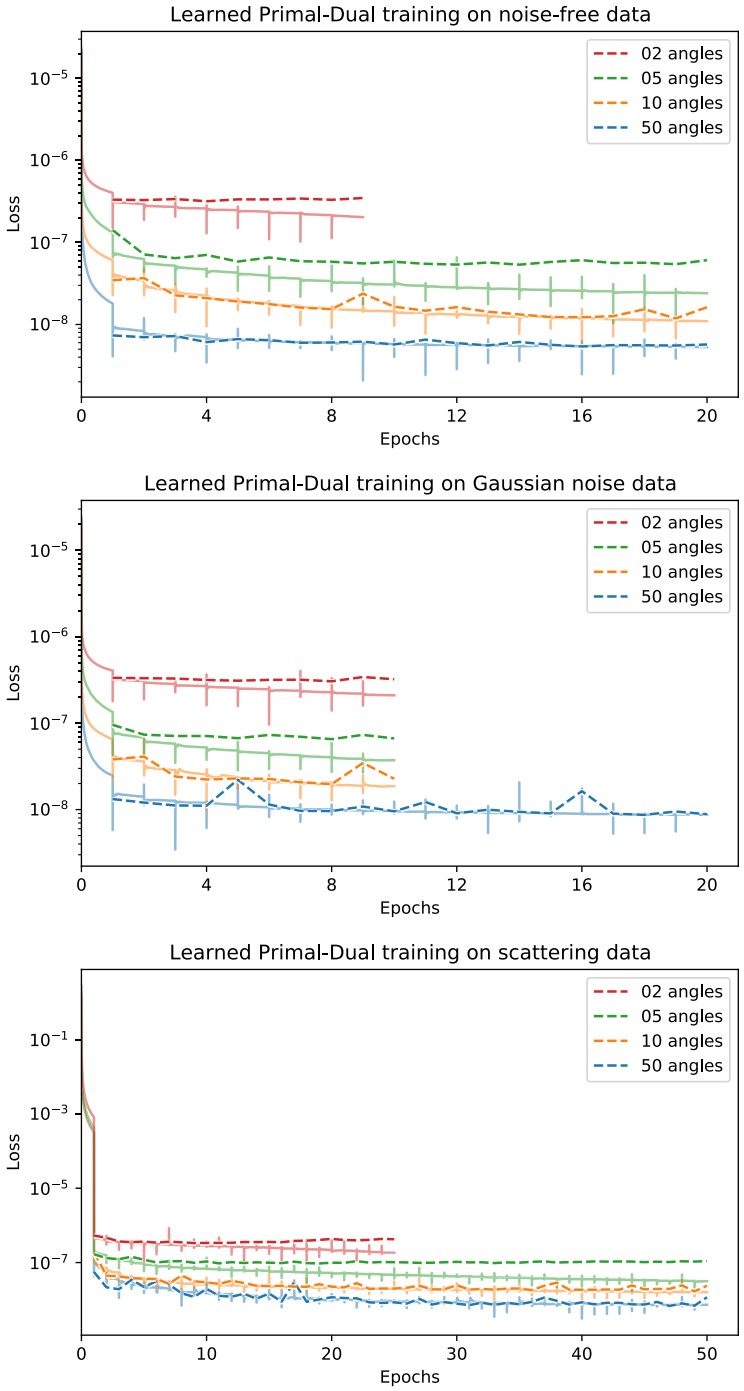

Figure A10. Training curves of Learned Primal-Dual on the Apple CT dataset. Dashed lines: average validation loss computed after every full training epoch; solid lines: running average of training loss since start of epoch. Duration of 20 epochs on full dataset: ≈10–17 days, varying with the number of angles.

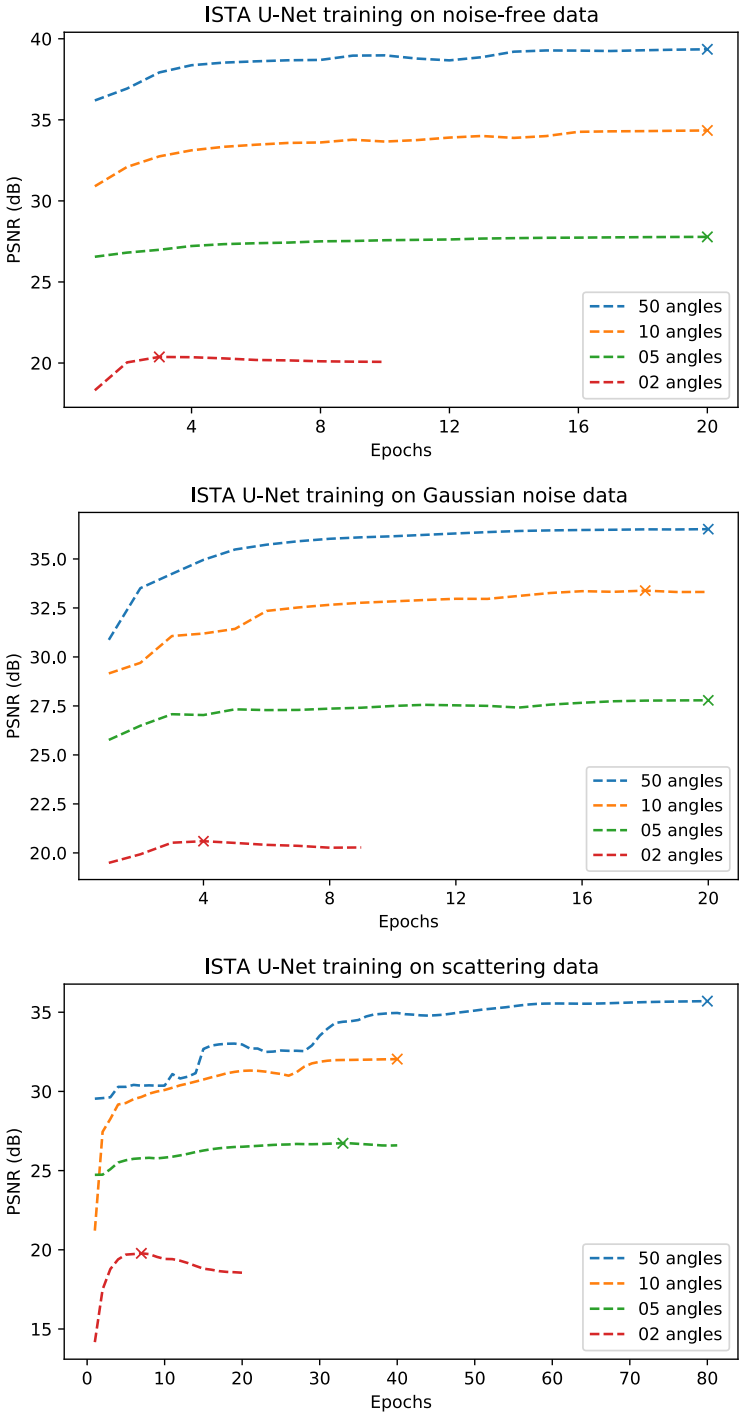

Figure A11. Training curves of ISTA U-Net on the Apple CT dataset. Dashed lines: average validation PSNR in decibel computed after every full training epoch; marks: selected model. Duration of 20 epochs on full dataset: ≈10 days for hidden layer width 32+ and 5 scales, respectively ≈5.5 days for hidden layer width 16+ and 6 scales.

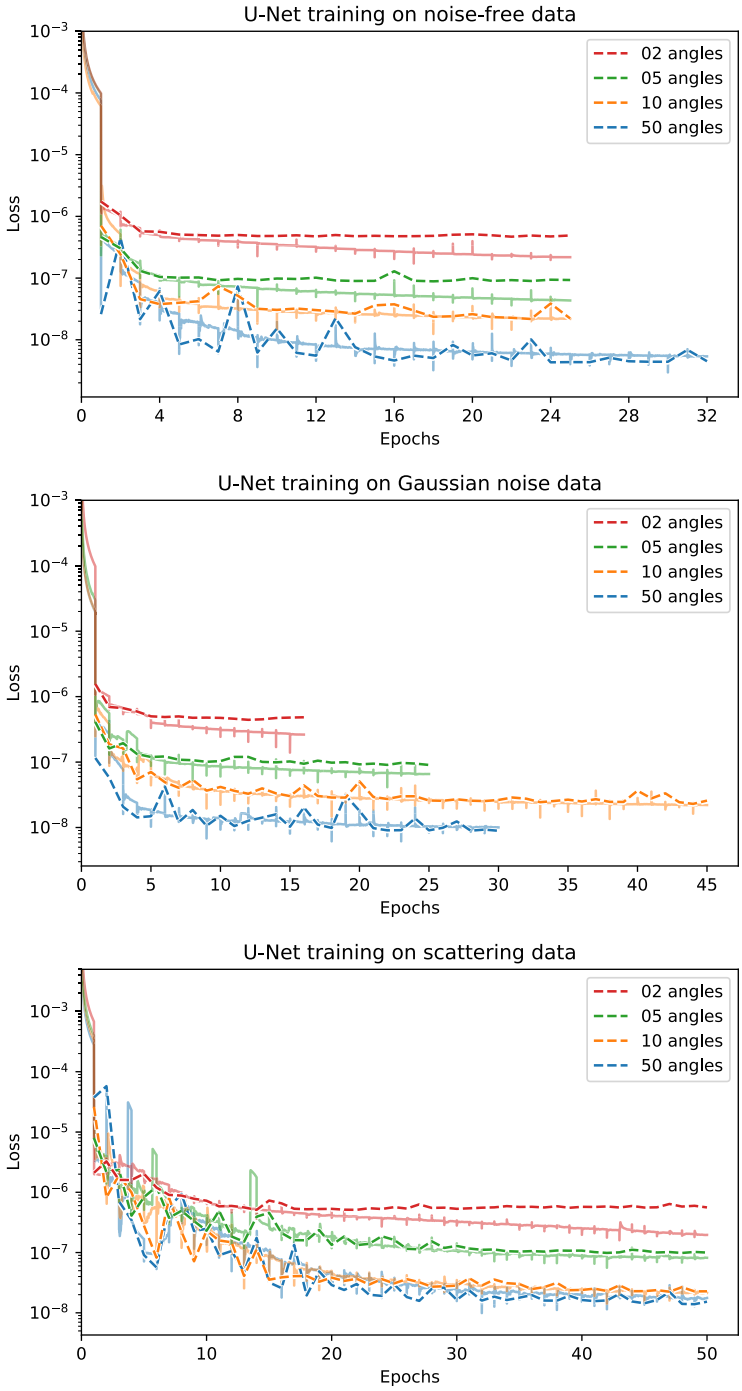

Figure A12. Training curves of U-Net on the Apple CT dataset. Dashed lines: average validation loss computed after every full training epoch; solid lines: running average of training loss since start of epoch. Duration of 20 epochs on full dataset: ≈1.5 days.

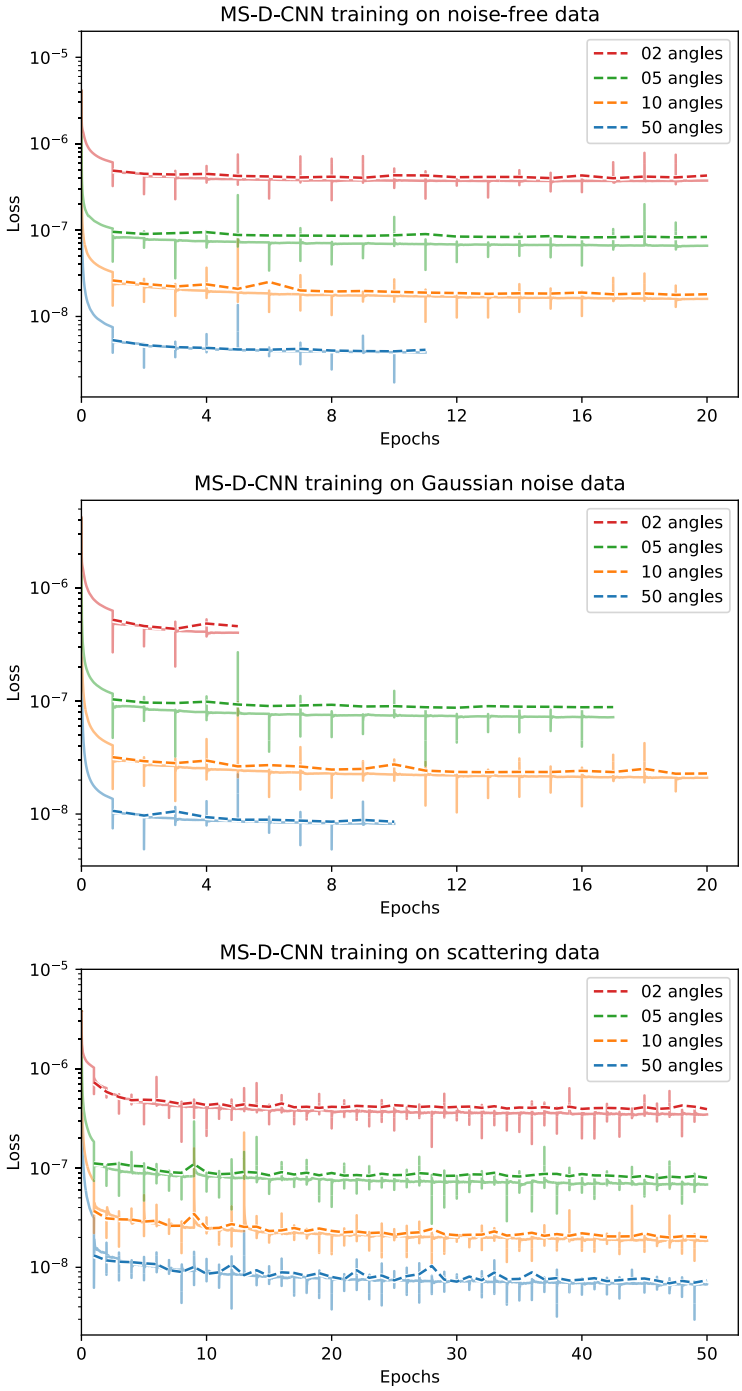

Figure A13. Training curves of MS-D-CNN on the Apple CT dataset. Dashed lines: average validation loss computed after every full training epoch; solid lines: running average of training loss since start of epoch. Duration of 20 epochs on full dataset: ≈20 days.

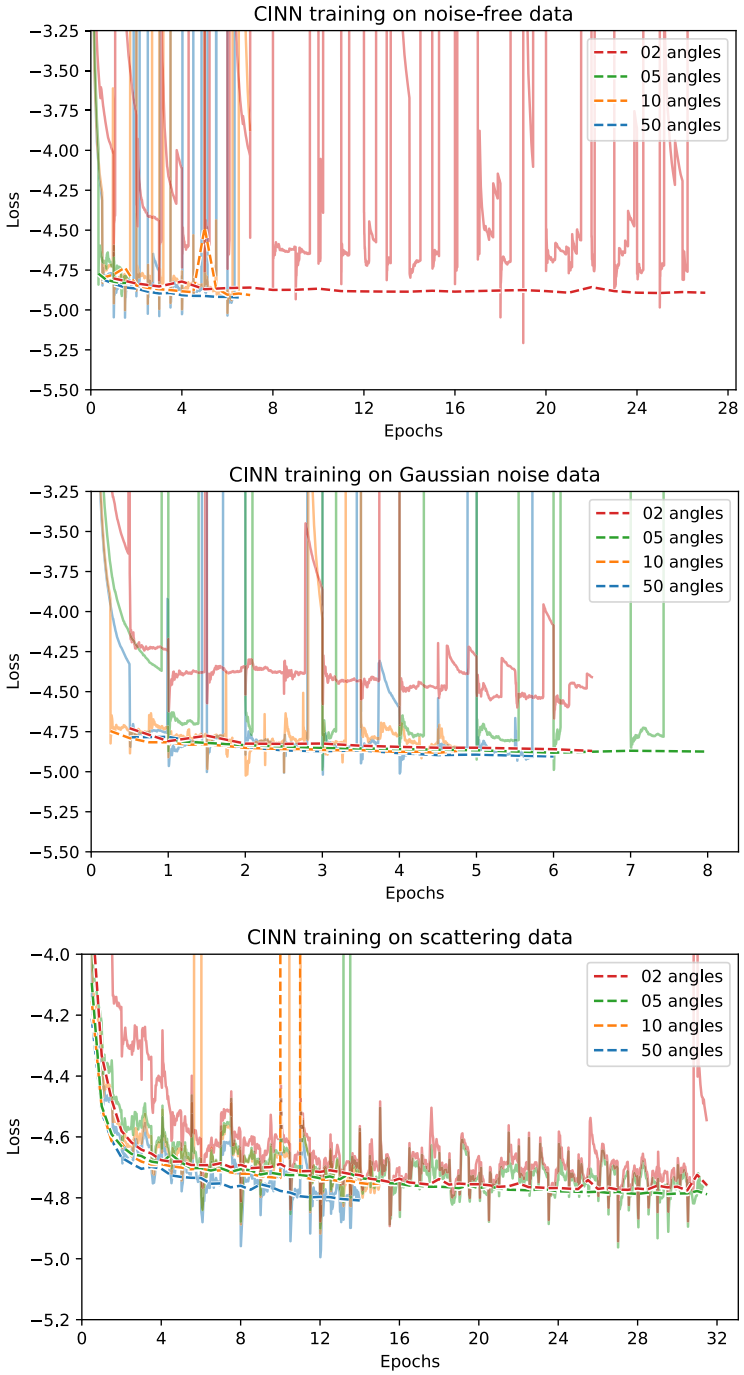

Figure A14. Training curves of CINN on the Apple CT dataset. Dashed lines: average validation loss computed after every full training epoch; solid lines: running average of training loss (at every 50-th step) since start of epoch. For some of the trainings, the epochs were divided into multiple shorter ones. Duration of 20 epochs on full dataset: ≈2.5 days (using 2 GPUs).

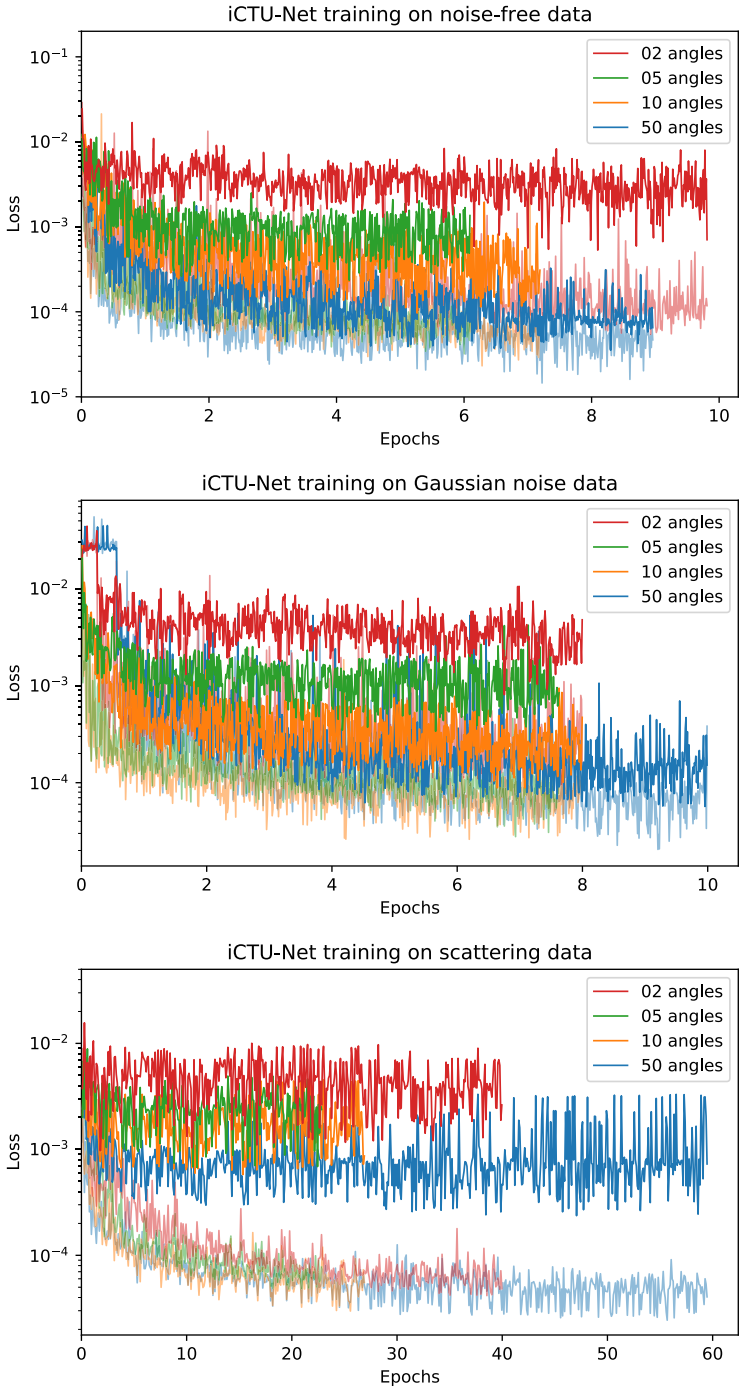

Figure A15. Training curves of iCTU-Net on the Apple CT dataset. Opaque lines: loss for a validation sample (after every 500-th step); semi-transparent lines: training loss (at every 500-th step). Duration of 20 epochs on full dataset: ≈3 days.

References

1. Liguori, C.; Frauenfelder, G.; Massaroni, C.; Saccomandi, P.; Giurazza, F.; Pitocco, F.; Marano, R.; Schena, E. Emerging clinical applications of computed tomography. *Med. Devices* **2015**, *8*, 265.
2. National Lung Screening Trial Research Team. Reduced lung-cancer mortality with low-dose computed tomographic screening. *N. Engl. J. Med.* **2011**, *365*, 395–409. [CrossRef]
3. Yoo, S.; Yin, F.F. Dosimetric feasibility of cone-beam CT-based treatment planning compared to CT-based treatment planning. *Int. J. Radiat. Oncol. Biol. Phys.* **2006**, *66*, 1553–1561. [CrossRef]
4. Swennen, G.R.; Mollemans, W.; Schutyser, F. Three-dimensional treatment planning of orthognathic surgery in the era of virtual imaging. *J. Oral Maxillofac. Surg.* **2009**, *67*, 2080–2092. [CrossRef]

5. De Chiffre, L.; Carmignato, S.; Kruth, J.P.; Schmitt, R.; Weckenmann, A. Industrial applications of computed tomography. *CIRP Ann.* **2014**, *63*, 655–677. [CrossRef]
6. Mees, F.; Swennen, R.; Van Geet, M.; Jacobs, P. *Applications of X-ray Computed Tomography in the Geosciences*; Special Publications; Geological Society: London, UK, 2003; Volume 215, pp. 1–6.
7. Morigi, M.; Casali, F.; Bettuzzi, M.; Brancaccio, R.; d'Errico, V. Application of X-ray computed tomography to cultural heritage diagnostics. *Appl. Phys. A* **2010**, *100*, 653–661. [CrossRef]
8. Coban, S.B.; Lucka, F.; Palenstijn, W.J.; Van Loo, D.; Batenburg, K.J. Explorative Imaging and Its Implementation at the FleX-ray Laboratory. *J. Imaging* **2020**, *6*, 18. [CrossRef]
9. McCollough, C.H.; Bartley, A.C.; Carter, R.E.; Chen, B.; Drees, T.A.; Edwards, P.; Holmes, D.R., III; Huang, A.E.; Khan, F.; Leng, S.; et al. Low-dose CT for the detection and classification of metastatic liver lesions: Results of the 2016 Low Dose CT Grand Challenge. *Med Phys.* **2017**, *44*, e339–e352. [CrossRef] [PubMed]
10. Radon, J. On the determination of functions from their integral values along certain manifolds. *IEEE Trans. Med Imaging* **1986**, *5*, 170–176. [CrossRef]
11. Natterer, F. The mathematics of computerized tomography (classics in applied mathematics, vol. 32). *Inverse Probl.* **2001**, *18*, 283–284.
12. Boas, F.E.; Fleischmann, D. CT artifacts: Causes and reduction techniques. *Imaging Med.* **2012**, *4*, 229–240. [CrossRef]
13. Wang, G.; Ye, J.C.; Mueller, K.; Fessler, J.A. Image Reconstruction is a New Frontier of Machine Learning. *IEEE Trans. Med. Imaging* **2018**, *37*, 1289–1296. [CrossRef] [PubMed]
14. Sidky, E.Y.; Pan, X. Image reconstruction in circular cone-beam computed tomography by constrained, total-variation minimization. *Phys. Med. Biol.* **2008**, *53*, 4777. [CrossRef]
15. Niu, S.; Gao, Y.; Bian, Z.; Huang, J.; Chen, W.; Yu, G.; Liang, Z.; Ma, J. Sparse-view X-ray CT reconstruction via total generalized variation regularization. *Phys. Med. Biol.* **2014**, *59*, 2997. [CrossRef] [PubMed]
16. Hestenes, M.R.; Stiefel, E. Methods of conjugate gradients for solving linear systems. *J. Res. Natl. Bur. Stand.* **1952**, *49*, 409–436. [CrossRef]
17. Arridge, S.; Maass, P.; Öktem, O.; Schönlieb, C.B. Solving inverse problems using data-driven models. *Acta Numer.* **2019**, *28*, 1–174. [CrossRef]
18. Lunz, S.; Öktem, O.; Schönlieb, C.B. Adversarial Regularizers in Inverse Problems. In *Advances in Neural Information Processing Systems*; Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2018; Volume 31, pp. 8507–8516.
19. Adler, J.; Öktem, O. Learned Primal-Dual Reconstruction. *IEEE Trans. Med. Imaging* **2018**, *37*, 1322–1332. TMI.2018.2799231. [CrossRef]
20. Jin, K.H.; McCann, M.T.; Froustey, E.; Unser, M. Deep Convolutional Neural Network for Inverse Problems in Imaging. *IEEE Trans. Image Process.* **2017**, *26*, 4509–4522. [CrossRef]
21. Pelt, D.M.; Batenburg, K.J.; Sethian, J.A. Improving Tomographic Reconstruction from Limited Data Using Mixed-Scale Dense Convolutional Neural Networks. *J. Imaging* **2018**, *4*, 128. [CrossRef]
22. Chen, H.; Zhang, Y.; Zhang, W.; Liao, P.; Li, K.; Zhou, J.; Wang, G. Low-dose CT via convolutional neural network. *Biomed. Opt. Express* **2017**, *8*, 679–694. [CrossRef]
23. Chen, H.; Zhang, Y.; Kalra, M.K.; Lin, F.; Chen, Y.; Liao, P.; Zhou, J.; Wang, G. Low-dose CT with a residual encoder-decoder convolutional neural network. *IEEE Trans. Med. Imaging* **2017**, *36*, 2524–2535. [CrossRef]
24. Yang, Q.; Yan, P.; Kalra, M.K.; Wang, G. CT image denoising with perceptive deep neural networks. *arXiv* **2017**, arXiv:1702.07019.
25. Yang, Q.; Yan, P.; Zhang, Y.; Yu, H.; Shi, Y.; Mou, X.; Kalra, M.K.; Zhang, Y.; Sun, L.; Wang, G. Low-dose CT image denoising using a generative adversarial network with Wasserstein distance and perceptual loss. *IEEE Trans. Med. Imaging* **2018**, *37*, 1348–1357. [CrossRef]
26. Feng, R.; Rundle, D.; Wang, G. Neural-networks-based Photon-Counting Data Correction: Pulse Pileup Effect. *arXiv* **2018**, arXiv:1804.10980.
27. Zhu, B.; Liu, J.Z.; Cauley, S.F.; Rosen, B.R.; Rosen, M.S. Image reconstruction by domain-transform manifold learning. *Nature* **2018**, *555*, 487–492. [CrossRef]
28. He, J.; Ma, J. Radon inversion via deep learning. *IEEE Trans. Med. Imaging* **2020**, *39*, 2076–2087. [CrossRef] [PubMed]
29. Li, Y.; Li, K.; Zhang, C.; Montoya, J.; Chen, G.H. Learning to reconstruct computed tomography images directly from sinogram data under a variety of data acquisition conditions. *IEEE Trans. Med Imaging* **2019**, *38*, 2469–2481. [CrossRef]
30. European Society of Radiology (ESR). The new EU General Data Protection Regulation: What the radiologist should know. *Insights Imaging* **2017**, *8*, 295–299. [CrossRef] [PubMed]
31. Kaissis, G.A.; Makowski, M.R.; Rückert, D.; Braren, R.F. Secure, privacy-preserving and federated machine learning in medical imaging. *Nat. Mach. Intell.* **2020**, *2*, 305–311. [CrossRef]
32. Leuschner, J.; Schmidt, M.; Baguer, D.O.; Maass, P. The LoDoPaB-CT Dataset: A Benchmark Dataset for Low-Dose CT Reconstruction Methods. *arXiv* **2020**, arXiv:1910.01113.
33. Armato, S.G., III; McLennan, G.; Bidaut, L.; McNitt-Gray, M.F.; Meyer, C.R.; Reeves, A.P.; Zhao, B.; Aberle, D.R.; Henschke, C.I.; Hoffman, E.A.; et al. The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): A Completed Reference Database of Lung Nodules on CT Scans. *Med. Phys.* **2011**, *38*, 915–931. [CrossRef]

34. Baguer, D.O.; Leuschner, J.; Schmidt, M. Computed tomography reconstruction using deep image prior and learned reconstruction methods. *Inverse Probl.* **2020**, *36*, 094004. [CrossRef]
35. Armato, S.G., III; McLennan, G.; Bidaut, L.; McNitt-Gray, M.F.; Meyer, C.R.; Reeves, A.P.; Zhao, B.; Aberle, D.R.; Henschke, C.I.; Hoffman, E.A.; et al. *Data From LIDC-IDRI*; The Cancer Imaging Archive: Frederick, MD, USA, 2015.10.7937/K9/TCIA.2015.LO9QL9SX. [CrossRef]
36. Buzug, T. *Computed Tomography: From Photon Statistics to Modern Cone-Beam CT*; Springer: Berlin/Heidelberg, Germany, 2008. [CrossRef]
37. Coban, S.B.; Andriiashen, V.; Ganguly, P.S. *Apple CT Data: Simulated Parallel-Beam Tomographic Datasets*; Zenodo: Geneva, Switzerland, 2020. [CrossRef]
38. Coban, S.B.; Andriiashen, V.; Ganguly, P.S.; van Eijnatten, M.; Batenburg, K.J. Parallel-beam X-ray CT datasets of apples with internal defects and label balancing for machine learning. *arXiv* **2020**, arXiv:2012.13346.
39. Leuschner, J.; Schmidt, M.; Ganguly, P.S.; Andriiashen, V.; Coban, S.B.; Denker, A.; van Eijnatten, M. Source Code and Supplementary Material for "Quantitative comparison of deep learning-based image reconstruction methods for low-dose and sparse-angle CT applications". *Zenodo* **2021**. [CrossRef]
40. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*; Springer: Heidelberg, Germany, 2015; pp. 234–241.
41. Zhou, Z.; Siddiquee, M.M.R.; Tajbakhsh, N.; Liang, J. Unet++: A nested u-net architecture for medical image segmentation. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*; Springer: Heidelberg, Germany, 2018; pp. 3–11.
42. Liu, T.; Chaman, A.; Belius, D.; Dokmanić, I. Interpreting U-Nets via Task-Driven Multiscale Dictionary Learning. *arXiv* **2020**, arXiv:2011.12815.
43. Comelli, A.; Dahiya, N.; Stefano, A.; Benfante, V.; Gentile, G.; Agnese, V.; Raffa, G.M.; Pilato, M.; Yezzi, A.; Petrucci, G.; et al. Deep learning approach for the segmentation of aneurysmal ascending aorta. *Biomed. Eng. Lett.* **2020**, 1–10.
44. Dashti, M.; Stuart, A.M. The Bayesian Approach to Inverse Problems. In *Handbook of Uncertainty Quantification*; Springer International Publishing: Cham, Switzerland, 2017; pp. 311–428. [CrossRef]
45. Adler, J.; Öktem, O. Deep Bayesian Inversion. *arXiv* **2018**, arXiv:1811.05910.
46. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. *arXiv* **2014**, arXiv:1406.2661.
47. Ardizzone, L.; Lüth, C.; Kruse, J.; Rother, C.; Köthe, U. Guided image generation with conditional invertible neural networks. *arXiv* **2019**, arXiv:1907.02392.
48. Denker, A.; Schmidt, M.; Leuschner, J.; Maass, P.; Behrmann, J. Conditional Normalizing Flows for Low-Dose Computed Tomography Image Reconstruction. *arXiv* **2020**, arXiv:2006.06270.
49. Hadamard, J. *Lectures on Cauchy's Problem in Linear Partial Differential Equations*; Dover: New York, NY, USA, 1952.
50. Nashed, M. A new approach to classification and regularization of ill-posed operator equations. In *Inverse and Ill-Posed Problems*; Engl, H.W., Groetsch, C., Eds.; Academic Press: Cambridge, MA, USA, 1987; pp. 53–75. [CrossRef]
51. Natterer, F.; Wübbeling, F. *Mathematical Methods in Image Reconstruction*; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2001.
52. Saad, Y. *Iterative Methods for Sparse Linear Systems*, 2nd ed.; Society for Industrial and Applied Mathematics: Philadelphia, PA, USA, 2003.
53. Björck, Å.; Elfving, T.; Strakos, Z. Stability of conjugate gradient and Lanczos methods for linear least squares problems. *SIAM J. Matrix Anal. Appl.* **1998**, *19*, 720–736. [CrossRef]
54. Chen, H.; Wang, C.; Song, Y.; Li, Z. Split Bregmanized anisotropic total variation model for image deblurring. *J. Vis. Commun. Image Represent.* **2015**, *31*, 282–293. [CrossRef]
55. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef] [PubMed]
56. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016. Available online: http://www.deeplearningbook.org (accessed on 1 March 2021).
57. Leuschner, J.; Schmidt, M.; Ganguly, P.S.; Andriiashen, V.; Coban, S.B.; Denker, A.; van Eijnatten, M. Supplementary Material for Experiments in "Quantitative comparison of deep learning-based image reconstruction methods for low-dose and sparse-angle CT applications". *Zenodo* **2021**. [CrossRef]
58. Leuschner, J.; Schmidt, M.; Baguer, D.O.; Bauer, D.; Denker, A.; Hadjifaradji, A.; Liu, T. LoDoPaB-CT Challenge Reconstructions compared in "Quantitative comparison of deep learning-based image reconstruction methods for low-dose and sparse-angle CT applications". *Zenodo* **2021**. [CrossRef]
59. Leuschner, J.; Schmidt, M.; Ganguly, P.S.; Andriiashen, V.; Coban, S.B.; Denker, A.; van Eijnatten, M. Apple CT Test Reconstructions compared in "Quantitative comparison of deep learning-based image reconstruction methods for low-dose and sparse-angle CT applications". *Zenodo* **2021**. [CrossRef]
60. Leuschner, J.; Schmidt, M.; Otero Baguer, D.; Erzmann, D.; Baltazar, M. DIVal Library. *Zenodo* **2021**. [CrossRef]

61. Knoll, F.; Murrell, T.; Sriram, A.; Yakubova, N.; Zbontar, J.; Rabbat, M.; Defazio, A.; Muckley, M.J.; Sodickson, D.K.; Zitnick, C.L.; et al. Advancing machine learning for MR image reconstruction with an open competition: Overview of the 2019 fastMRI challenge. *Magn. Reson. Med.* **2020**, *84*, 3054–3070. [CrossRef]
62. Putzky, P.; Welling, M. Invert to Learn to Invert. In *Advances in Neural Information Processing Systems*; Wallach, H., Larochelle, H., Beygelzimer, A., dAlch'e-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2019; Volume 32, pp. 446–456.
63. Etmann, C.; Ke, R.; Schönlieb, C. iUNets: Learnable Invertible Up- and Downsampling for Large-Scale Inverse Problems. In Proceedings of the 30th IEEE International Workshop on Machine Learning for Signal Processing (MLSP 2020), Espoo, Finland, 21–24 September 2020; pp. 1–6. [CrossRef]
64. Ziabari, A.; Ye, D.H.; Srivastava, S.; Sauer, K.D.; Thibault, J.; Bouman, C.A. 2.5D Deep Learning For CT Image Reconstruction Using A Multi-GPU Implementation. In Proceedings of the 2018 52nd Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, 28–31 October 2018; pp. 2044–2049. [CrossRef]
65. Scherzer, O.; Weickert, J. Relations Between Regularization and Diffusion Filtering. *J. Math. Imaging Vis.* **2000**, *12*, 43–63. [CrossRef]
66. Perona, P.; Malik, J. Scale-space and edge detection using anisotropic diffusion. *IEEE Trans. Pattern Anal. Mach. Intell.* **1990**, *12*, 629–639. [CrossRef]
67. Mendrik, A.M.; Vonken, E.; Rutten, A.; Viergever, M.A.; van Ginneken, B. Noise Reduction in Computed Tomography Scans Using 3-D Anisotropic Hybrid Diffusion With Continuous Switch. *IEEE Trans. Med Imaging* **2009**, *28*, 1585–1594. [CrossRef] [PubMed]
68. Adler, J.; Lunz, S.; Verdier, O.; Schönlieb, C.B.; Öktem, O. Task adapted reconstruction for inverse problems. *arXiv* **2018**, arXiv:1809.00948.
69. Boink, Y.E.; Manohar, S.; Brune, C. A partially-learned algorithm for joint photo-acoustic reconstruction and segmentation. *IEEE Trans. Med. Imaging* **2019**, *39*, 129–139. [CrossRef]
70. Handels, H.; Deserno, T.M.; Maier, A.; Maier-Hein, K.H.; Palm, C.; Tolxdorff, T. (Eds.) *Bildverarbeitung für die Medizin 2019*; Springer Fachmedien Wiesbaden: Wiesbaden, Germany, 2019. [CrossRef]
71. Mason, A.; Rioux, J.; Clarke, S.E.; Costa, A.; Schmidt, M.; Keough, V.; Huynh, T.; Beyea, S. Comparison of objective image quality metrics to expert radiologists' scoring of diagnostic quality of MR images. *IEEE Trans. Med. Imaging* **2019**, *39*, 1064–1072. [CrossRef] [PubMed]
72. Coban, S.B.; Lionheart, W.R.B.; Withers, P.J. Assessing the efficacy of tomographic reconstruction methods through physical quantification techniques. *Meas. Sci. Technol.* **2021**. [CrossRef]
73. Goodfellow, I.J.; Shlens, J.; Szegedy, C. Explaining and Harnessing Adversarial Examples. In Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015), San Diego, CA, USA, 7–9 May 2015.
74. Antun, V.; Renna, F.; Poon, C.; Adcock, B.; Hansen, A.C. On instabilities of deep learning in image reconstruction and the potential costs of AI. *Proc. Natl. Acad. Sci. USA* **2020**. [CrossRef]
75. Gottschling, N.M.; Antun, V.; Adcock, B.; Hansen, A.C. The troublesome kernel: Why deep learning for inverse problems is typically unstable. *arXiv* **2020**, arXiv:2001.01258.
76. Schwab, J.; Antholzer, S.; Haltmeier, M. Deep null space learning for inverse problems: Convergence analysis and rates. *Inverse Probl.* **2019**, *35*, 025008. [CrossRef]
77. Chambolle, A.; Pock, T. A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math. Imaging Vis.* **2011**, *40*, 120–145. [CrossRef]
78. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. In Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015), San Diego, CA, USA, 7–9 May 2015.
79. Hinton, G.; Srivastava, N.; Swersky, K. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. *Lect. Notes* **2012**, *14*, 1–31.
80. Winkler, C.; Worrall, D.; Hoogeboom, E.; Welling, M. Learning likelihoods with conditional normalizing flows. *arXiv* **2019**, arXiv:1912.00042.
81. Dinh, L.; Sohl-Dickstein, J.; Bengio, S. Density estimation using Real NVP. In Proceedings of the 5th International Conference on Learning Representations (ICLR 2017), Toulon, France, 24–26 April 2017.
82. Dinh, L.; Krueger, D.; Bengio, Y. NICE: Non-linear Independent Components Estimation. In Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015), San Diego, CA, USA, 7–9 May 2015.
83. Kingma, D.P.; Dhariwal, P. Glow: Generative Flow with Invertible 1x1 Convolutions. In Proceedings of the Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018 (NeurIPS 2018), Montréal, QC, Canada, 3–8 December 2018; Bengio, S., Wallach, H.M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R., Eds.; 2018; pp. 10236–10245.
84. Daubechies, I.; Defrise, M.; De Mol, C. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Commun. Pure Appl. Math.* **2004**, *57*, 1413–1457. [CrossRef]
85. Gregor, K.; LeCun, Y. Learning fast approximations of sparse coding. In Proceedings of the 27th International Conference on International Conference on Machine Learning, Haifa, Israel, 21–24 June 2010; pp. 399–406.

86. Lempitsky, V.; Vedaldi, A.; Ulyanov, D. Deep Image Prior. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 9446–9454. [CrossRef]
87. Dittmer, S.; Kluth, T.; Maass, P.; Otero Baguer, D. Regularization by Architecture: A Deep Prior Approach for Inverse Problems. *J. Math. Imaging Vis.* **2019**, *62*, 456–470. [CrossRef]
88. Chakrabarty, P.; Maji, S. The Spectral Bias of the Deep Image Prior. *arXiv* **2019**, arXiv:1912.08905.
89. Heckel, R.; Soltanolkotabi, M. Denoising and Regularization via Exploiting the Structural Bias of Convolutional Generators. *Int. Conf. Learn. Represent.* **2020**.
90. Adler, J.; Kohr, H.; Ringh, A.; Moosmann, J.; Banert, S.; Ehrhardt, M.J.; Lee, G.R.; Niinimäki, K.; Gris, B.; Verdier, O.; et al. Operator Discretization Library (ODL). *Zenodo* **2018**. [CrossRef]
91. Van Aarle, W.; Palenstijn, W.J.; De Beenhouwer, J.; Altantzis, T.; Bals, S.; Batenburg, K.J.; Sijbers, J. The ASTRA Toolbox: A platform for advanced algorithm development in electron tomography. *Ultramicroscopy* **2015**, *157*, 35–47. [CrossRef] [PubMed]
92. Coban, S. SophiaBeads Dataset Project Codes. Zenodo. 2015. Available online: http://sophilyplum.github.io/sophiabeads-datasets/ (accessed on 10 June 2020)
93. Wang, T.; Nakamoto, K.; Zhang, H.; Liu, H. Reweighted Anisotropic Total Variation Minimization for Limited-Angle CT Reconstruction. *IEEE Trans. Nucl. Sci.* **2017**, *64*, 2742–2760. [CrossRef]
94. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems 32*; Wallach, H., Larochelle, H., Beygelzimer, A., dAlch'e-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: Red Hook, NY, USA, 2019; pp. 8024–8035.

Paper 4

An educated warm start for deep-image-prior-based micro CT reconstruction

An Educated Warm Start For Deep Image Prior-Based Micro CT Reconstruction

Riccardo Barbano†, Johannes Leuschner†, Maximilian Schmidt, Alexander Denker, Andreas Hauptmann, Peter Maass and Bangti Jin

***Abstract*—Deep image prior (DIP) was recently introduced as an effective unsupervised approach for image restoration tasks. DIP represents the image to be recovered as the output of a deep convolutional neural network, and learns the network's parameters such that the model output matches the corrupted observation. Despite its impressive reconstructive properties, the approach is slow when compared to supervisedly learned, or traditional reconstruction techniques. To address the computational challenge, we bestow DIP with a two-stage learning paradigm: (i) perform a supervised pretraining of the network on a simulated dataset; (ii) fine-tune the network's parameters to adapt to the target reconstruction task. We provide a thorough empirical analysis to shed insights into the impacts of pretraining in the context of image reconstruction. We showcase that pretraining considerably speeds up and stabilizes the subsequent reconstruction task from real-measured 2D and 3D micro computed tomography data of biological specimens. The code and additional experimental materials are available at educateddip.github.io/docs.educated_deep_image_prior/.**

I. Introduction

Inverse problems in imaging center around recovering an unknown image $x \in \mathbb{R}^n$ of interest from the noisy measurement $y_\delta = Ax + \eta$, where $y_\delta \in \mathbb{R}^m$ is the noisy measurement data, A the linear forward operator, and η an i.i.d. noise (e.g. Gaussian noise $\eta \sim \mathcal{N}(0, \sigma^2 I)$). Due to the inherent ill-posedness of the problem, suitable regularization is crucial and is key for a successful recovery of x [1]–[3].

Over the last years, deep learning methods have been successfully applied to solve all types of imaging problems, with supervised training being the dominant paradigm [4], [5]. That means, a deep neural network is trained to restore the image from noisy data using a set of paired training data. A large number of such high-quality paired training data may be needed [6]. Except simulated data, these are usually not obtainable, or too expensive to collect. Further challenges arise from the distributional shifts of the test data (e.g. change of image class, noise level or forward operator at test time). Ideally, the trained model should be robust to these changes, and transfer its reconstructive properties from one domain to another using as little additional data as possible [7]–[9]. Unfortunately, this is often not the case.

An effective solution to these challenges is deep image prior (DIP) [10], which represents a new approach to regularize image restoration. Rather than taking the supervised route, DIP learns to reconstruct without reference data, by assuming that a natural image can be well represented by a convolutional neural network (CNN). This is achieved by training the network's parameters to generate an image that fits the data y_δ (often equipped with suitable early stopping). The method is very attractive for imaging tasks with scarce training data. DIP has received enormous attention in the imaging community, and delivered state-of-the-art performance for unsupervised methods on a number of imaging tasks, including computed tomography (CT) [6], [11], magnetic resonance imaging (MRI) [12], positron emission tomography (PET) [13]–[15] and compressive ptychography [16], closely matching its supervised counterparts.

While DIP has been shown to be effective, it is not free from drawbacks. Notably, it requires "fresh training" each time it is deployed, which leads to high computational overhead and demanding VRAM requirements at test time when compared to supervised counterparts [4], [17], [18]; the latter ones only require one feed-forward pass through the network and thus computationally cheap. This inefficiency is considerably exacerbated by the fact that DIP requires a lengthy (and unstable) optimization process [6], [19]. For example, reconstructing a single image of resolution $(501\,\mathrm{px})^2$ requires approximately 30-50k iterations to reach the early-stopping point, which translates to 3-5 h of computing-time on NVIDIA GeForce RTX 2080Ti/1080Ti. It gets even worse in the 3D setting: a $(167\,\mathrm{px})^3$ reconstruction takes approximately one day on NVIDIA GeForce RTX 3090 using mixed precision! This hinders its applicability to solve imaging inverse problems, especially when fast reconstruction is critical. These observations motivate us to explore the following:

> *Can DIP benefit from pretraining for accelerating subsequent reconstructive tasks? If so, can we easily construct an informative dataset to warm-start DIP? How do inductive biases of pretraining impact the reconstructive task?*

Pretraining is one well-established paradigm to address data scarcity in supervised learning [20], [21]. Models are often pretrained using large-scale datasets, and fine-tuned on target tasks that have less training data [22]. However, the idea of pretraining has not received the attention it deserves for

R. Barbano is with the Department of Computer Science, University College London, UK (e-mail: riccardo.barbano.19@ucl.ac.uk).

J. Leuschner, M. Schmidt, A. Denker and P. Maass are with the Center for Industrial Mathematics, University of Bremen, Germany (e-mail: {jleuschn, maximilian.schmidt,adenker,pmaass}@uni-bremen.de).

A. Hauptmann is with the Research Unit of Mathematical Sciences, University of Oulu, Finland, and also with the Department of Computer Science, University College London, UK (e-mail: andreas.hauptmann@oulu.fi).

B. Jin is with Department of Mathematics, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong (e-mail: btjin@math.cuhk.edu.hk,bangti.jin@gmail.com)

† equal contributors.

 DOI: 10.1109/TCI.2022.3233188

DIP, and presents a new challenge. The challenge is to learn (via supervised pretraining) feature representations that are transferable and generalizable to subsequent fine-tuning.

To overcome the computational challenge, we systematically explore a supervised pretraining strategy for accelerating DIP-based μCT reconstruction, and introduce an effective two-stage learning paradigm. Our contributions can be summarized as follows. We develop an effective strategy to greatly accelerate the convergence of DIP for μCT reconstruction, by recasting DIP within the "supervised pretraining + unsupervised fine-tuning" paradigm. We show that carefully designed pretraining with simulated data from a synthetic image class can considerably speed up and stabilize DIP-based μCT reconstruction with real-measured data, including computationally demanding 3D tasks, for which we develop a specialized U-Net architecture to perform DIP-based μCT reconstruction under the constraint of 24 GB VRAM. To the best of our knowledge, this is the first successful 3D μCT reconstruction using DIP. The experiment results show that despite its simplicity, it can be highly effective. Further, we conduct a thorough experimental study to shed insights into the mechanism of knowledge transfer between the supervised pretraining and unsupervised fine-tuning stages, including a novel linear analysis of pretraining, which exhibits sparsity-promoting in the parameters' bases.

The paper is organized as follows. We describe the standard DIP in Section II and related works in Section III. In Section IV, we present the two-stage framework for DIP. We give the experimental details and results in Sections V and VI, and analyze the impact of pretraining in Section VII.

II. Deep Image Prior

The idea of DIP [10] is to find a minimizer of the fidelity $\|Ax - y_\delta\|^2$, by representing the unknown x as the output of a CNN, $x = \varphi_\theta(z)$, where $z \in \mathbb{R}^n$ is a fixed random vector (often pixel-wise i.i.d. samples of random noise), and $\theta \in \mathbb{R}^p$ denotes the network's parameters to be learned. A U-Net [23] like architecture is commonly used for the network. DIP solves

$$\theta^* \in \operatorname*{argmin}_\theta \|A\varphi_\theta(z) - y_\delta\|^2,$$

and presents $\varphi_{\theta^*}(z)$ as the reconstruction. Note that the training of the network parameters θ coincides with the recovery process, and has to be repeated for each measurement. The procedure is unsupervised, and guided by the principle of matching the forward projected network output $A\varphi_\theta(z)$ to the measurement data y_δ. Due to the overparameterization of the neural networks used in DIP, a direct minimization of the loss can suffer from overfitting. DIP often uses early-stopping to deliver a satisfactory reconstruction: the update of θ is stopped early to avoid overfitting to the noise [10]. This has motivated developing automated rules for early stopping [24], [25].

III. Related Works

a) Deep Image Prior: Since the first proposal in [10], there have been several important developments on DIP. Heckel et al. [26] propose deep decoder, using under-parameterized networks to ease the need for early-stopping. Dittmer et al. [27] study DIP through the lens of regularization theory [1]–[3], and Cheng et al. [28] discuss its connection with Gaussian processes as the number of architecture channels grows to infinity, and propose the use of Bayesian learning. There are several efforts to combine DIP with explicit regularization to improve the reconstruction quality. [6], [29] propose the use of total variation penalty for stabilizing the learning process, and [30] combines DIP with regularization by denoising. Besides, the use of explicit regularization significantly relaxes the need of early stopping. Jo et al. [24] propose to penalize the complexity of the reconstruction using Stein's unbiased risk estimator. See also [25] for a stopping criterion based on monitoring the running variance of iterate sequence and references therein for further discussions. [6] suggests at test-time to start optimizing a randomly initialized DIP to match a reconstruction, produced by another method. Heckel and Soltanolkotabi [31] prove that for compressed sensing, an untrained CNN can approximately reconstruct signals and images that are sufficiently structured, from a near minimal number of random measurements. The very recent work [32] establishes the equivalence of "analytic" DIP with the standard Tikhonov regularization, and several basic properties in the lens of classical regularization theory. This work complements and expands on these existing studies by addressing the computational challenge associated with regularized DIP, especially the works [6], [29], where the regularized DIP was proposed and empirically demonstrated.

b) Advances in Pretraining: Supervised pretraining on ImageNet has been established as a common practice in computer vision. Neural networks are pretrained to solve image classification, and transferred to downstream tasks (e.g. object detection [33], [34] and semantic segmentation [35]). However, pretraining on ImageNet does not necessarily improve the accuracy of the downstream task [36], and similar observations about pretraining on ImageNet are made about medical image classification [37]. Within tomographic imaging, several works [38], [39] employ transfer learning to adapt a trained neural network from one task setting to another. Our work shares similarities with these works in adapting to changes of the image distribution. These works focus on supervised end-to-end fine-tuning, whereas we focus on an unsupervised learning framework: we study pretraining with a synthetic dataset as a means for accelerating DIP reconstruction on measured μCT data, and provide a detailed analysis of its acceleration mechanism. Very recently, Gilton et al [7] proposes to fine-tune the pretrained model so as to accommodate model errors, but unlike this work, the image distribution is unchanged. Inspired by an early version of this paper, Knopp and Grosser [11] also demonstrated the potential of warm-starting DIP for dynamic tomography.

IV. Proposed Method

The TV-regularized DIP approach obtains x^* by

$$\theta_{\mathrm{t}}^* \in \operatorname*{argmin}_\theta \Big\{ l_{\mathrm{t}}(\theta) := \|A\,\varphi_\theta(z) - y_\delta\|^2 + \gamma\,\mathrm{TV}\,(\varphi_\theta(z)) \Big\},$$
$$x^* = \varphi_{\theta_{\mathrm{t}}^*}(z), \tag{1}$$

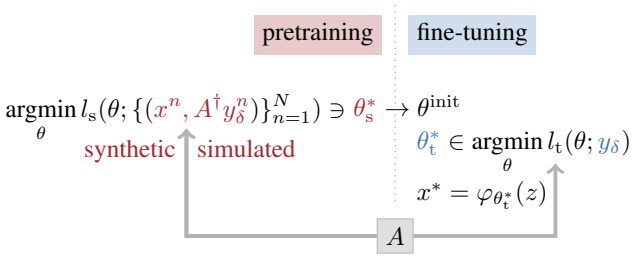

Fig. 1. A two-stage learning paradigm. The parameters θ of the U-Net are first optimized on a dataset comprising ordered pairs of synthetic ground truth images x^n and simulated measurement y_δ^n. The optimal configuration θ_s^* is then used to warm-start the unsupervised fine-tuning on real μCT data.

where φ_θ is a CNN, and $\gamma \geq 0$ balances the data consistency with the regularization term $\mathrm{TV}(\varphi_\theta(z))$, which denotes the total variation seminorm on the network output $\varphi_\theta(z)$, defined as $\mathrm{TV}(x) = \|\nabla_h x\|_1 + \|\nabla_v x\|_1$, where ∇_h and ∇_v denote the derivative in the horizontal and vertical directions. Several studies [6], [29] found that incorporating the total variation penalty is beneficial to DIP. The loss l_t in (1) is optimized with Adam [40], by randomly initializing θ. The learning is performed as (single-batch) test time adaptation to y_δ.

In this work, we recast DIP into the "supervised pretraining + unsupervised fine-tuning" paradigm as a two-stage process, called educated DIP (EDIP); see Fig. 1 for a schematic illustration of the framework. In the first stage, we pretrain the network $\varphi_\theta(A^\dagger y_\delta)$, where $A^\dagger$ is an approximate inverse operator (e.g. filtered back-projection (FBP) for CT [41]). The training is carried out on a synthetic dataset $\mathcal{D} = \{(x^n, y_\delta^n)\}_{n=1}^N$, composed of N pairs drawn from the joint distribution of ground truth x^n and corresponding simulated measurement y_δ^n. This step is tailored to the target reconstruction task in (1), and learns the optimal parameters θ_s^* via supervised training,

$$\theta_\mathrm{s}^* \in \operatorname*{argmin}_\theta \Big\{ l_\mathrm{s}(\theta) := \frac{1}{N} \sum_{(x^n, y_\delta^n) \in \mathcal{D}} \|\varphi_\theta(A^\dagger y_\delta^n) - x^n\|^2 \Big\}. \quad (2)$$

Note that φ_θ receives $A^\dagger y_\delta^n$ as its input (instead of the random noise in [10]), serving as a post-processing reconstructor [42]. The objective of this stage is to enforce "benignant" inductive biases via supervised learning. This educates DIP with knowledge contained in the dataset $\mathcal{D}$, which is then exploited, but still needs to be amended, in solving the reconstruction task in (1).

In the second stage, for a given new query measurement y_δ, we use the optimal parameters θ_s^* obtained in the pretraining stage to initialize the network $\varphi_\theta(A^\dagger y_\delta)$ in (1) so as to get DIP up to speed in handling target tasks on real-measured data. That is, we regard the DIP optimization as a self-adaptation step, where the parameters θ are fine-tuned unsupervisedly, with their drift conditioned on θ_s^*. Note that the robustness of this method at test time does not rely solely on how well the pretraining stage anticipates distributional shifts. The model makes a good use of pretraining — the supervised pretraining stage sets and constrains the stage — but adapts to distributional shifts at test time, and reserves its right to amend the received supervision.

There are several possible variants of the basic framework. U-Net consists of two parts, a decoder with parameters θ_dec, and an encoder with parameters θ_enc. A direct variant of EDIP is to fine-tune only the decoder parameters θ_dec, but fixing the encoder parameters to the educated guess $\theta_{\mathrm{s,enc}}^*$, which are regarded as a shared (between stages) feature extractor. At test time, we solve (1) only with respect to θ_dec and rely on the pretraining to construct a suitable "universal" encoding. Thus, the learned reconstructor φ_{θ^*} recovers from the measurement data with $\theta^* = (\theta_{\mathrm{s,enc}}^*, \theta_{\mathrm{t,dec}}^*)$. This variant with the fixed encoder (FE) is termed as EDIP-FE.

V. Datasets

A. Synthetic Training Dataset

We pretrain on a synthetic training dataset of images composed of ellipses or ellipsoids with random position, shape, orientation and intensity values, which are commonly used to train and evaluate learned reconstruction methods. This image class encompasses basic building blocks of more complex images, while favoring piece-wise smoothness. Synthetic data is particularly useful when it is infeasible to collect high-quality ground truth images reassembling the image class of the target reconstructive task, while enabling the learning of features tailored to the inversion of the forward operator A. In the experiments, we use datasets of 32 000 training and 3200 validation images generated on-the-fly using ODL [43]. The image resolution and the distribution of the ellipses / ellipsoids can be easily adapted to match different target data. The synthetic projection data is computed by forward projecting the ground truth images and adding 5 % white noise. Fig. 2 shows an exemplary ground truth image and reconstructions obtained by the FBP and U-Net from the simulated noisy data.

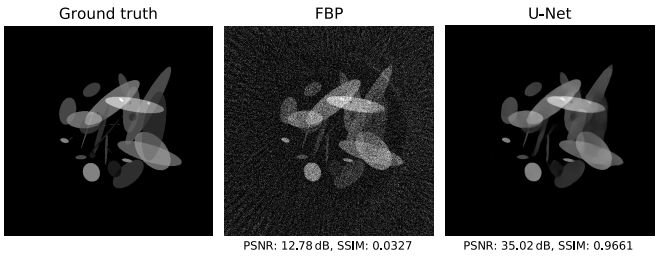

Fig. 2. An exemplary ground truth image used in the pretraining stage. The FBP and U-Net reconstructions are also shown. The measurement data y_δ is simulated using the Walnut `Sparse 120` setting, adding 5% white noise.

B. Real μCT Measurement Data

We evaluate our approach on two real μCT datasets to showcase the effectiveness of the approach. The forward operator A is a ray transform matching a 2D or 3D cone-beam geometry (cf. Fig. 3 for 2D). The scanner rotates around the object (or, equivalently, the object is rotated inside the scanner), taking projections from different source angles λ. Within each projection, each detector pixel (e.g. parameterized by γ) measures the intensity for a specific line, attenuated by the object.

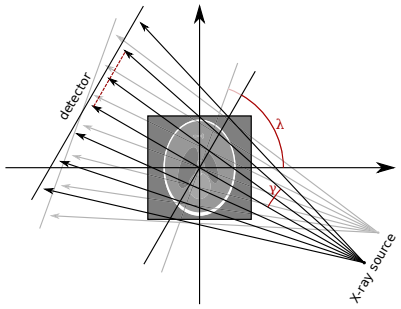

Fig. 3. Diagram of the 2D cone-beam geometry (a.k.a. fan-beam geometry).

a) X-ray Lotus Root Dataset: μCT measurements of a Lotus root slice filled with different materials are available from [44]. The dataset contains fan-beam measurements corresponding to a 2D volume slice, with 120 projections at angles equally distributed over $[0, 360°)$ and 429 detector pixel values each. The sparse matrix modeling the forward operator for an image resolution $(128\,\mathrm{px})^2$ is used. In the evaluation, we consider the setting of `Sparse 20`: a 6-fold angular sub-sampling, 20 angles, equally distributed over $[0, 360°)$. We use a TV-regularized reconstruction from all 120 projection angles, obtained by Adam, as the reference solution.

b) X-ray Walnut Dataset: A collection of cone-beam μCT measurement data from 42 Walnuts was provided in [45]. For each walnut, a set of three 3D cone-beam measurements is included, each obtained with a different source position. Projections are acquired at 1200 angles equally distributed over $[0, 360°)$, with a resolution of 972 detector rows and 768 detector columns. A volume resolution of $(501\,\mathrm{px})^3$ is used. We consider reconstructing a single 2D slice from a suitable subset of detector pixel measurements, and 3D reconstruction with a downscaled image resolution of $(167\,\mathrm{px})^3$. For the 2D task, we use the setting of `Sparse 120`: a 10-fold angular sub-sampling with 120 angles, equally distributed over $[0, 360°)$; for 3D, we consider the settings `3D Sparse 20` and `3D Sparse 60` with 20 and 60 equally distributed angles, and sub-sample the projection rows and columns by a factor of 3. The 3D settings are chosen to mimic industrial applications, where a high degree of sparsity is often desired. The approximations $A^\dagger y_\delta$ are computed via the Feldkamp-Davis-Kress (FDK) algorithm [46]. FDK is an FBP-based algorithm with a weighting step for cone-beam measurements, and is still denoted as "FBP". To achieve accurate automatic differentiation of the forward projection operator in 2D, we utilize its sparse matrix representation. In 3D, we opt for forward and backward projection routines of ASTRA via tomosipo [47]. We use the ground truth provided with the dataset [45], which was obtained with accelerated gradient descent using the measurements from all 1200 projection angles and all three source positions.

VI. Experiments and Results

Throughout, we denote the type of the network input z used for a method in brackets: for example, "DIP (noise)" refers to the standard DIP with noise input, while "EDIP (FBP)" stands for the educated DIP with FBP input.

A. Neural Network Architecture

For 2D settings, we adopt the U-Net proposed by [6], but replace batch-normalization layers with group-normalization layers. For 3D μCT reconstructions, we fine-tune the architecture, cf. Fig. 4, since the standard 3D U-Net — originally introduced for segmentation [48] — does not meet our memory constraint, and a naively reduced version leads to sub-optimal reconstructions. We modify the U-Net architecture as follows: (i) reduce the numbers of channels per convolutional layer in the encoder; (ii) increase the expressivity of the decoder by chaining subsequent convolutional layers with decreasing number of channels; (iii) remove skip connections. Due to memory constraints (i.e. 24 GB VRAM), we use a 3-scale 3D U-Net.

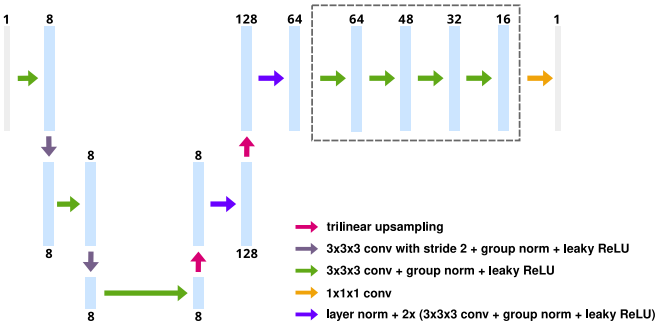

Fig. 4. The architecture of the proposed 3D U-Net. Each light-blue bar corresponds to a multi-channel feature map. Arrows denote the different operations.

B. Evaluation Metrics

We measure the reconstruction quality via peak signal-to-noise ratio (PSNR), and include structural similarity index measure (SSIM) [49] for reconstructions. To assess the convergence speed, we employ two metrics: steady PSNR and rise time (denoted by $\star$ in the figures). The steady PSNR is the median PSNR over the last 5k iterations. The rise time is the iteration number at which we reach the baseline PSNR (i.e. DIP's steady PSNR) up to a threshold $0.1\,\mathrm{dB}$. In addition, we always consider the iteration-wise median PSNR over repeated runs of the same experiment (with varying seeds) for these metrics; we use 5 runs for 2D and 3 runs for 3D. The variability between runs does arise not only from random initialization of the network parameters or noise input, but also from numerical effects in parallel computations on GPU. The optimal reconstruction $\varphi_{\theta_{\text{min-loss}}}(z)$ is taken from the iteration with minimum loss value $l_\mathrm{t}(\theta_{\text{min-loss}}) = \min_{i \in 0 \dots N} l_\mathrm{t}(\theta^{[i]})$. This remedies non-monotonous loss minimization, yet the (E)DIP optimization plots and steady PSNR computations use the actual iterate to facilitate a direct analysis.

C. Hyperparameter's Selection

The learning rate and regularization parameter γ are fine-tuned for standard DIP. For EDIP, we do not conduct additional hyperparameter search, but use the values identified for DIP. These hyperparameters values also perform well for EDIP,

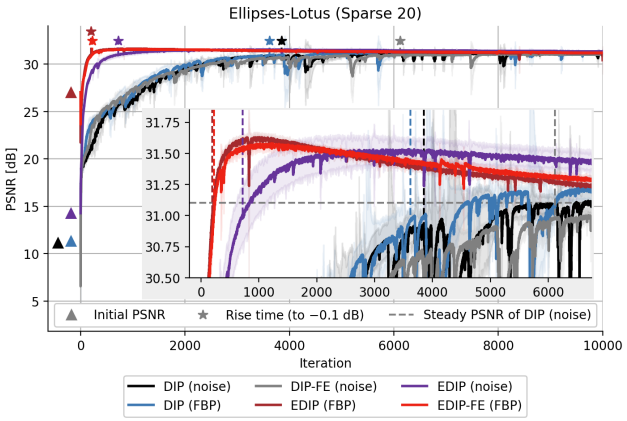

Fig. 5. The optimization of EDIP versus DIP on Lotus `Sparse 20`. The symbols ⋆ and ▲ denote initial PSNR and rise time, respectively, and the horizontal dashed line indicates the steady PSNR of DIP (noise).

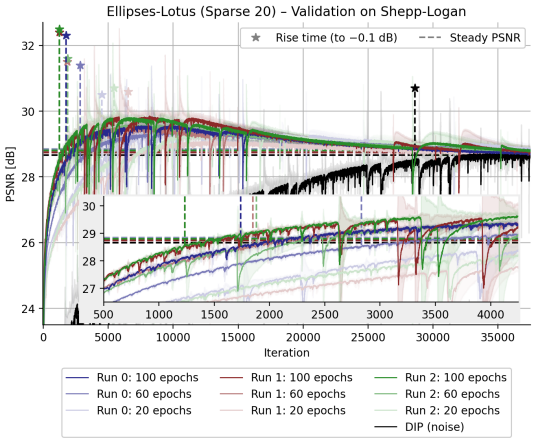

Fig. 6. Checkpoint selection on the Shepp-Logan phantom for the initial EDIP (FBP) model parameters for the Lotus `Sparse 20` setting.

which saves us from performing an individual search for each pretraining checkpoint (to be defined next).

D. Selection of the Checkpoints

Multiple parameters' configurations (i.e. θ_s^*) may be obtained from the pretraining stage: repeated runs (varying random initializations) and multiple checkpoints along the optimization trajectory of each run. From a set of checkpoints, one needs to identify solutions maximizing the speed-up at test time. More broadly, this is an open question. In the experiments below, the selection strategy is based on assessing the performance on the Shepp-Logan phantom [50], a standard test image within the medical imaging community. The checkpoint leading to the shortest rise time is then selected, among those with a steady PSNR that is at most $0.25\,\mathrm{dB}$ lower than the maximum steady PSNR of any checkpoint. This selection is carried out for 2D reconstruction settings; 3D runs use the best performing checkpoint for computational reasons. For Lotus `Sparse 20`, we repeat the pretraining 3 times (varying the seed) and collect checkpoints after every 20 epochs, training for a maximum of 100 epochs. For the Walnut `Sparse 120`, we pretrain for 20 epochs, and retain the minimum validation loss checkpoint of each run. For the 3D Walnut settings, we pretrain for a maximum of 2 epochs, and retain checkpoints every 0.125 epochs (i.e. 4k gradient updates).

E. The Lotus Root

Table I shows the convergence properties of EDIP and DIP for Lotus `Sparse 20`. We include in our analysis cases where the FBP $A^\dagger y_\delta$ is fed as the input (instead of noise) when solving (1) for DIP, and inputting noise for EDIP. EDIP significantly outperforms DIP in terms of the convergence speed for either a fixed noise image or FBP.

EDIP only takes 195 (and 723 for noise input) iterations to reach $-0.1\,\mathrm{dB}$ of the baseline PSNR, against 4.1k iterations needed for DIP. Thus, pretraining greatly accelerates the convergence. The optimization process is considerably more stable (cf. Fig. 5), implying a possibly much more favorable loss landscape for EDIP. Thus, pretraining stabilizes the optimization process of DIP, which is highly desirable in practice. Note that EDIP-FE, which fixes the encoder parameters to the pretrained ones $\theta^*_{s,\mathrm{enc}}$, is as fast as EDIP, and the reconstruction quality of EDIP and EDIP-FE are largely comparable with each other. With fewer parameters to be updated, EDIP-FE is computationally lighter than EDIP (since backpropagation is only needed for the decoder, and the forward pass through the encoder can be pre-computed beforehand). Fig. 7 shows the reconstruction (along with the reference and FBP) for Lotus `Sparse 20`. We observe that pretraining can also boost the performance of DIP: EDIP considerably overshoots the baseline PSNR, cf. Fig. 5. This suggests that pretraining, if coupled with proper early-stopping (approximately a few hundred iterations after the rise time),

TABLE I
QUANTITATIVE EVALUATION FOR THE LOTUS `SPARSE 20`.

Ellipses-Lotus `Sparse 20`	Rise time	(Max PSNR; iters)	Steady PSNR
DIP (noise)	3848	(31.17; 8846)	31.10
DIP (FBP)	3622	(31.25; 8813)	31.17
DIP-FE (noise)	6118	(31.10; 9818)	31.00
EDIP (FBP)	195	(31.65; 981)	31.21
EDIP (noise)	723	(31.53; 3548)	31.39
EDIP-FE (FBP)	226	(31.59; 1421)	31.26
TV	–	–	30.73

TABLE II
CHECKPOINTS' COMPARISON FROM THE PRETRAINING STAGE FOR EDIP (FBP) ON LOTUS `SPARSE 20`. THE CHECKPOINT FROM RUN 2 AFTER 100 EPOCHS IS SELECTED USING THE SHEPP-LOGAN DATA (CF. FIG. 6)

	Epochs	Rise time	(Max PSNR; iters)
Run 0	100	247	(31.49; 1545)
	60	174	(31.56; 842)
	20	291	(31.61; 1614)
Run 1	100	162	(31.53; 779)
	60	243	(31.53; 1755)
	20	390	(31.56; 1518)
Run 2	100	195	(31.65; 981)
	60	194	(31.58; 1083)
	20	318	(31.51; 1706)

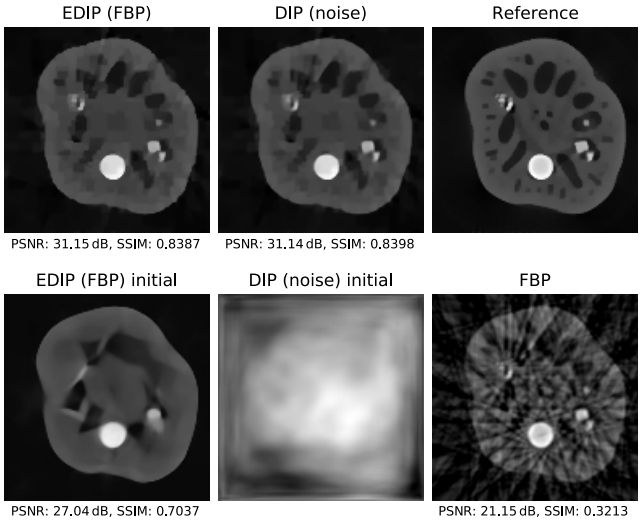

Fig. 7. EDIP versus DIP reconstruction on Lotus `Sparse 20`. From the 5 runs (varying the seed), the one with the (closest to) median PSNR was selected for each method.

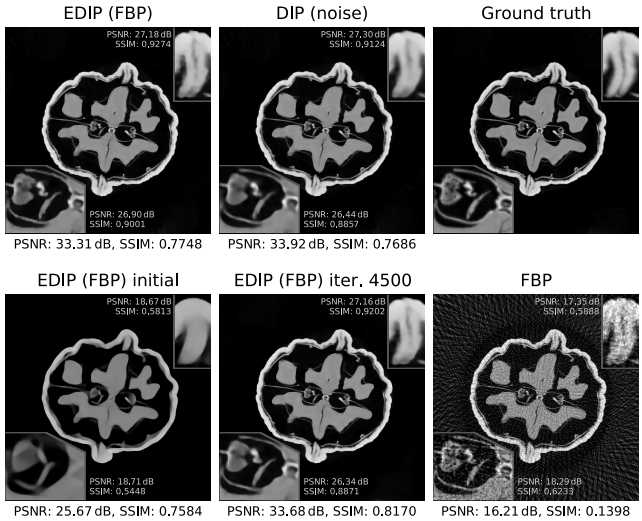

Fig. 8. EDIP versus DIP reconstruction of Walnut `sparse 120`.

TABLE III
QUANTITATIVE EVALUATION FOR THE WALNUT.

Ellipses/Ellipsoids-Walnut	`Sparse 120`			`3D Sparse 20`			`3D Sparse 60`		
	Rise time	(Max PSNR; iters)	Steady PSNR	Rise time	(Max PSNR; iters)	Steady PSNR	Rise time	(Max PSNR; iters)	Steady PSNR
DIP (noise)	20 373	(34.02; 25 357)	33.87	17 200	(30.68; 23 477)	30.37	49 041	(34.05; 58 901)	33.93
DIP (FBP)	13 778	(34.07; 28 094)	33.90	13 016	(31.32; 25 063)	31.19	27 873	(34.37; 53 731)	34.22
EDIP (FBP)	4496	(33.92; 13 039)	33.56	3739	(31.48; 10 689)	30.94	11 247	(34.35; 40 810)	34.18
EDIP-FE (FBP)	4384	(33.91; 12 540)	33.70	2979	(31.38; 10 749)	30.93	14 520	(34.33; 45 259)	34.15
TV	–	–	31.67	–	–	28.89	–	–	33.35

can lead to better reconstructions.

To maximize the speed-up, we select the warm-start configuration θ_s^* on the Shepp-Logan phantom. Fig. 6 shows the validation runs. We select θ_s^* from run 2 after 100 epochs since it results in the smallest rise time. Interestingly, a substantial overshoot of baseline PSNR is observed on the Shepp-Logan phantom, possibly due to its in-distribution nature with respect to the ellipses. Table II reports the rise time at test time for different checkpoints collected for each run. The results indicate that the checkpoint selection does impact the achievable acceleration factor, but not the maximum PSNR.

F. The Walnut

Fig. 8 shows the reconstructed Walnut slice; see Table III for quantitative results. A speed-up is observed, similar to the Lotus root: EDIP takes about 30 min at rise time (approximately 4.4k iterations), whereas DIP (with noise input) takes 2 h and 30 min at rise time (approximately 20.4k iterations) with NVIDIA GeForce RTX 2080Ti. A TV regularized reconstruction of the Walnut takes 6 min, and requires 1.7k gradient steps to converge to 31.67 dB. EDIP takes only 3 min (after 421 iterations) to match 31.67 dB. In 6 min, EDIP reaches 32.80 dB, with a gain of 1.1 dB. Finally, DIP-FE / EDIP-FE report similar performances to DIP / EDIP.

On a minor note, it is observed that EDIP better reconstructs finer structures (e.g. the wrinkled shell), and DIP suffers from over-smoothing artifacts. This concurs with the observation for Lotus `Sparse 20`: by incorporating the knowledge contained in the synthetic training data, pretraining can boost the performance of DIP.

Similar observations can be made for reconstructing the Walnut volume, cf. Fig. 9 for 3D reconstructions along the yz, xz, and xy axes and Table III for quantitative results. EDIP reconstruction from the `3D Sparse 20` data takes approx. 1.5 h with a NVIDIA GeForce RTX 3090, and leads to 33.77 dB in PSNR, compared to 7.3 h and 5.53 h for DIP (with noise / FBP as input). EDIP matches the PSNR of a TV reconstruction in about 30 min, gains 1 dB over TV after additional 20 min, and it takes 2.3 h to observe a 2 dB gain. The `3D Sparse 60` leads to similar speed-up. It takes 20 h for DIP with noise as input. Inputting the FBP results already in a considerable speed-up (about 11 h), whereas EDIP requires only 4 h. In sum, pretraining on the synthetic ellipsoids dataset greatly accelerates the convergence of DIP for 3D μCT reconstruction.

Last, we briefly comment on the convergence of the optimization process, cf. Fig. 10. The overall convergence behavior for 2D and 3D is similar to Lotus `Sparse 20`: pretraining stabilizes DIP optimization and greatly accelerates the convergence. Fig. 11 shows the convergence and stability of the loss in (1). The variation of the loss value is reduced if EDIP is used. As a practical post-hoc strategy to overcome the instability of the DIP optimization scheme, the reconstructed image is taken as the network output at minimum loss.

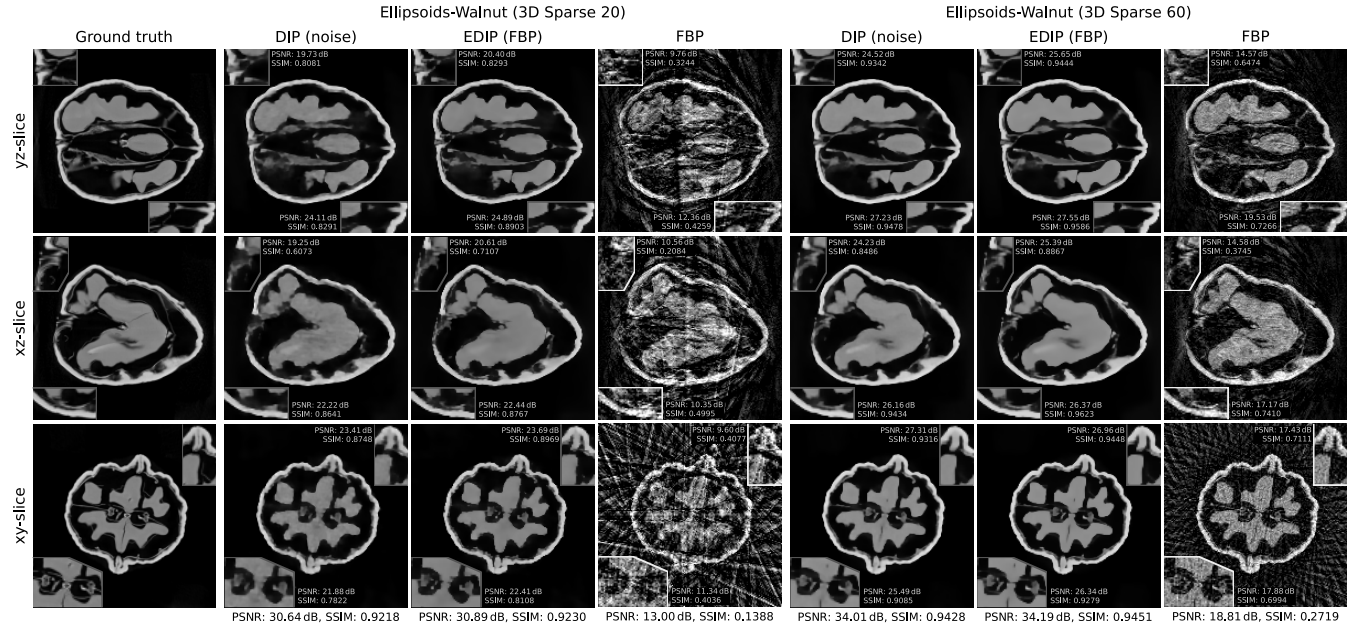

Fig. 9. 3D Walnut reconstruction of EDIP pretrained on ellipsoids dataset, compared to standard DIP, at three different slices.

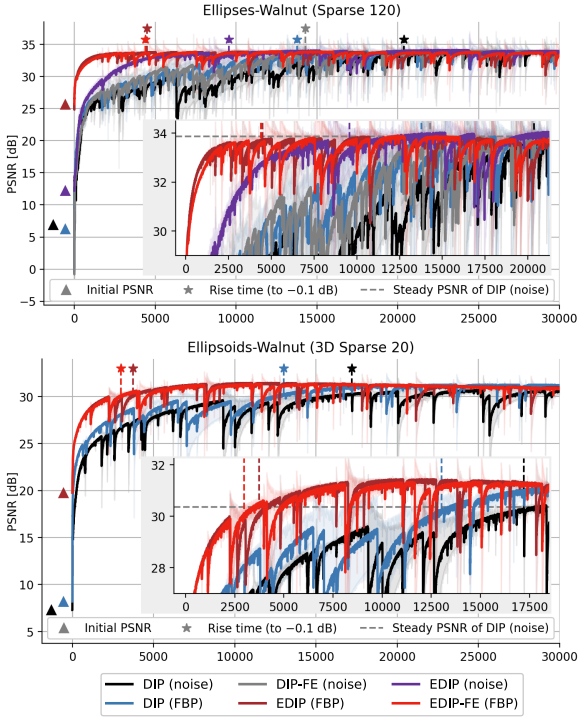

Fig. 10. The optimization of EDIP versus DIP for the Walnut reconstruction in 2D (top) and 3D (bottom). The symbols ⋆ and ▲ denote initial PSNR and rise time, respectively, and the horizontal dashed line indicates the steady PSNR of DIP (noise).

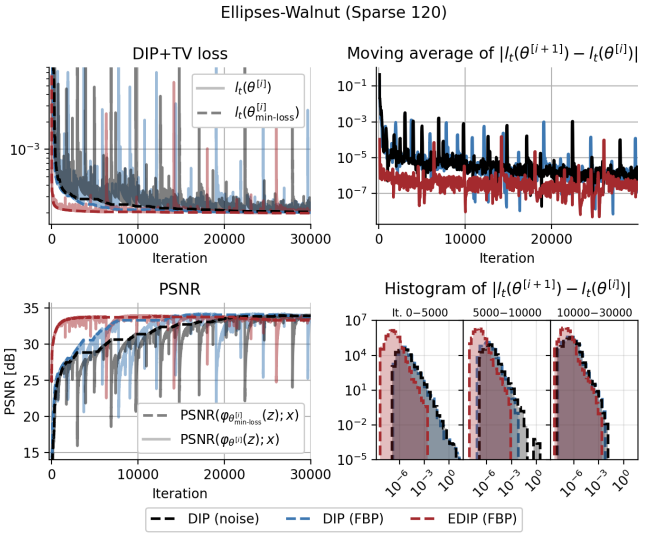

Fig. 11. The min-loss and PSNR computed with the min-loss output for Walnut `Sparse 120` (left). Loss variation (i.e. $|l_t(\theta^{[i+1]}) - l_t(\theta^{[i]})|$) and respective histograms computed over three intervals (right). The moving average uses a window size of 100 iterations.

VII. Investigation of the Role of Pretraining

In this section, we first motivate why we use a standard pretraining strategy instead of resorting to more sophisticated schemes, and then we shed insight into the mechanism of knowledge transfer via pretraining, highlighting favorable as well as detrimental properties.

a) Standard, Adversarial, Meta?: In this work, we adopt the standard pretraining paradigm within our pretraining stage, as described in Sec. IV. The choice is informed by comparing standard pretraining, adversarial pretraining [51], [52] and model agnostic meta-learning (MAML) [53], [54] on the Lotus `Sparse 20`. Adversarial pretraining uses a projected gradient descent attack (PGD-L_2 [55], [56]). MAML-based pretraining obtains a parameters' configuration training on six different tasks, comprising three different image classes: ellipses, rectangles [57], and natural images from the PASCAL VOC segmentation dataset [58], as well as two different noise distributions: Gaussian and Poisson. We do not vary

the forward operator A, since the pretraining stage is tailored to a known acquisition geometry; varying the structure of A (e.g., via sparsification) would only withhold from the model operator-specific knowledge, and introduce artifacts that are not expected to be found in the subsequent reconstruction tasks. We investigate whether the parameters' configurations, found with adversarial pretraining and MAML, lead to general representations adapting faster to the subsequent reconstruction problem. It is observed from Tab. IV that all three pretraining strategies lead to parameters' configurations that adapt to the subsequent reconstruction task with approximately similar speed-up. Even if the adaptation to the subsequent task shows on par properties, adversarial pretraining and MAML introduce a significant computational overhead, which we find unnecessary. The latter can be attributed to the facts that adversarial pretraining requires the inclusion of an inner loop optimization to design the attack (adding 62h to the wall-clock time); MAML's outer loop updates θ_{s}, while the inner one (with one step of stochastic gradient descent) adapts θ_{s} to a given task. MAML, instead, increases ($\times 5$) the overall VRAM required.

TABLE IV
QUANTITATIVE EVALUATION OF ALTERNATIVE PRETRAINING STRATEGIES FOR THE LOTUS ALONG WITH THE WALL-CLOCK TIME RECORDED ON A NVIDIA RTX 2080TI.

Ellipses-Lotus `Sparse 20`

	Rise time	(Max PSNR; iters)	(VRAM; batch size)	Time
EDIP (FBP)	195	(31.65; 981)	(5941 MiB; 32)	23h
Adv.-L_2-EDIP (FBP)	143	(31.24; 1175)	(6093 MiB; 32)	85h
MAML-EDIP (FBP)	545	(31.54; 1512)	(7949 MiB; 8)	31h

b) In Need to Amend: Figs. 7 and 8 show that the reconstructions obtained by directly deploying the pretrained network (i.e. $\varphi_{\theta_{\mathrm{s}}^*}$) on the FBP of the real-measured μCT data do enjoy good reconstructive properties, but the images tend to be overly-smooth and severely affected by ellipses-like artifacts, which are naturally present in the synthetic training dataset. Indeed, initializing the network's parameters to the pretrained configuration, on both Lotus and Walnut, shows a gain of 5.8 dB, and of 9.4 dB (`Sparse 120`), 6.7 dB (`3D Sparse 20`), 2 dB (`3D Sparse 60`) over the FBP. The pretrained model enjoys high input-robustness, and feature reuse plays a very important role in the EDIP reconstruction. However, the feature reuse mechanism leads to undesirable hallucinatory behaviors, as evidenced by the ellipses-like artifacts, which is a form of inductive biases induced by the synthetic image class. This also indicates the importance of properly designing the synthetic dataset used in the pretraining stage, from which the features are learned, and the strong dissimilarity between the synthetic training data and real test data may actually deteriorate the performance. In the supplementary materials, we showcase one potential pitfall of the "supervised pretraining + unsupervised fine-tuning" paradigm for DIP, resorting to synthetic data generated by a by far too specific and less diverse image class, i.e., human brain images for the supervised learning stage.

The knowledge enforced via the synthetic dataset needs to be properly amended so that the reconstructed images recover a more realistic texture. This is achieved at the fine-tuning stage by enforcing the data consistency. Amending the knowledge acquired via pretraining protects from hallucinations due to (inevitable) distributional shifts, thereby overcoming a well-known drawback of supervised learned reconstructors [59].

c) Investigating Feature Reuse: In a similar spirit to [60], we feed a noise image to EDIP (trained on pairs of FBP and ground truth image), which makes any visual features learned in the pretraining stage useless. This allows us to disentangle influencing factors involved in the fine-tuning stage. We consistently observe faster convergence of EDIP with respect to the standard DIP for the Lotus dataset. EDIP (fed with FBP) still results in faster convergence, which agree well with the intuition that decreasing feature reuse leads to diminishing benefits. Fig. 12 (left) shows that EDIP remolds the noise image differently compared to the standard DIP. The learned inductive biases prioritize reshaping the noise image as ellipse-like structures. The model makes an educated reconstruction. The features learned during pretraining are invariant of the input. The pretrained model is then adapted by enforcing data-consistency via (1).

On the Walnut, cf. Fig. 12 (right), the benefit of pretraining is less pronounced, if a noise image input is used. This might be due to the fact that the Walnut has a higher resolution and many more fine details, which are not present in the training dataset. Nonetheless, pretraining can still remold noise input into a walnut faster than DIP, yet the FBP input (used in the pretraining) is even more effective. These observations fully agree with that for Lotus.

d) Getting θ_{s}^* Right: The starting point θ_{s}^* of fine-tuning can impact the adaptation speed. A selection procedure of θ_{s}^* is desired to maximize transferable performance (e.g. speed-up). On the 2D setting, pretraining for more epochs (100 vs. 20) leads to a faster adaptation. This is clearly observed on the Lotus, possibly due to the in-distribution nature of the image class with respect to the ellipses dataset. However, on more complex tasks (`3D Sparse 20` and `3D Sparse 60`), extensive pretraining leads to overfitting the image class, and enforcing dataset-specific knowledge appears detrimental to the transfer. Fig. 13 shows that extensively pretraining U-Net for 2 epochs (i.e. 64k gradient updates with 32k ellipsoid volumes), albeit yielding the highest initial PSNR, leads to a sub-optimal convergence: the network output is effectively constrained, as an over-trained ϕ_{θ^*} after 2 epochs has little freedom to amend. This is also observed on the `3D Sparse 60` setting.

e) Spectral Evaluation: We propose a spectral analysis to understand the "education" by linearizing the non-linear forward map $F(\theta) = A\varphi_\theta(A^\dagger y_\delta)$ at θ_0:

$$F(\theta) = F(\theta_0) + F'(\theta_0)(\theta - \theta_0),$$

with $F'(\theta_0) = A\varphi'_{\theta_0} \in \mathbb{R}^{m\times p}$ with $\varphi'_{\theta_0} = \partial\varphi_\theta/\partial\theta|_{\theta=\theta_0} \in \mathbb{R}^{n\times p}$ denoting the Jacobian of the network's output w.r.t. θ. We use the subspace spanned by leading right singular vectors v_i of $F'(\theta_0)$ (i.e. with the largest singular values) as a faithful representation of the network's parameter space, which determines the dynamics of the learning process. Due to the high-dimensionality of the output and parameter spaces,

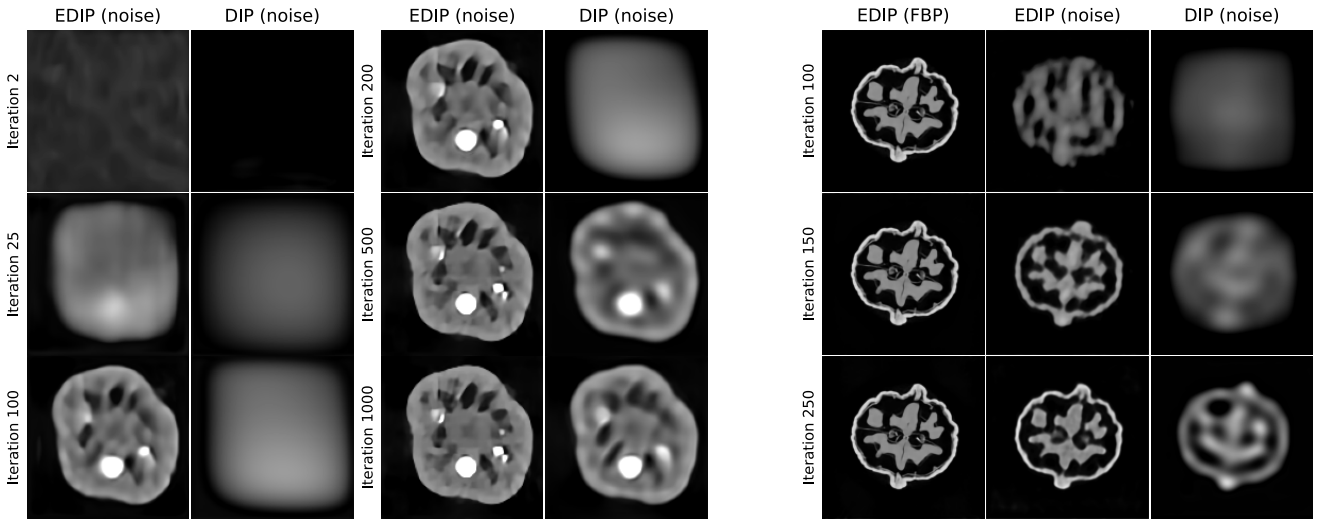

Fig. 12. Iterates collected throughout the EDIP/DIP reconstruction from Lotus `Sparse 20` (left) and Walnut `Sparse 120` (right), after different numbers of iterations. A video showing the reconstruction process is available at https://educateddip.github.io/docs.educated_deep_image_prior/.

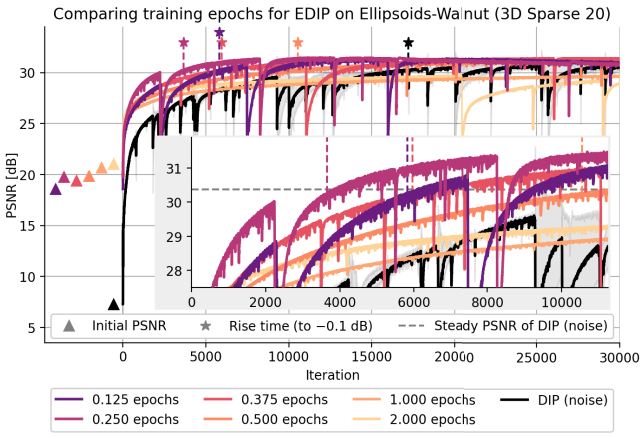

Fig. 13. The optimization of EDIP using parameters from different checkpoints for EDIP (FBP) on Walnut `3D Sparse 20`. The symbols ⋆ and ▲ denote initial PSNR and rise time, respectively, and the horizontal dashed line indicates the steady PSNR of DIP (noise).

directly computing φ'_{θ_0} is intractable. We approximate the first ℓ singular vectors of $F'(\theta_0)$ via randomized singular value decomposition (rSVD) [61], [62], and proceed in two steps (cf. Algorithm 1):

Stage #1: Randomized Range Finder. To construct a subspace capturing most of the action of $F'(\theta_0)$, we draw a Gaussian random matrix $\Omega \in \mathbb{R}^{p\times\ell}$ and form $\bar{F} = F'(\theta_0)\Omega \in \mathbb{R}^{m\times\ell}$. To avoid the direct evaluation of φ'_{θ_0}, for any column ω of Ω, we use a finite difference approximation: $\varphi'_{\theta_0}\omega = (\varphi_{\theta_0+\epsilon\omega} - \varphi_{\theta_0-\epsilon\omega})/(2\epsilon)$, where $\epsilon > 0$ is a small constant. Then we find an orthonormal matrix $Q \in \mathbb{R}^{m\times\ell}$ for the range of $\bar{F}$, using the standard QR factorization [61], [62].

Stage #2: Direct SVD. Next we construct a low-rank matrix $B = Q^\top F'(\theta_0) \in \mathbb{R}^{\ell\times p}$, or equivalently, $B^\top = F'(\theta_0)^\top Q$, which can be computed via backpropagation, and then approximate the singular values and the right singular vectors of $F'(\theta_0)$ by that of $B \approx U\Sigma V^\top$ (with the last few discarded as oversampling: default choice 5). Since the size of $B \in \mathbb{R}^{\ell\times p}$ is much smaller than that of $F'(\theta_0)$, a direct SVD computation is indeed feasible.

Algorithm 1 rSVD for Linearized Forward Map

Require: the Jacobian matrix $F'(\theta_0)$, the target rank κ, and oversampling parameter o

1: Draw a $p\times(\ell = \kappa+o)$ Gaussian random matrix $\Omega = (\omega_{ij})$
2: Form $\bar{F} = F'(\theta_0)\Omega$
3: Construct an orthonormal basis Q of $\mathrm{range}(\bar{F})$ using QR decomposition
4: Form the matrix $B = Q^\top F'(\theta_0)$
5: Compute the SVD of $B = W\tilde{\Sigma}_\ell\tilde{V}_\ell$
6: Return $\tilde{\Sigma}_\kappa, \tilde{V}_\kappa$

In the analysis, we use the 995 leading singular values and the corresponding right singular vectors, which are used to represent the parameters. We investigate EDIP and DIP, both receiving the FBP as the input, and respectively approximate the singular vectors of the Jacobian, evaluated at three checkpoints during the fine-tuning stage (θ^{init}, $\theta^{[100]}$, θ^{conv}). Fig. 14 summarizes our empirical findings, showing the right singular values component-wise plots and Hoyer measure of sparsity [63], [64]. Hoyer measure takes a value 0 if the vector is dense (i.e. all components are equal and non-zero) and 1 if it is 1-sparse. The histogram is computed for the two sets of singular vectors, i.e., $\{v_1, \ldots, v_{20}\}$ and $\{v_{976}, \ldots, v_{995}\}$, separately, in order to examine the behavior at the different frequency bands. For DIP, the singular vectors are equally distributed throughout the parameter space (at θ^{init}) and across different singular values. During the fine-tuning stage, we observe a "relevance shift" towards the decoder's parameters (at $\theta^{[100]}$ and at θ^{conv}, respectively), which is attributed to the fact that the heavy-lifting of representing the target image is actually done by the decoder. This is also consistent with our experimental findings: EDIP-FE shows very similar reconstruction properties to EDIP. For EDIP, pretraining enforces a hierarchical structure

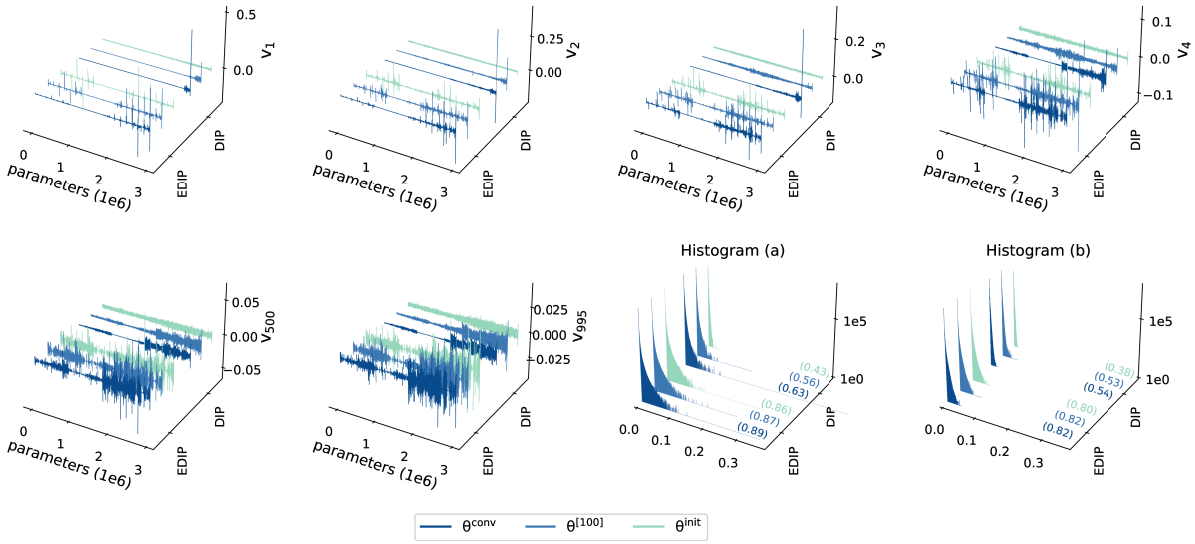

Fig. 14. The evolution of right singular vectors of the linearized forward map (i.e., the Jacobian) w.r.t. the network parameters θ for EDIP (FBP) versus DIP (FBP) on Lotus `Sparse 20` dataset. The parameters are ordered like they occur in the network, i.e. lower positions on the parameters axis refer to the encoder while higher positions refer to the decoder. (a) and (b) show mean histograms for the right singular vectors $v_1, \ldots, v_{20}$ and $v_{976}, \ldots, v_{995}$, which represent the low-frequency and high-frequency bands of the singular vectors, respectively; the numbers in brackets denote Hoyer measure of sparsity [63], [64].

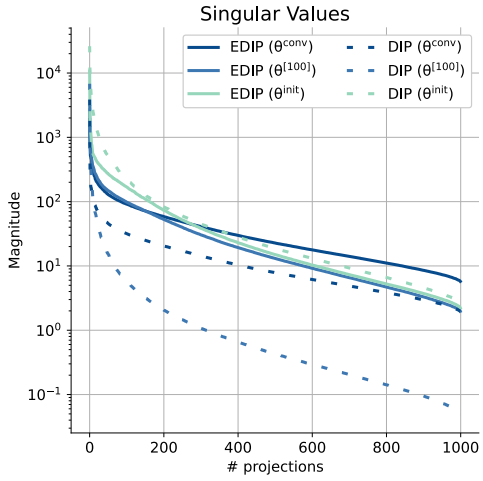

Fig. 15. The singular values of the linearized forward map (i.e., the Jacobian) w.r.t. the network parameters θ, at θ^{conv} and θ^{init} for EDIP (FBP) and DIP (noise) on Lotus `Sparse 20` data.

(i.e. a relevance shift towards the decoder's parameters), and again sparsity is clearly observed after pretraining. Pretraining strongly promotes sparsity in the basis of the parameter space, which is further promoted in the fine-tuning stage. This is observed in both low and high frequency bands. It is worth noting that even though individual singular vectors exhibit sparsity, the parameter vector θ does not necessary exhibit a very high level of sparsity, since the linear combination might spoil it. The emerging sparsity during pretraining may facilitate pruning the network, which however is still to be systematically explored.

Interestingly, pretraining also induces a shift in the singular values spectrum, and the overall behavior does not vary much during adaptation, cf. Fig. 15. In contrast, for DIP, the shift is quite dramatic in terms of the magnitude, as well as the number of singular values larger than a given threshold. This may offer an explanation to the very different dynamics of the optimization scheme for the pretrained model and the model trained from scratch: in the linearized regime, the singular value spectrum essentially determines the dynamics of gradient type algorithms (along with the learning rate), and the dramatic shift of the singular value spectrum of the DIP Jacobian may have contributed to the undesirable unsteady convergence behavior of DIP and indicates the necessity of carefully tuning the learning rate schedule in order to achieve a stable convergence behavior.

VIII. Conclusions

Our work advances unsupervised deep learning-based tomographic reconstruction. We develop a two-stage learning paradigm for accelerating DIP in image reconstruction. It consists of a supervised pretraining stage on a simulated dataset to educate DIP and then a fine-tuning stage which adapts the network parameters to a single test image. The extensive experimental evaluation clearly shows that pretraining on simulated data can significantly speed up, and stabilize DIP reconstruction for 2D / 3D real-measured sparse-view μCT. The empirical study also indicates that the pretraining stage can facilitate learning a suitable feature representation, and that adapting only the decoder's parameters during the fine-tuning stage is sufficient to ensure good reconstruction accuracy. The novel spectral analysis of the linearized model indicates a strong correlation of the sparsity pattern with the pretraining, and a drastically different shift of the singular values spectrum for the standard DIP and the educated version.

There are several avenues for further research. First, there are other techniques for learning a good initialization for neural networks, e.g., model-agnostic meta-learning (MAML)

[53] and adversarial pretraining [51], [52]. These strategies are also promising, but their full potentials are yet to be explored within the context of DIP reconstruction. In the spirit of ANIL (Almost No Inner Loop) [54], we would suggest using a variant that simplifies the inner loop optimization so as to improve the scalability of MAML. Second, given the emerging sparsity pattern in singular vectors, it is natural to ask whether one can exploit for even faster adaptation, e.g., via pruning or optimizing in low-dimensional subspaces. Third, the proposal utilizes the specific forward operator in the pretraining stage, and hence the pretrained neural network is specialized, where specialization to the target task is believed to be helpful. However, addressing multiple settings (e.g., different imaging modalities and multiple image classes) simultaneously is of course of interest.

ACKNOWLEDGMENT

R.B. was supported by the i4health PhD studentship (UK EPSRC EP/S021930/1), and by The Alan Turing Institute under the UK EPSRC grant EP/N510129/1. J.L., M.S., and A.D. were funded by the German Research Foundation (DFG; GRK 2224/1). J.L. and M.S. additionally acknowledge support from the DELETO project funded by the Federal Ministry of Education and Research (BMBF, project number 05M20LBB). A.D. further acknowledges support from the Klaus Tschira Stiftung via the project MALDISTAR (project number 00.010.2019). A.H. acknowledges funding from the Academy of Finland projects 338408, 336796, 334817. P.M. was supported by the Sino-German Center for Research Promotion (CDZ) via the Mobility Programme 2021: Inverse Problems – Theories, Methods and Implementations (IP–TMI). The research of B.J. is supported by UK EPSRC grants EP/T000864/1 and EP/V026259/1.

REFERENCES

[1] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of Inverse Problems*. Kluwer, Dordrecht, 1996. 1, 2

[2] O. Scherzer, M. Grasmair, H. Grossauer, M. Haltmeier, and F. Lenzen, *Variational Methods in Imaging*. New York, NY: Springer, 2009. 1, 2

[3] K. Ito and B. Jin, *Inverse Problems: Tikhonov Theory and Algorithms*. World Scientific, Hackensack, NJ, 2015. 1, 2

[4] G. Ongie, A. Jalal, R. G. Baraniuk, C. A. Metzler, A. G. Dimakis, and R. Willett, "Deep learning techniques for inverse problems in imaging," *IEEE J. Sel. Areas Inform. Theory*, vol. 1, no. 1, pp. 39–56, 2020. 1

[5] S. Arridge, P. Maass, O. Öktem, and C.-B. Schönlieb, "Solving inverse problems using data-driven models," *Acta Numer.*, vol. 28, pp. 1–174, 2019. 1

[6] D. O. Baguer, J. Leuschner, and M. Schmidt, "Computed tomography reconstruction using deep image prior and learned reconstruction methods," *Inverse Problems*, vol. 36, no. 9, p. 094004, 2020. 1, 2, 3, 4, 13

[7] D. Gilton, G. Ongie, and R. Willett, "Model adaptation for inverse problems in imaging," *IEEE Trans. Comput. Imag.*, vol. 7, pp. 661–674, 2021. 1, 2

[8] R. Barbano, Z. Kereta, A. Hauptmann, S. R. Arridge, and B. Jin, "Unsupervised knowledge-transfer for learned image reconstruction," *Inverse Problems*, vol. 38, no. 10, p. 104004, 2022. 1

[9] S. Lunz, A. Hauptmann, T. Tarvainen, C.-B. Schönlieb, and S. Arridge, "On learned operator correction in inverse problems," *SIAM J. Imag. Sci.*, vol. 14, no. 1, pp. 92–127, 2021. 1

[10] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *CVPR*, 2018, pp. 9446–9454. 1, 2, 3

[11] T. Knopp and M. Grosser, "Warmstart approach for accelerating deep image prior reconstruction in dynamic tomography," *Proceedings of Machine Learning Research, Medical Imaging with Deep Learning 2022, 13 pp.*, 2022. 1, 2

[12] M. Z. Darestani and R. Heckel, "Accelerated MRI with un-trained neural networks," *IEEE Trans. Comput. Imag.*, vol. 7, pp. 724–733, 2021. 1

[13] K. Gong, C. Catana, J. Qi, and Q. Li, "PET image reconstruction using deep image prior," *IEEE Trans. Med. Imag.*, vol. 38, no. 7, pp. 1655–1665, 2019. 1

[14] J. Cui, K. Gong, N. Guo, C. Wu, K. Kim, H. Liu, and Q. Li, "Populational and individual information based PET image denoising using conditional unsupervised learning," *Phys. Med. & Biol.*, vol. 66, no. 15, p. 155001, 2021. 1

[15] J. Cui, K. Gong, N. Guo, C. Wu, X. Meng, K. Kim, K. Zheng, Z. Wu, L. Fu, B. Xu *et al.*, "PET image denoising using unsupervised deep learning," *Eur. J. Nuclear Med. Mol. Imag.*, vol. 46, no. 13, pp. 2780–2789, 2019. 1

[16] S. Barutcu, D. Gürsoy, and A. K. Katsaggelos, "Compressive ptychography using deep image and generative priors," *Preprint, arXiv:2205.02397*, 2022. 1

[17] V. Monga, Y. Li, and Y. C. Eldar, "Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing," *IEEE Signal Proc. Mag.*, vol. 38, no. 2, pp. 18–44, 2021. 1

[18] A. Hauptmann, F. Lucka, M. Betcke, N. Huynh, J. Adler, B. Cox, P. Beard, S. Ourselin, and S. Arridge, "Model-based learning for accelerated, limited-view 3-d photoacoustic tomography," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1382–1393, 2018. 1

[19] J. Leuschner, M. Schmidt, P. S. Ganguly, V. Andriiashen, S. B. Coban, A. Denker, D. Bauer, A. Hadjifaradji, K. J. Batenburg, P. Maass, and M. van Eijnatten, "Quantitative comparison of deep learning-based image reconstruction methods for low-dose and sparse-angle ct applications," *Journal of Imaging*, vol. 7, no. 3, 2021. 1

[20] S. Azizi, B. Mustafa, F. Ryan, Z. Beaver, J. Freyberg, J. Deaton, A. Loh, A. Karthikesalingam, S. Kornblith, T. Chen *et al.*, "Big self-supervised models advance medical image classification," *Preprint, arXiv:2101.05224*, 2021. 1

[21] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," *Preprint, arXiv:1801.06146*, 2018. 1

[22] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition," in *ICML*, 2014, pp. 647–655. 1

[23] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*. Springer, 2015, pp. 234–241. 2

[24] Y. Jo, S. Y. Chun, and J. Choi, "Rethinking deep image prior for denoising," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 5087–5096. 2

[25] H. Wang, T. Li, Z. Zhuang, T. Chen, H. Liang, and J. Sun, "Early stopping for deep image prior," *Preprint, arXiv:2112.06074*, 2021. 2

[26] R. Heckel and P. Hand, "Deep decoder: Concise image representations from untrained non-convolutional networks," in *ICLR*, 2019. 2

[27] S. Dittmer, T. Kluth, P. Maass, and D. Otero Baguer, "Regularization by architecture: A deep prior approach for inverse problems," *J. Math. Imag. Vision*, vol. 62, no. 3, pp. 456–470, 2020. 2

[28] Z. Cheng, M. Gadelha, S. Maji, and D. Sheldon, "A Bayesian perspective on the deep image prior," in *CVPR*, 2019, pp. 5443–5451. 2

[29] J. Liu, Y. Sun, X. Xu, and U. S. Kamilov, "Image restoration using total variation regularized deep image prior," in *ICASSP*, 2019. 2, 3

[30] G. Mataev, P. Milanfar, and M. Elad, "DeepRED: Deep image prior powered by RED," in *ICCV Workshops*, Oct 2019. 2

[31] R. Heckel and M. Soltanolkotabi, "Compressive sensing with untrained neural networks: Gradient descent finds the smoothest approximation," in *Proceedings of the 37th International Conference on Machine Learning, PMLR 119*, 2020, pp. 4149–4158. 2

[32] C. Arndt, "Regularization theory of the analytic deep prior approach," *Inverse Problems*, vol. 38, no. 11, p. 115005, 2022. 2

[33] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *CVPR*, 2016, pp. 779–788. 2

[34] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," in *Advances in Neural Inf. Process. Syst.*, 2015, pp. 91–99. 2

[35] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *CVPR*, 2015, pp. 3431–3440. 2

[36] K. He, R. Girshick, and P. Dollar, "Rethinking imagenet pre-training," in *ICCV*, 2019, pp. 4918–4927. 2

[37] M. Raghu, C. Zhang, J. Kleinberg, and S. Bengio, "Transfusion: Understanding transfer learning for medical imaging," in *Proceedings of the 33rd Neural Information Processing Systems*, 2019, pp. 3347–3357. 2

[38] Y. Han, J. Yoo, H. H. Kim, H. J. Shin, K. Sung, and J. C. Ye, "Deep learning with domain adaptation for accelerated projection-reconstruction MR," *Magn. Reson. Med.*, vol. 80, no. 3, pp. 1189–1205, 2018. 2

[39] S. U. H. Dar, M. Özbey, A. Çatlı, and B. T. Çukur, "A transfer-learning approach for accelerated mri using deep neural networks," *Magn Reson Med*, vol. 84, no. 2, pp. 663–685, 2020. 2

[40] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd ICLR*, Y. Bengio and Y. LeCun, Eds., 2015. 3

[41] X. Pan, E. Y. Sidky, and M. Vannier, "Why do commercial CT scanner still employ traditional, filtered back-projection for image reconstruction?" *Inverse Problems*, vol. 25, no. 12, pp. 123 009, 36, 2009. 3

[42] K. H. Jin, M. T. McCann, E. Froustey, and M. Unser, "Deep convolutional neural network for inverse problems in imaging," *IEEE Trans. Image Proc.*, vol. 26, no. 9, pp. 4509–4522, 2017. 3

[43] J. Adler, H. Kohr, A. Ringh, J. Moosmann, S. Banert, M. J. Ehrhardt, G. R. Lee, K. Niinimäki, B. Gris, O. Verdier, J. Karlsson, G. Zickert, W. J. Palenstijn, O. Öktem, C. Chen, H. A. Loarca, and M. Lohmann, "Operator Discretization Library (ODL)," 2018, *Zenodo*. 3

[44] T. A. Bubba, A. Hauptmann, S. Huotari, J. Rimpeläinen, and S. Siltanen, "Tomographic X-ray data of a lotus root filled with attenuating objects," *Preprint, arXiv:1609.07299*, 2016. 4

[45] H. Der Sarkissian, F. Lucka, M. van Eijnatten, G. Colacicco, S. B. Coban, and K. J. Batenburg, "Cone-Beam X-Ray CT Data Collection Designed for Machine Learning: Samples 1-8," 2019, *Zenodo*. [Online]. Available: https://doi.org/10.5281/zenodo.2686726 4

[46] L. A. Feldkamp, L. C. Davis, and J. W. Kress, "Practical cone-beam algorithm," *J. Opt. Soc. Am. A*, vol. 1, no. 6, pp. 612–619, 1984. 4

[47] A. Hendriksen, D. Schut, W. J. Palenstijn, N. Viganò, J. Kim, D. Pelt, T. van Leeuwen, and K. J. Batenburg, "Tomosipo: Fast, flexible, and convenient 3D tomography for complex scanning geometries in Python," *Optics Expr.*, Oct 2021. 4

[48] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, and O. Ronneberger, "3d eu-net: learning dense volumetric segmentation from sparse annotation," in *MICCAI*. Springer, 2016, pp. 424–432. 4

[49] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Proc.*, vol. 13, no. 4, pp. 600–612, 2004. 4

[50] L. A. Shepp and B. F. Logan, "The Fourier reconstruction of a head section," *IEEE Trans. Nuclear Sci.*, vol. 21, no. 3, pp. 21–43, 1974. 5

[51] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2015. [Online]. Available: http://arxiv.org/abs/1412.6572 7, 11

[52] M. Yi, L. Hou, J. Sun, L. Shang, X. Jiang, Q. Liu, and Z. Ma, "Improved ood generalization via adversarial training and pretraing," in *International Conference on Machine Learning*. PMLR, 2021, pp. 11 987–11 997. 7, 11

[53] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International Conference on Machine Learning (ICML)*, 2017, pp. 1126–1135. 7, 11

[54] A. Raghu, M. Raghu, S. Bengio, and O. Vinyals, "Rapid learning or feature reuse? towards understanding the effectiveness of maml," in *International Conference on Learning Representations*, 2019. 7, 11

[55] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," in *2017 IEEE Symposium on Security and Privacy (SP)*, 2017, pp. 39–57. 7

[56] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, "Towards deep learning models resistant to adversarial attacks," *International Conference on Learning Representations*, 2018. 7

[57] R. Barbano, J. Leuschner, J. Antorán, B. Jin, and J. M. Hernández-Lobato, "Bayesian experimental design for computed tomography with the linearised deep image prior," *Preprint, arXiv:2207.05714*, 2022, presented at ICML Workshop on Adaptive Experimental Design and Active Learning in the Real World (ReALML) 2022, July 22, Baltimore, MD, USA. 7

[58] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, 2010. 7

[59] V. Antun, F. Renna, C. Poon, B. Adcock, and A. C. Hansen, "On instabilities of deep learning in image reconstruction and the potential costs of AI," *Proc. Nat. Acad. Sci.*, vol. 117, no. 48, pp. 30 088–95, 2020. 8

[60] B. Neyshabur, H. Sedghi, and C. Zhang, "What is being transferred in transfer learning?" in *Advances in Neural Information Processing Systems*, vol. 33, 2020. 8

[61] N. Halko, P.-G. Martinsson, and J. A. Tropp, "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions," *SIAM Rev.*, vol. 53, no. 2, pp. 217–288, 2011. 9

[62] J. A. Tropp, "Randomized algorithms for matrix computations," Caltech CMS Lecture Notes 2019-01, Pasadena, July 2019. 9

[63] P. O. Hoyer, "Non-negative matrix factorization with sparseness constraints," *J. Mach. Learn. Res.*, vol. 5, pp. 1457–1469, 2004. 9, 10

[64] N. Hurley and S. Rickard, "Comparing measures of sparsity," *IEEE Trans. Inform. Theory*, vol. 55, no. 10, pp. 4723–4741, 2009. 9, 10

SUPPLEMENTARY MATERIAL A
μCT MEASUREMENT DATA

A. Cone-Beam Geometry

On the Lotus root, we employ the sparse matrix provided with the dataset. For the 2D Walnut setting, a sparse matrix resembling the 2D cone-beam projection is constructed from the ASTRA geometry, by selecting a single volume slice, and a suitable subset of the 3D cone-beam projection lines. This is a non-standard 2D fan-beam setting: (i) the rotation axis is slightly tilted; (ii) the voxels / pixels are weighted according to the 3D projections, which differs from the 2D projection weighting. Specifically, in the integration of the beams for each detector "pixel", the contributing area / interval is spreading in two vs. one dimension(s) with increasing distance from the source, so the beam density decreases antiproportionally to the squared distance vs. antiproportionally to the distance. For the 3D Walnut settings, ASTRA's direct projection routines are employed via tomosipo. The backward gradients are approximated by back-projection. The geometry definition has been adapted to match the sub-sampling applied to the volume and the measurements.

B. X-ray Walnut Details

From the collection of 42 Walnuts, we consider measurements of Walnut 1 taken with source position (or orbit) 2. The slice with offset $+3\,\mathrm{px}$ from the middle slice (i.e. zero-based index 253) is selected for the 2D reconstruction task. A subset of projection values is determined from the provided ASTRA geometry by computing the 3D forward projection of a mask, containing ones for the selected 2D slice and zeros for all other voxels. We choose one single detector row per column and angle with maximum intensity. A sparse matrix representing the forward projection is constructed from the ASTRA forward projection routine for each unit vector, for which the transposed matrix gives an exact adjoint of the Jacobian, used in computing the gradient of (1). The more efficient ASTRA back-projection routine is not directly applicable due to the pseudo-2D geometry: some of the excluded detector rows close to the selected ones contribute to the selected 2D slice in the back-projection. Another workaround (without matrix assembly) is to copy the measurement values from the selected rows to the neighboring rows (a.k.a. edge-mode padding); we use this to compute approximate FDK reconstructions. For computing the gradient of the data fitting term in (1), using the padding followed by the back-projection via ASTRA leads to degraded results, so we use the sparse matrix multiplication instead, which yields accurate gradients.

The implementation and the sparse matrix are available at https://educateddip.github.io/docs.educated_deep_image_prior/.

SUPPLEMENTARY MATERIAL B
METHODOLOGY

A. 2D Network architecture

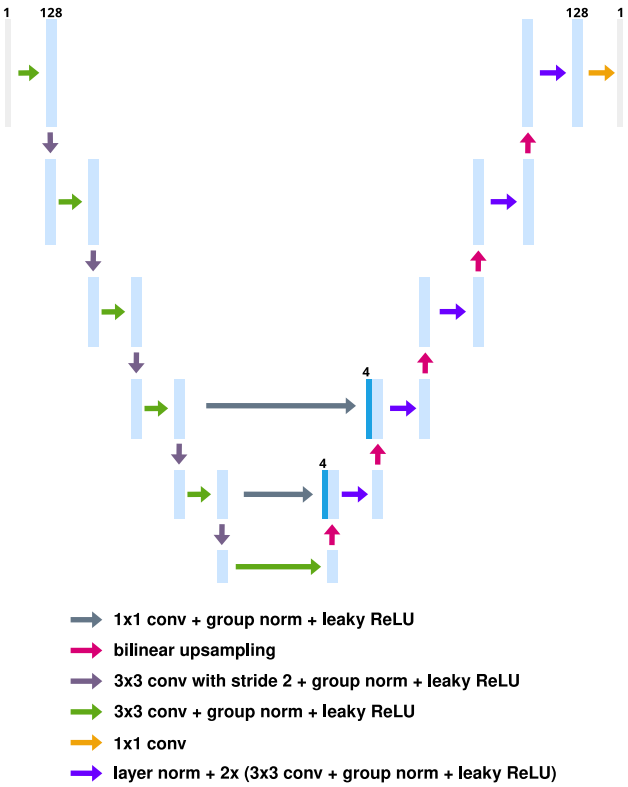

Fig. 16. The architecture of the U-Net used for the 2D experiments. Each light-blue bar corresponds to a multi-channel feature map. Arrows denote the different operations. The number of channels is set to 128 at every scale.

Figure 16 shows the network architecture used. We adopted the architecture proposed by [6], with the only difference being that we replace batch-normalization layers with group-normalization layers.

See also Figure 4 in the main text showing the U-Net architecture used for the 3D experiments.

B. The Loss

Our DIP implementation uses the loss function

$$l'_t(\theta) := \tfrac{1}{m}\|A\varphi_\theta(z) - y_\delta\|_2^2 + \gamma' \,\mathrm{TV}(\varphi_\theta(z)),$$

with the anisotropic total variation penalty $\mathrm{TV}(x) = \|\nabla_h x\|_1 + \|\nabla_v x\|_1$, where m is the number of detector pixels (length of y_δ) and ∇_h and ∇_v are the discrete difference operators in the horizontal and vertical directions, respectively.

C. Hyperparameter Search

For each setting, suitable hyperparameters for DIP (noise) are selected by grid search. While the learning rate $1\mathrm{e}{-4}$ is (near) optimal in all cases, the TV-regularization parameter γ' varies both with the μCT geometry and between validation data (i.e. Shepp-Logan phantom, simulated data) and test data (i.e. Lotus or Walnut, real data).

TABLE V
HYPERPARAMETERS FOR (E)DIP ON VALIDATION AND TEST DATA.

Validation	Learn. rate	γ'	Iters.
Lotus `Sparse 20`	1e−4	4e−5	37 500
Lotus `Limited 45`	1e−4	1e−6	15 000
↪ EDIP (FBP)	1e−4	4e−6	10 000
Walnut `Sparse 120`	1e−4	2e−7	50 000
Test	**Learn. rate**	γ'	**Iters.**
Lotus `Sparse 20`	1e−4	1e−4	10 000
Lotus `Limited 45`	1e−4	6.5e−5	10 000
Walnut `Sparse 120`	1e−4	2e−7	30 000
↪ EDIP[-FE] (noise) pretrained on ellipses	5e−4 to 1e−4	2e−7	30 000
Walnut `3D Sparse 20`	1e−4	1e−1	30 000
Walnut `3D Sparse 60`	5e−5	1e−1	60 000

The hyperparameters used for DIP and EDIP are listed in Table V. The parameters are fine-tuned on DIP (noise), except for the override values specified in the rows starting with "↪". For only two cases, we observe the hyperparameters that are optimal for DIP (noise) to be severely sub-optimal for EDIP. For instance, no speed-up is observed for EDIP (noise), applied to the Walnut `Sparse 120`, after pretraining on the ellipses dataset, if the default learning rate 1e−4 is used; while a higher learning rate leads to an unstable optimization. A "warm-up" learning rate scheduling with an initial learning rate of 5e−4, which is linearly decreased to 1e−4 over the first 5k iterations reveals a substantial speed-up. We use the same learning rate scheduling with DIP (noise), but fail to observe any improvement. Similarly, we observe that validating on the Shepp-Logan phantom for the Lotus `Limited 45` setting requires the regularization parameter γ' to be increased to 4e−6 (instead of 1e−6) for EDIP (FBP) to converge.

TABLE VI
HYPERPARAMETERS FOR LOTUS GOLD-STANDARD REFERENCE RECONSTRUCTION.

Reference	Learn. rate	γ'	Iters.
Lotus (full 120) TV	1e-3	5e−5	1000

TABLE VII
HYPERPARAMETERS FOR TV BASELINES ON TEST DATA.

Test	Learn. rate	γ'	Iters.
Lotus `Sparse 20` TV	5e−4	1e−4	5000
Lotus `Limited 45` TV	5e−4	4e−5	5000
Walnut `Sparse 120` TV	5e−4	4e−7	10 000
Walnut `3D Sparse 20` TV	5e−4	2e−1	5000
Walnut `3D Sparse 60` TV	5e−4	1e−1	5000

SUPPLEMENTARY MATERIAL C
EXTENDED EXPERIMENTAL RESULTS

Here we report additional details about the experiments.

A. The Lotus (Continued)

We also include a limited-view setting, named Lotus `Limited 45`: 45 angles, range $[0, 135°)$ in steps of 3°. Fig. 17 shows exemplary reconstructions on the test-fold of the synthetic datasets used for pretraining, for both `Sparse 20` and `Limited 45`. The FBP suffers severe streak artifacts, but the trained U-Net can recover the shapes well.

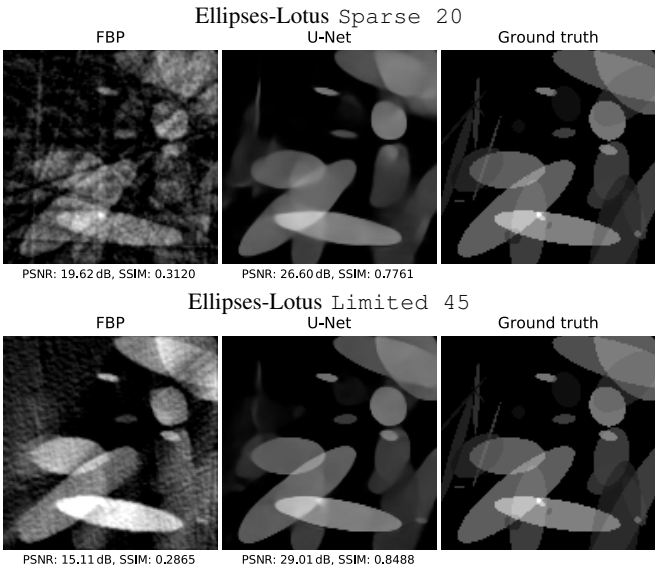

Fig. 17. Exemplary reconstructions from the synthetic training datasets for Lotus `Sparse 20` and `Limited 45`.

The PSNR convergence of EDIP on Lotus root for the `Limited 45` setting is shown in Fig. 18; the reconstructions are reported in Fig. 19. These numerical results indicate analogous conclusions as for the case of `Sparse 20`.

Table VIII reports overall tabular results for Lotus `Sparse 20` and Lotus `Limited 45`. Rise time is defined to be the minimal number of iterations after which the PSNR reaches steady PSNR of DIP (noise) minus 0.1 dB. Both maximum PSNR and steady PSNR are computed using the iteration-wise median PSNR history over the 5 repeated runs (varying the random seed). For steady PSNR, the median value of the median PSNR history over the last 5k iterations is considered. The convergence of TV is observed to be very stable, and we report the final PSNR. Initial PSNR is the mean value over the 5 repeated runs.

It is observed that pretraining can substantially accelerate and stabilize the convergence of DIP. The acceleration factor is more substantial, when considering the FBP as input. The maximum PSNR (Max. PSNR) and steady PSNR suggest that pretraining also improves the reconstruction quality. The performance of EDIP-FE is largely comparable to EDIP.

Fig. 20 shows the convergence of the loss in (1) and of the PSNR, where the PSNR is computed using the network output with minimum loss reached until the current iteration. Using the minimum loss output is a practical way to overcome the instability of DIP optimization, clearly observed in the plots with the raw data in the main analysis. Pretraining greatly accelerates and stabilizes subsequent unsupervised training of EDIP, when compared to the standard DIP. This indicates a more favorable optimization landscape of EDIP / EDIP-FE than that of DIP. A stable convergence in practice is important for designing stopping rules for DIP / EDIP.

TABLE VIII
QUANTITATIVE EVALUATION FOR LOTUS SPARSE 20 AND LOTUS LIMITED 45 WITH EDIP BEING PRETRAINED ON ELLIPSES DATA.

Ellipses-Lotus	Sparse 20				Limited 45			
	Rise time	(Max PSNR; iters)	Steady PSNR	Init PSNR	Rise time	(Max PSNR; iters)	Steady PSNR	Init PSNR
DIP (noise)	3848	(31.17; 8846)	31.10	11.17	5470	(29.85; 9690)	29.69	11.17
DIP (FBP)	3622	(31.25; 8813)	31.17	11.33	5419	(29.84; 8898)	29.69	11.32
DIP-FE (noise)	6118	(31.10; 9818)	31.00	11.17	5142	(29.82; 8884)	29.69	11.17
DIP-FE (FBP)	4516	(31.19; 7677)	31.13	11.33	5056	(29.83; 9891)	29.67	11.32
EDIP (FBP)	195	(31.65; 981)	31.21	27.04	524	(29.83; 2734)	29.68	27.55
EDIP (noise)	723	(31.53; 3548)	31.39	14.28	682	(29.94; 4445)	29.80	14.34
EDIP-FE (FBP)	226	(31.59; 1421)	31.26	27.04	245	(29.85; 5533)	29.72	27.55
EDIP-FE (noise)	1414	(31.46; 4278)	31.39	14.28	1279	(29.95; 7095)	29.86	14.34
TV	–	–	30.73	–	–	–	29.62	–

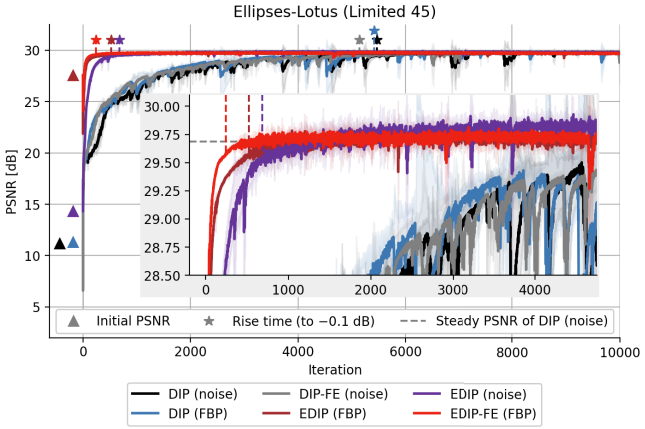

Fig. 18. The optimization of EDIP versus DIP on Lotus Limited 45. All traces are the mean PSNR of 5 runs (varying the seed). The notations ▲ and ⋆ denote the initial PSNR and rise time, respectively, and the horizontal dashed line indicates steady PSNR of DIP (noise).

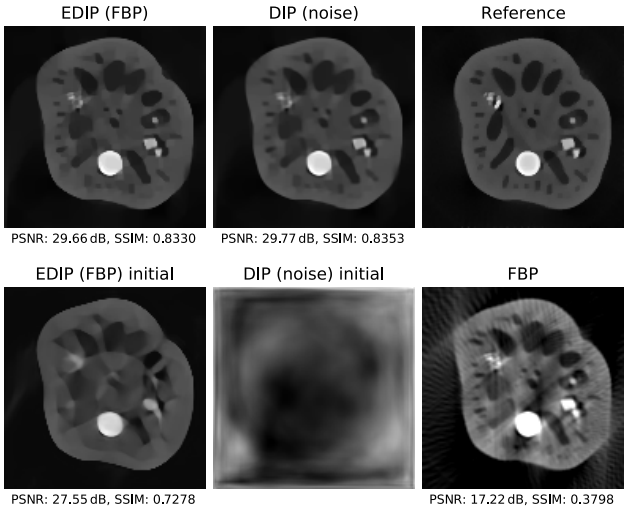

Fig. 19. Lotus reconstruction of EDIP versus DIP on Lotus Limited 45 data. From the 5 runs (varying the seed), the one with the (closest to) median PSNR was selected for each method. The reported reconstructions are the best reconstruction (i.e. reconstruction at the minimum loss value).

B. The Walnut (Continued)

The quantitative results in Table IX validate our findings on the Lotus root. See also Figs. 21 and 22–23 for convergence behavior and exemplary reconstructions.

TABLE IX
QUANTITATIVE EVALUATION FOR WALNUT SPARSE 120 WITH EDIP BEING PRETRAINED ON ELLIPSES DATA. FOR THE EXPERIMENTS MARKED WITH "*" A HIGHER INITIAL LEARNING RATE WAS USED (SEE TABLE V).

Ellipses-Walnut Sparse 120	Rise time	(Max PSNR; iters)	Steady PSNR	Init PSNR
DIP (noise)	20 373	(34.02; 25 357)	33.87	6.88
DIP (FBP)	13 778	(34.07; 28 094)	33.90	6.26
DIP-FE (noise)	14 289	(34.02; 23 573)	33.88	6.88
DIP-FE (FBP)	13 421	(34.19; 23 266)	33.97	6.26
EDIP (FBP)	4496	(33.92; 13 039)	33.56	25.67
EDIP (noise) *	9561	(34.12; 23 352)	33.95	12.22
EDIP-FE (FBP)	4384	(33.91; 12 540)	33.70	25.67
EDIP-FE (noise) *	21 760	(33.89; 29 159)	33.75	12.22
TV	–	–	31.67	–

TABLE X
QUANTITATIVE EVALUATION FOR WALNUT 3D SPARSE 20 AND 3D SPARSE 60 WITH EDIP BEING PRETRAINED ON ELLIPSOIDS DATA. BOTH MAXIMUM PSNR AND STEADY PSNR ARE COMPUTED USING THE ITERATION-WISE MEDIAN PSNR HISTORY OVER 3 REPEATED RUNS (VARYING THE RANDOM SEED). FOR STEADY PSNR, THE MEDIAN VALUE OF THE MEDIAN PSNR HISTORY OVER THE LAST 5K ITERATIONS IS CONSIDERED. THE CONVERGENCE OF TV IS VERY STABLE, AND WE REPORT THE FINAL PSNR. INITIAL PSNR IS THE MEAN VALUE OVER THE 3 REPEATED RUNS. ALL PSNR VALUES ARE IN dB.

Ellipsoids-Walnut	3D Sparse 20				3D Sparse 60			
	Rise time	(Max PSNR; iters)	Steady PSNR	Init PSNR	Rise time	(Max PSNR; iters)	Steady PSNR	Init PSNR
DIP (noise)	17 200	(30.68; 23 477)	30.37	7.29	49 041	(34.05; 58 901)	33.93	7.29
DIP (FBP)	13 016	(31.32; 25 063)	31.19	8.19	27 873	(34.37; 53 731)	34.22	8.62
EDIP (FBP)	3739	(31.48; 10 689)	30.94	19.77	11 247	(34.35; 40 810)	34.18	20.17
EDIP-FE (FBP)	2979	(31.38; 10 749)	30.93	19.77	14 520	(34.33; 45 259)	34.15	20.17
TV	–	–	28.89	–	–	–	33.35	–

SUPPLEMENTARY MATERIAL D
VALIDATING PRETRAINING

Different checkpoints are obtained from multiple pretraining runs (varying the random seed), and by collecting checkpoints along the optimization trajectory from each run. We identify the parameters' configuration to be used at test time from these checkpoints by selecting the one with the best performance on a validation set. To this end, we design a reconstructive task based on the Shepp-Logan phantom, a standard test image created to assess reconstruction algorithms. The phantom is by construction within the ellipses data manifold and shares the same noise distribution of ellipses measurements. The checkpoint leading to the shortest rise time is selected, among those with a steady PSNR that is at most 0.25 dB lower than the maximum reached steady PSNR.

We repeat the pretraining three times (varying the seed) and collect checkpoints after every 20 epochs for Lotus Sparse 20 and Lotus Limited 45, training for a maximum of 100 epochs. We also include the checkpoint for which the model shows minimum validation loss. For Walnut Sparse 120 we pretrain for 20 epochs, and retain only the minimum validation loss checkpoint. Fig. 24 shows the convergence of the pretraining on the ellipses datasets for the Lotus and the Walnut settings, along with the learning rate scheduling.

At the validation stage, each checkpoint is evaluated by performing EDIP fine-tuning on simulated data of the Shepp-Logan phantom. The validation runs for Lotus Sparse 20, Lotus Limited 45, and Walnut Sparse 120 are shown in Figs. 25 and 27, respectively. In the Lotus settings, starting EDIP fine-tuning using checkpoints from a later epoch (e.g. 60, 80, 100) is more beneficial. Nonetheless, even pretraining for fewer epochs (e.g. 20) can already greatly benefit the EDIP fine-tuning, although to a lesser degree. Pretraining considerably ameliorates the quality of the reconstruction of the Shepp-Logan phantom for both Lotus and Walnut settings. Especially for the Lotus Limited 45 setting, it substantially increases the reconstruction quality.

We then investigate whether the selected checkpoints that then are used for the test data — both the Lotus and the Walnut could be considered an out-of-distribution image class — are still optimal as we switch from the simulated measurements of the Shepp-Logan phantom to the real-measured test data. Fig. 26 and Fig. 28 show the PSNR convergence on the test data using different checkpoints. While we observe a different behavior between validation and test data, the validation selects one of the best two checkpoint.

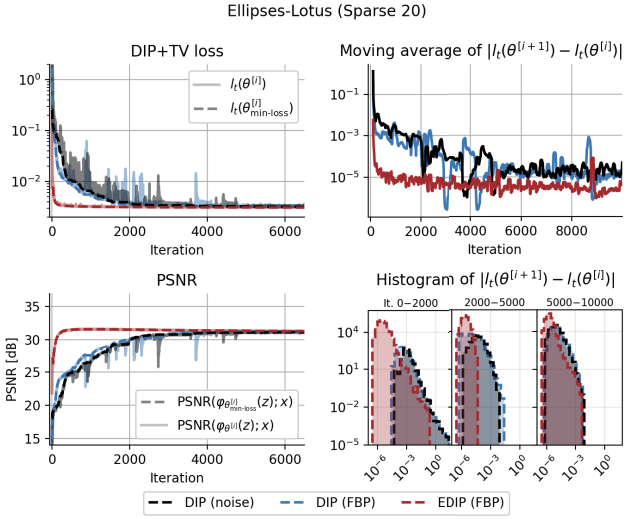

Fig. 20. Min-loss and PSNR computed with the min-loss output for Lotus `Sparse 20` (left). Loss variation (i.e. $|l_t(\theta^{[i+1]}) - l_t(\theta^{[i]})|$) and respective histograms computed over three intervals (right). The moving average uses a window size of 100 iterations.

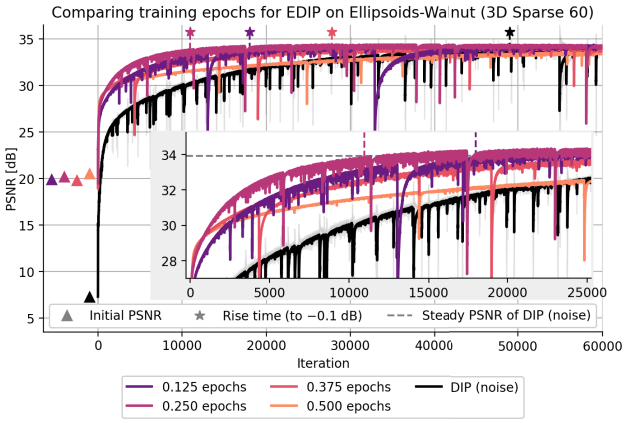

Fig. 21. The optimization of EDIP using different checkpoints for EDIP (FBP) on Walnut `3D Sparse 60` data. The notations ▲ and ⋆ denote the initial PSNR and rise time, respectively, and the horizontal dashed line indicates steady PSNR of DIP (noise).

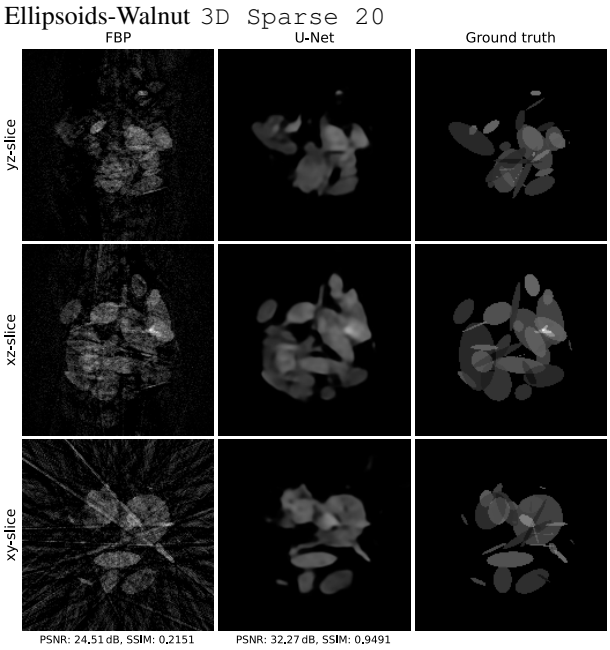

Fig. 22. Exemplary reconstructions from the synthetic training dataset of ellipsoids images for Walnut `3D Sparse 20`.

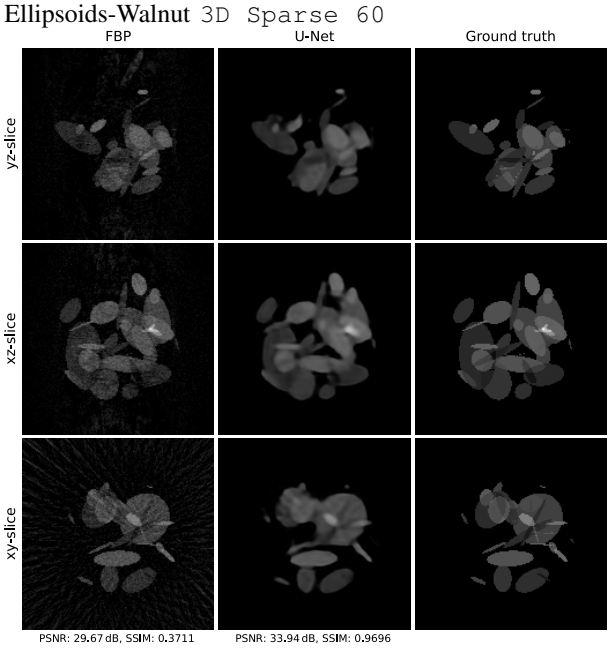

Fig. 23. Exemplary reconstructions from the synthetic training dataset of ellipsoids images for Walnut `3D Sparse 60`.

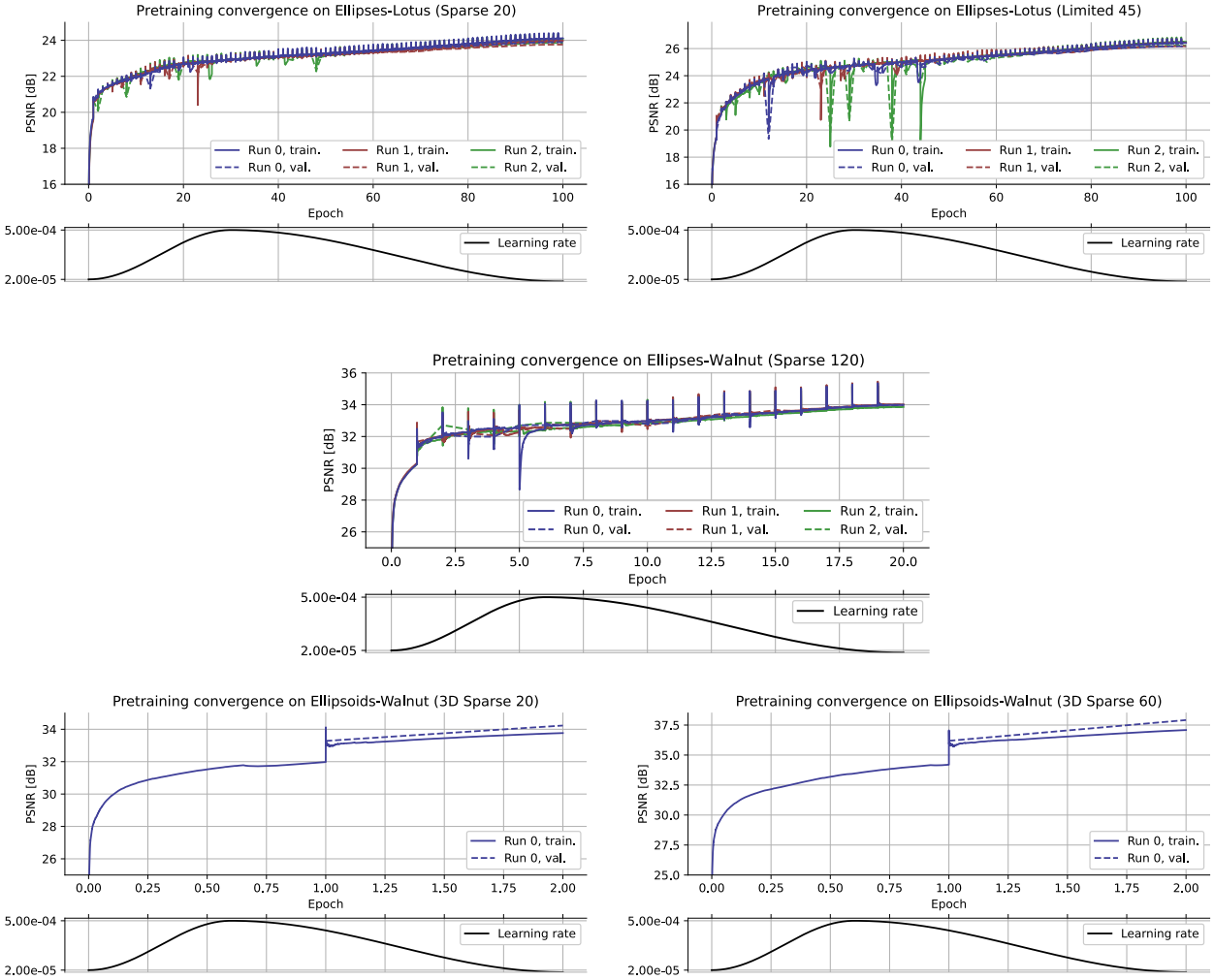

Fig. 24. Pretraining convergence. Solid lines show the running mean of the training loss since the start of the respective epoch; dashed lines show the mean validation loss evaluated after each epoch (on a set of 3200 held-out images).

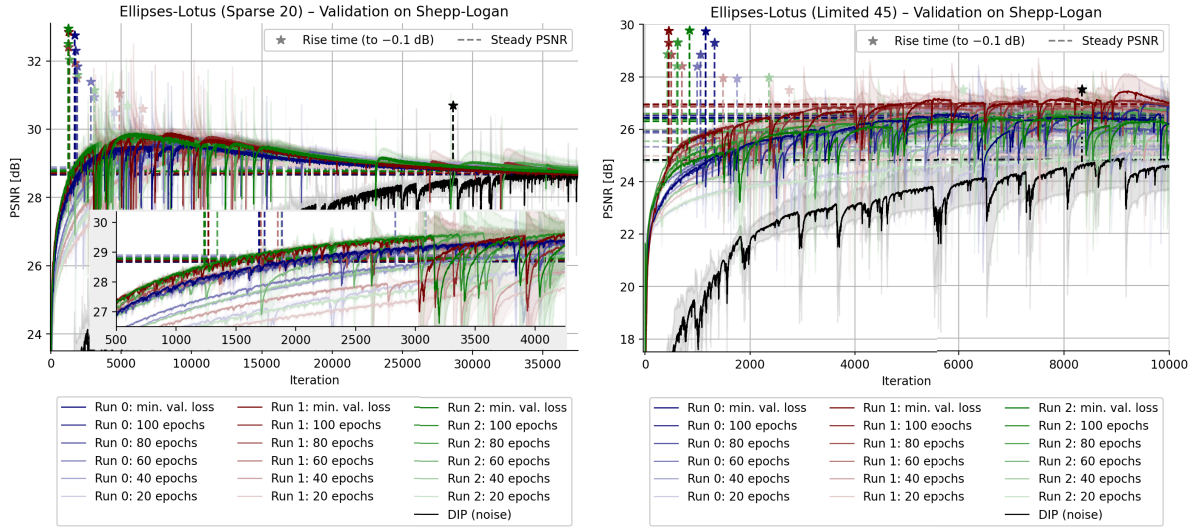

Fig. 25. Validation runs on the Shepp-Logan phantom for selecting the initial EDIP (FBP) model parameters for data in the Lotus `Sparse 20` and `Limited 45` geometry. For `Sparse 20` the model from training run 2 after 100 epochs is selected because it has the shortest rise time (with a sufficiently high steady PSNR), whilst, for `Limited 45` run 1 after 100 epochs is selected.

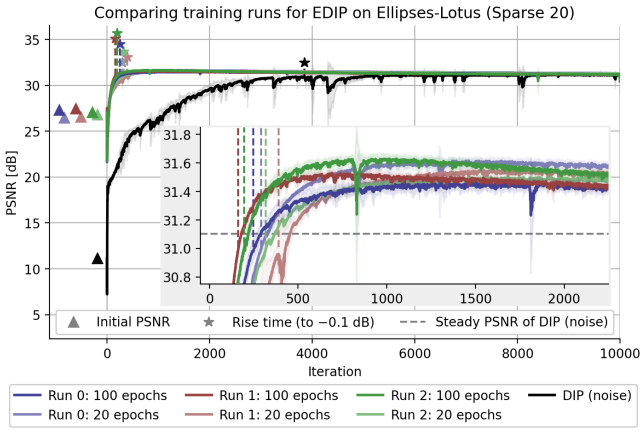

Fig. 26. Optimization of EDIP using different checkpoints considered during validation (see Fig. 25) for EDIP (FBP) on Lotus `Sparse 20` data. The parameters from run 2 after 100 epochs are selected by the validation. The notations ▲ and ⋆ denote the initial PSNR and rise time, respectively, and the horizontal dashed line indicates steady PSNR of DIP (noise).

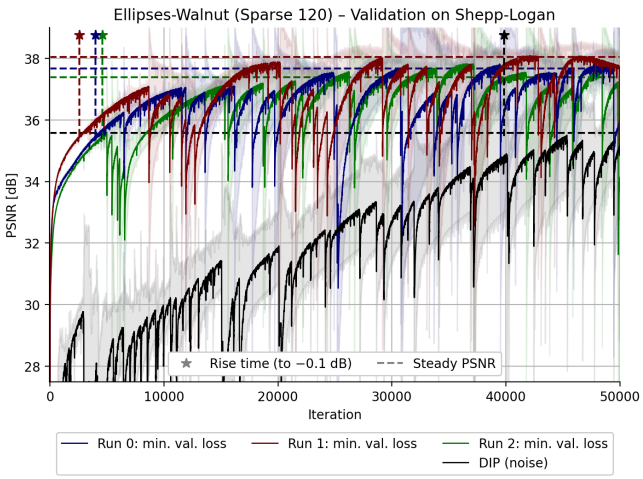

Fig. 27. Validation runs on the Shepp-Logan phantom for selecting the initial EDIP (FBP) model parameters for the Walnut `Sparse 120` geometry. The model from training run 1 is selected because it has the shortest rise time (with a high steady PSNR).

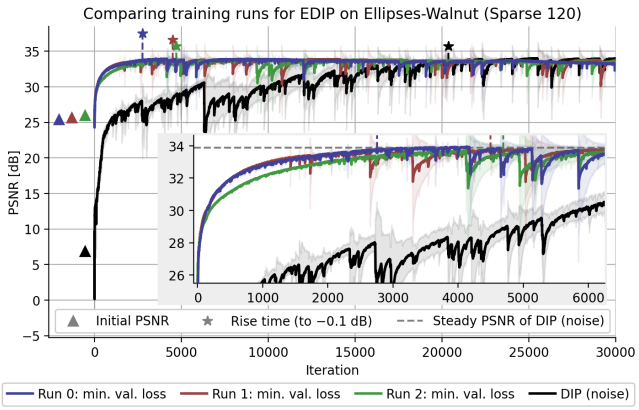

Fig. 28. Optimization of EDIP using parameters from different training runs considered during validation (see Fig. 27) for EDIP (FBP), on Walnut `Sparse 120` data. The parameters from run 1 are selected by the validation. The notations ▲ and ⋆ denote the initial PSNR and rise time, respectively, and the horizontal dashed line indicates steady PSNR of DIP (noise).

SUPPLEMENTARY MATERIAL E
ABLATION STUDY AND LIMITATIONS

We showcase one potential pitfall of the "supervised pretraining + unsupervised fine-tuning" paradigm for DIP, resorting to a by far too specific and less diverse image class. Instead of the ellipses dataset, we use human brain images for the supervised learning stage. We consider MRI images of the human brain from the ACRIN-FMISO-Brain (ACRIN 6684) dataset from https://wiki.cancerimagingarchive.net/x/kQIGAg. For the synthetic dataset, we normalize the extracted 2D slices and (mis)interpret the values to be X-ray attenuation coefficients. We use a random data split on patient level, leading to 30 917 training images and 4524 validation images. Both training and validation images are augmented by random rotations. Fig. 29 shows an exemplary reconstruction of the brain dataset, whilst Fig. 30 reports the pretraining convergence.

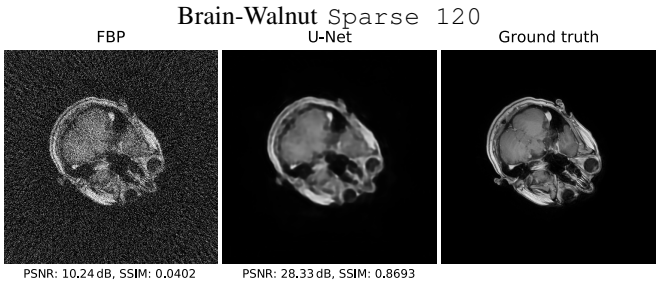

Fig. 29. Exemplary reconstructions from the synthetic training dataset of brain images for Walnut Sparse 120.

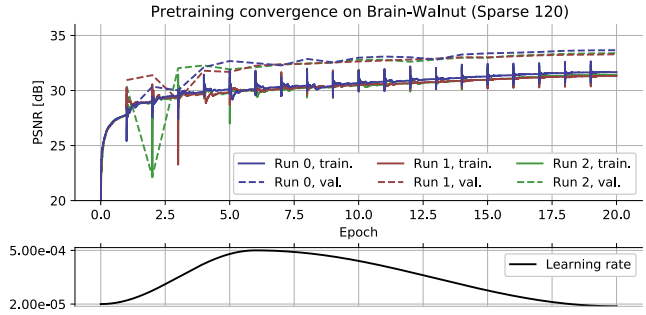

Fig. 30. Pretraining convergence.

In Fig. 31, we show the validation on the Shepp-Logan. Fig. 32 compares DIP and EDIP trained on the brain dataset. EDIP performs worse than DIP.

Fig. 33 suggests that checkpoints from repeated pretraining runs also lead to similar subpar results. We observe the inadequacy of the brain dataset (of its education!). Pretraining on the brain dataset induces too dataset specific inductive biases from which EDIP fails to escape, leading to slow convergence and sub-optimal steady PSNR. Possibly the implicit regularization exerted by the pretraining on the brain dataset essentially restricts the networks from leaving a "pretrained landscape" of sub-optimal parameters' configurations.

We then check whether using earlier checkpoints would lead to better transferable performances. We, indeed, observe that

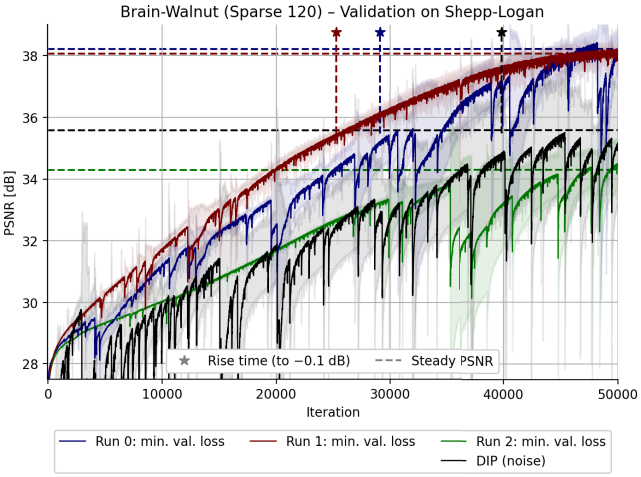

Fig. 31. Validation runs on the Shepp-Logan phantom for selecting the initial EDIP (FBP) model parameters pretrained on the brain dataset for data in the Walnut Sparse 120 geometry. The model from training run 1 is selected because it has the shortest rise time. Despite the relatively high number of 50k iterations, the (E)DIP optimizations do not fully converge yet.

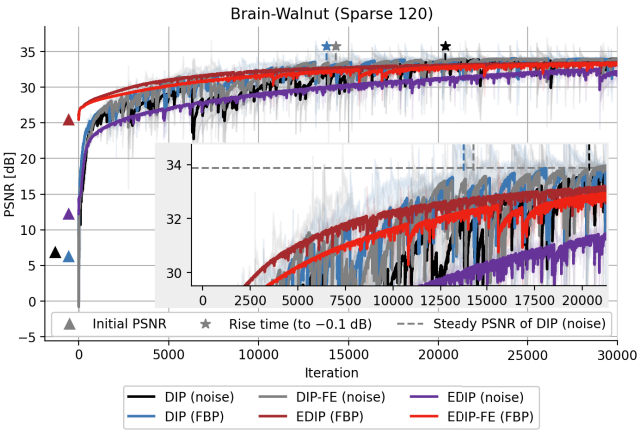

Fig. 32. The optimization of EDIP versus DIP pretrained on the brain dataset compared to standard DIP on Walnut Sparse 120 measurement data. All traces are the mean PSNR of 5 repetitions of the same experimental run (varying the random seed). See Tab. XI for complementary tabular results. The notations ▲ and ⋆ denote the initial PSNR and rise time, respectively.

TABLE XI
QUANTITATIVE EVALUATION RESULTS FOR EDIP ON WALNUT SPARSE 120 AFTER PRETRAINING ON THE BRAIN DATASET FOR 20 EPOCHS. NO RISE TIME CAN BE REPORTED, BECAUSE THE PSNR IS NOT REACHING THE STEADY PSNR OF DIP (NOISE) MINUS 0.1 dB WITHIN THE 30K ITERATIONS. SEE TABLE IX FOR THE CORRESPONDING RESULTS FROM STANDARD DIP AND FROM PRETRAINING ON ELLIPSES DATA.

Brain-Walnut Sparse 120 — pretrained for 20 epochs

	Rise time	(Max PSNR; iters)	Steady PSNR	Init PSNR
EDIP (FBP)	–	(33.51; 29 982)	33.35	25.49
EDIP (noise)	–	(32.67; 29 875)	32.29	12.23
EDIP-FE (FBP)	–	(33.43; 29 862)	33.24	25.49
EDIP-FE (noise)	–	(31.06; 29 989)	30.39	12.23

an early-stopping of the pretraining stage on the brain dataset ameliorates EDIP, cf. Fig. 35. The longer we pretrain on the brain dataset, the worse EDIP performs subsequently.

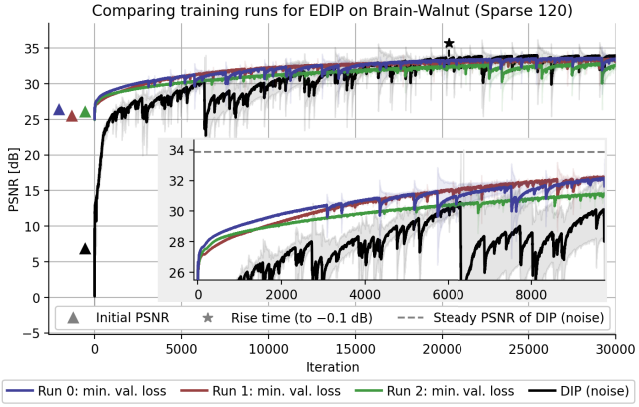

Fig. 33. The optimization of EDIP using parameters from different training runs considered during validation (see Fig. 31) for EDIP (FBP), pretrained on the brain dataset, on Walnut `Sparse 120` measurement data. The parameters from run 1 are the ones selected by the validation. The notations ▲ and ⋆ denote the initial PSNR and rise time, respectively, and the horizontal dashed line indicates steady PSNR of DIP (noise).

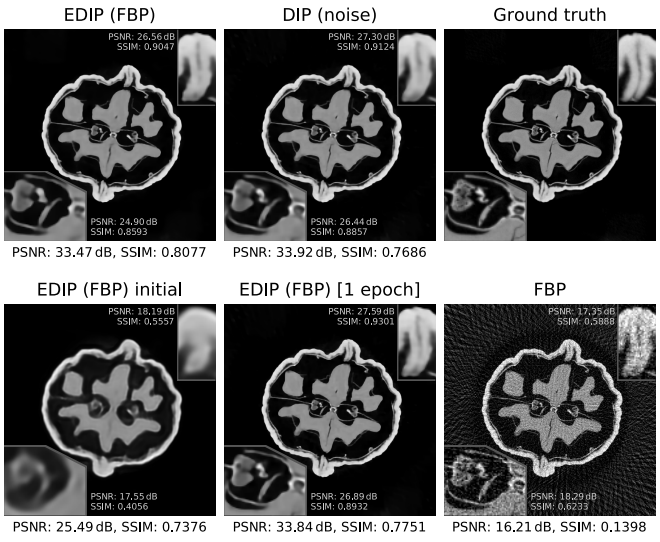

Fig. 34. Walnut reconstruction of EDIP pretrained on brain dataset, compared to standard DIP. From the 5 runs (varying the seed), the one with the (closest to) median PSNR was selected for each method. See Fig. 8 for the Walnut reconstruction with EDIP pretrained on the ellipses dataset.

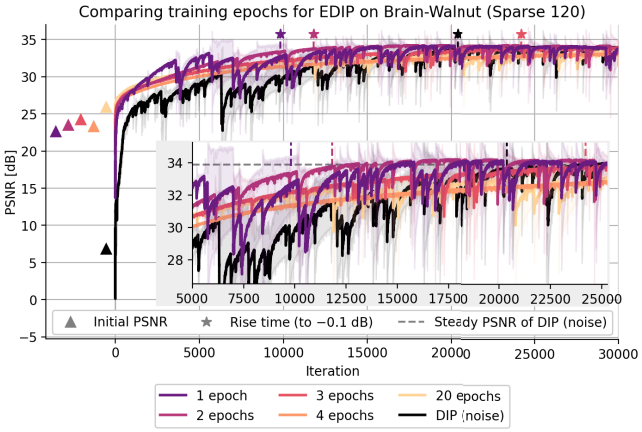

Fig. 35. The optimization of EDIP using parameters from different epochs for EDIP (FBP) on Walnut `Sparse 120` measurement data while pretraining on the brain dataset. The notations ▲ and ⋆ denote the initial PSNR and rise time, respectively, and the horizontal dashed line indicates steady PSNR of DIP (noise).

Paper 5

Uncertainty estimation for computed tomography with a linearised deep image prior

Published in Transactions on Machine Learning Research (12/2023)

Uncertainty Estimation for Computed Tomography with a Linearised Deep Image Prior

Javier Antorán[*] *ja666@cam.ac.uk*
Department of Engineering, University of Cambridge

Riccardo Barbano[*] *riccardo.barbano@tum.de*
School of Computation, Information and Technology, Technical Univeristy of Munich

Johannes Leuschner *jleuschn@uni-bremen.de*
Center for Industrial Mathematics, University of Bremen

José Miguel Hernández-Lobato *jmh233@cam.ac.uk*
Department of Engineering, University of Cambridge

Bangti Jin *b.jin@cuhk.edu.hk*
Department of Mathematics, The Chinese University of Hong Kong

Reviewed on OpenReview: *https://openreview.net/forum?id=FWyabz82fH*

Abstract

Existing deep-learning based tomographic image reconstruction methods do not provide accurate uncertainty estimates of their reconstructions, hindering their real-world deployment. This paper develops a method, termed as linearised deep image prior (DIP), to estimate the uncertainty associated with reconstructions produced by the DIP with total variation (TV) regularisation. We endow the DIP with conjugate Gaussian-linear model type error-bars computed from a local linearisation of the neural network around its optimised parameters. To preserve conjugacy, we approximate the TV regulariser with a Gaussian surrogate. This approach provides pixel-wise uncertainty estimates and a marginal likelihood objective for hyperparameter optimisation. We demonstrate the method on synthetic data and real-measured high-resolution 2D μCT data, and show that it provides superior calibration of uncertainty estimates relative to previous probabilistic formulations of the DIP. Our code is available at `https://github.com/educating-dip/bayes_dip`.

1 Introduction

Inverse problems in imaging aim to recover an unknown image $x \in \mathbb{R}^{d_x}$ from the noisy measurement $y \in \mathbb{R}^{d_y}$

$$y = \mathrm{A}x + \eta, \tag{1}$$

where $\mathrm{A} \in \mathbb{R}^{d_y \times d_x}$ is a linear forward map, and η i.i.d. Gaussian noise, i.e. $\eta \sim \mathcal{N}(0, \sigma_y^2 \mathrm{I})$. Many tomographic reconstruction problems take this form, e.g. computed tomography (CT). Due to the inherent ill-posedness of the problem, e.g. $d_y \ll d_x$, suitable regularisation / prior is crucial for the successful recovery of x (Tikhonov & Arsenin, 1977; Engl et al., 1996; Ito & Jin, 2014).

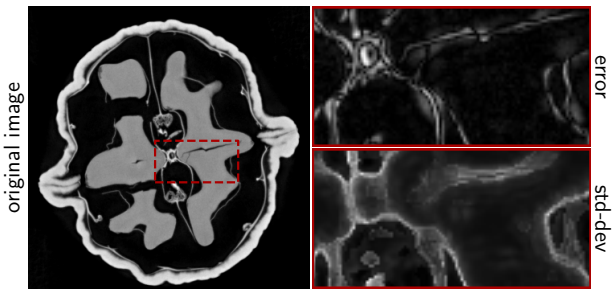

Figure 1: X-ray reconstruction ($501 \times 501\,\mathrm{px}^2$) of a walnut (left), the absolute error of its CT reconstruction (top) and pixel-wise uncertainty (bottom).

[*] Equal contribution.

In recent years, deep-learning based approaches have achieved outstanding performance on a wide variety of tomographic problems (Arridge et al., 2019; Ongie et al., 2020; Wang et al., 2020). Most deep learning methods are supervised; they rely on large volumes of paired training data. Alas, these often fail to generalise out-of-distribution (Antun et al., 2020); small deviations from the distribution of the training data can lead to severe reconstruction artefacts. Pathologies of this sort call for both unsupervised deep learning methods—free from training data and thus mitigating hallucinatory artefacts (Bora et al., 2017; Heckel & Hand, 2019; Tölle et al., 2021)—and uncertainty quantification (Kompa et al., 2021; Vasconcelos et al., 2022)—informing the user about (un)reliability in reconstructions.

We focus on the deep image prior (DIP), perhaps the most widely adopted unsupervised deep learning approach (Ulyanov et al., 2018). DIP regularises the reconstructed image $\hat{x}$ by reparametrising it as the output of a deep convolutional neural network (CNN). It does not require paired training data, relying solely on the structural biases induced by the CNN architecture. The DIP has proven effective on tasks ranging from denoising and deblurring to challenging tomographic reconstructions (Liu et al., 2019; Baguer et al., 2020; Knopp & Grosser, 2022; Darestani & Heckel, 2021; Gong et al., 2019; Cui et al., 2021; Barutcu et al., 2022). Nonetheless, the DIP only provides point reconstructions without uncertainty estimates.

In this work, we equip DIP reconstructions with reliable uncertainty estimates, which is an under-explored topic. In literature, there are two notable probabilistic reformulations of the DIP (Cheng et al., 2019; Tölle et al., 2021), but their focus is on preventing overfitting rather than accurately estimating uncertainty. Distinctly from these, we only estimate the uncertainty associated with a specific reconstruction, instead of characterising a full posterior over all candidate images. We achieve this by computing Gaussian-linear model type error-bars for a local linearisation of the DIP around its mode (Mackay, 1992; Khan et al., 2019; Immer et al., 2021b), and refer to the method as *linearised* DIP. Linearised approaches have recently provided state-of-the-art uncertainty estimates for supervised deep learning models (Daxberger et al., 2021b). Unfortunately, the total variation (TV) regulariser, ubiquitous in CT reconstruction, makes inference in the linearised DIP intractable and it does not lend itself to standard Laplace (i.e. local Gaussian) approximations (Helin et al., 2022). We tackle this issue using predictive complexity prior (PredCP) (Nalisnick et al., 2021) to construct covariance kernels that induce properties similar to that of the TV prior while preserving Gaussian-linear conjugacy. Finally, we discuss several techniques to scale the method to large DIP networks and high-resolution 2D images.

We showcase our approach on high-resolution CT reconstructions of real-measured 2D μCT projection data, cf. fig. 1. Empirically, the method's pixel-wise uncertainty estimates predict reconstruction errors more accurately than existing approaches to uncertainty estimation with the DIP. This is not at the expense of accuracy in reconstruction: the reconstruction obtained using the standard regularised DIP method (Baguer et al., 2020) is preserved as the predictive mean, ensuring compatibility with advancements in DIP research.

The contributions of this work can be summarised as follows.

- We propose a novel approach to bestow reconstructions from the TV-regularised DIP with uncertainty estimates, by constructing a local linear model by linearising the DIP around its optimised reconstruction and providing the model's error-bars as a surrogate for those of the DIP.
- We give an efficient implementation of the method, scaling up to high-resolution μCT data, and yielding far more accurate uncertainty estimation than existing probabilistic formulations of the DIP.

The rest of this paper is organised as follows. Section 2 provides an extended discussion of the related work. Section 3 recalls preliminaries for the linearised DIP. Section 4 discusses the design of a tractable Gaussian prior mimicking the TV prior. Section 5 and section 6 present the linearised DIP and its efficient implementation. Section 7 presents the experimental investigations on synthetic and real-measured high-resolution μCT data. Section 8 concludes the article. Fully detailed derivations and additional experimental results are given in the supplementary material (SM).

Since this paper's first appearance, the proposed method was used by Barbano et al. (2022b) to actively select X-ray scanning angles, resulting in a 30% reduction in angles needed to obtain a given reconstruction PSNR, and extended by Antoran et al. (2023), scaling it to larger problems by drawing samples with SGD.

2 Related Work

2.1 Advances in the deep image prior

Since its introduction by Ulyanov et al. (2018; 2020), the DIP has been improved with early stopping (Wang et al., 2021), TV regularisation (Liu et al., 2019; Baguer et al., 2020) and pretraining (Barbano et al., 2022c; Knopp & Grosser, 2022; Barbano et al., 2023). We build upon these recent advancements by providing a scalable method to estimate the error-bars of DIP's reconstructions. Obtaining reliable uncertainty estimates for DIP reconstructions is a relatively unexplored topic. Building upon Garriga-Alonso et al. (2019) and Novak et al. (2019), Cheng et al. (2019) show that in the infinite-channel limit, the DIP converges to a Gaussian process (GP). In the finite-channel regime, the authors approximate the posterior distribution over the DIP's parameters with stochastic gradient Langevin dynamics (SGLD) (Welling & Teh, 2011). Laves et al. (2020) and Tölle et al. (2021) use factorised Gaussian variational inference (Blundell et al., 2015) and MC dropout (Hron et al., 2018; Vasconcelos et al., 2022), respectively. These probabilistic treatments of DIP primarily aim to prevent overfitting, as opposed to accurately estimating uncertainty. While they can deliver uncertainty estimates, their quality tends to be poor. In fact, obtaining reliable uncertainty estimates from deep-learning based approaches, like the DIP, largely remains a challenging open problem (Antorán, 2019; Snoek et al., 2019; Ashukha et al., 2020; Foong et al., 2020; Barbano et al., 2022a; Antorán et al., 2020). In the present work, we obtain uncertainty estimation by performing Bayesian inference with respect to the DIP model locally linearised around its optimised parameters. This is distinct from the aforementioned approaches in that we only model a local mode of the posterior distribution.

2.2 Bayesian inference in linearised neural networks

The Laplace method is first applied to deep learning in (Mackay, 1992). It has seen a recent popularisation as the best performing approach when it comes to Bayesian reasoning with neural networks (Daxberger et al., 2021b;a). Specifically, Khan et al. (2019) and Immer et al. (2021b) show that the linearization step improves the quality of uncertainty estimates. Immer et al. (2021a), Antorán et al. (2022) and Antorán et al. (2022) explore the linear model's evidence for model selection. Daxberger et al. (2021b) and Maddox et al. (2021) introduce subnetwork and finite differences approaches, respectively, for scalable inference with linearised models. Inference in the linearised model is highly attractive compared to alternative approaches because it is post-hoc and it preserves the reconstruction obtained through the DIP optimisation as the predictive mean. This line of work is also related to the neural tangent kernel (Jacot et al., 2018; Lee et al., 2019; Novak et al., 2020), in which NNs are linearised at initialisation.

3 Preliminaries

3.1 Total variation regularisation

The imaging problem given in eq. (1) admits multiple solutions consistent with the observation y. Thus, regularisation is needed for stable reconstruction. Total variation (TV) is perhaps the most well established regulariser (Rudin et al., 1992; Chambolle et al., 2010). The anisotropic TV semi-norm of an image vector $x \in \mathbb{R}^{d_x}$ imposes an L^1 constraint on image gradients:

$$\mathrm{TV}(x) = \sum_{i,j} |X_{i,j} - X_{i+1,j}| + \sum_{i,j} |X_{i,j} - X_{i,j+1}|, \tag{2}$$

where $X \in \mathbb{R}^{h \times w}$ denotes the vector x reshaped into an image of height h by width w, and $d_x = h \cdot w$. This leads to the regularised reconstruction formulation

$$\hat{x} \in \operatorname*{argmin}_{x \in \mathbb{R}^{d_x}} \mathcal{L}(x) \quad \text{with} \quad \mathcal{L}(x) := \|\mathrm{A}x - y\|_2^2 + \lambda \mathrm{TV}(x), \tag{3}$$

where the hyperparameter $\lambda > 0$ determines the strength of the regularisation relative to the fit term.

3.2 Bayesian inference for inverse problems

The Bayesian framework provides a consistent approach to uncertainty estimation in imaging problems (Kaipio & Somersalo, 2005; Stuart, 2010; Seeger & Nickisch, 2011). The image to be recovered is treated as a random variable. Instead of finding a single best reconstruction $\hat{x}$, we aim to find a posterior distribution $p(x|y)$ that scores every candidate $x \in \mathbb{R}^{d_x}$ according to its agreement with the observation y and prior belief $p(x)$. The loss in eq. (3) can be viewed as the negative log of an unnormalised posterior, i.e. $p(x|y) \propto \exp(-\mathcal{L}(x))$, and $\hat{x}$ as its mode, i.e. the *maximum a posteriori* (MAP) estimate. The least squares loss corresponds to a Gaussian likelihood $p(y|x) = \mathcal{N}(y;\, \mathrm{A}x, \mathrm{I})$ and the TV regulariser to a prior over images $p(x) \propto \exp(-\lambda \mathrm{TV}(x))$.

The posterior is obtained by updating the prior over images with the likelihood as

$$p(x|y) = p(y)^{-1} p(y|x) p(x), \tag{4}$$

for $p(y) = \int p(y|x)p(x)\mathrm{d}x$ the normalising constant, also known as the marginal likelihood (MLL). This latter quantity provides an objective for optimising hyperparameters, e.g. the regularisation strength λ. The presence of different reconstructions with high probability under the posterior indicates uncertainty.

Our work partially departs from this framework in that it *solely concerns itself with characterising plausible reconstructions around the mode* $\hat{x}$ (Mackay, 1992). This has two key advantages, i) *tractability*: the likelihood induced by NN reconstructions is strongly multi-modal, and both analytically and computationally intractable. In contrast, the posterior for the local model is Gaussian; ii) *interpretablity*: even if we could obtain the full posterior, downstream stakeholders not versed in probability are likely to have little use for it. A single reconstruction and its pixel-wise uncertainty may be more interpretable to end-users (Bhatt et al., 2021).

3.3 The Deep Image Prior (DIP)

The DIP (Ulyanov et al., 2018; 2020) reparametrises the reconstructed image as the output of a CNN $x(\theta)$ with learnable parameters $\theta \in \mathbb{R}^{d_\theta}$ and a fixed input, which we have omitted from our notation for clarity. The DIP can be seen as a reparametrisation that provides a favourable structural bias towards natural images. Penalising the TV of the DIP's output avoids the need for early stopping and improves reconstruction fidelity (Liu et al., 2019; Baguer et al., 2020). The resulting optimisation problem is given by

$$\hat{\theta} \in \operatorname*{argmin}_{\theta \in \mathbb{R}^{d_\theta}} \|\mathrm{A}x(\theta) - y\|_2^2 + \lambda \mathrm{TV}(x(\theta)), \tag{5}$$

and the recovered image is given by $\hat{x} = x(\hat{\theta})$. U-Net is the standard choice of CNN architecture (Ronneberger et al., 2015). Although the parameters θ must be optimised separately for each new measurement y, we follow (Barbano et al., 2022c; Knopp & Grosser, 2022) to reduce the cost with task-agnostic pretraining.

3.4 Bayesian inference with linearised neural networks

Adopting the DIP parametrisation of the reconstructed image, as in section 3.3, makes the Bayesian posterior in eq. (4) intractable. Instead, this work only characterises the uncertainty associated with a specific regularised reconstruction $\hat{x}$, obtained via eq. (5). To this end, we take a *tangent linear model* of the CNN $x(\theta)$ around its optimised parameters $\hat{\theta}$ (Mackay, 1992; Khan et al., 2019; Immer et al., 2021b),

$$h(\theta) \coloneqq x(\hat{\theta}) + J(\theta - \hat{\theta}), \tag{6}$$

where $J \coloneqq \frac{\partial x(\theta)}{\partial \theta}|_{\theta=\hat{\theta}} \in \mathbb{R}^{d_x \times d_\theta}$ is the Jacobian of the CNN function $x(\theta)$ with respect to its parameters θ evaluated at $\hat{\theta}$. We obtain error-bars for the DIP reconstruction $x(\hat{\theta})$ using $h(\theta)$. For Gaussian noise and a Gaussian prior on θ, we have a conjugate setting; the posterior over the linearised model's reconstructions is a Gaussian $\mathcal{N}(x; x(\hat{\theta}), \Sigma_{x|y})$, and the marginal likelihood of the linearised model can be used to tune hyperparameters (Mackay, 1992; Immer et al., 2021a; Antorán et al., 2022; Antorán et al., 2022).

Computing both the posterior covariance $\Sigma_{x|y}$ and the marginal likelihood naively has cost $\mathcal{O}(d_\theta^3)$. For large U-Nets, this is impracticable (Daxberger et al., 2021b). In section 5 and section 6, we derive a dual approach with a cost $\mathcal{O}(d_y^3)$ and detail an efficient implementation. Furthermore, when using the (non-quadratic) TV regulariser, conjugacy is lost. Indeed, the TV regulariser does not admit a Laplace (quadratic) approximation.

4 The total variation as a conditionally Gaussian prior

First, we study the construction of tractable non-DIP-based priors for CT reconstruction. The gained understanding sheds insights into incorporating the TV-based priors into the linearised DIP framework. The regularised loss in eq. (3) can be interpreted as the negative log of an unnormalised posterior over reconstructions. In this context, the TV regulariser corresponds to the prior

$$p(x) = Z_\lambda^{-1} \exp(-\lambda \mathrm{TV}(x)), \tag{7}$$

where $Z_\lambda = \int \exp(-\lambda \mathrm{TV}(x))\, \mathrm{d}x$ is its normalisation constant (the prior is improper, since constant vectors are in the null space of the derivative operator). Working with the prior $p(x)$ is intractable since Z_λ does not admit a closed form. The Laplace method, which consists of a locally quadratic approximation, does not solve the issue because the second derivative of the TV regulariser is zero everywhere it is defined.

To enforce local smoothness in the reconstruction, we construct a Gaussian prior $\mathcal{N}(x; \mu, \Sigma_{xx})$ with mean $\mu \in \mathbb{R}^{d_x}$ and covariance $\Sigma_{xx} \in \mathbb{R}^{d_x \times d_x}$ given by the Matern-$1/2$ kernel

$$[\Sigma_{xx}]_{ij,i'j'} = \sigma^2 \exp\left(\frac{-\mathrm{d}(i-i', j-j')}{\ell}\right), \tag{8}$$

where i, j index the spatial locations of pixels of x, as in eq. (2), and $\mathrm{d}(a, b) = \sqrt{a^2 + b^2}$. The hyperparameter $\sigma^2 \in \mathbb{R}^+$ informs the pixel amplitude while the lengthscale parameter $\ell \in \mathbb{R}^+$ determines the correlation strength between nearby pixels. The expected TV associated with our Gaussian prior is

$$\kappa := \mathbb{E}_{x \sim \mathcal{N}(\mu, \Sigma_{xx})}[\mathrm{TV}(x)] = c\sigma\sqrt{1 - \exp(-\ell^{-1})}, \tag{9}$$

with c a constant. See appendix A for a derivation. Below we may omit the dependence of κ on (ℓ, σ^2) from the notation. For fixed pixel amplitude σ^2, the expected reconstruction TV κ is a bijection of the lengthscale ℓ. We leverage this fact within the PredCP framework of Nalisnick et al. (2021) to construct a prior over ℓ that favours reconstructions with low expected TV

$$p(\ell) = \mathrm{Exp}(\kappa)\, |\partial\kappa/\partial\ell|\,, \tag{10}$$

where Exp is the density of the exponential distribution. The resulting hierarchical prior over images

$$x|\ell \sim \mathcal{N}(\mu, \Sigma_{xx}), \quad \ell \sim \mathrm{Exp}(\kappa)\, |\partial\kappa/\partial\ell| \tag{11}$$

is Gaussian for fixed ℓ, and thus the prior is conditionally conjugate to Gaussian-linear likelihoods. Figure 2 shows agreement between samples, drawn with Hamiltonian Monte Carlo, from the described TV-PredCP prior and the intractable TV prior, both qualitatively and in terms of distribution over image TV. The TV prior produces samples with more correlated nearby pixel values than the factorised prior. The TV-PredCP prior captures this effect and produces even smoother samples, likely due to the presence of longer range correlation in the Matern-$1/2$ covariance.

5 The linearised DIP

In this section, we build a probabilistic model to characterise posterior reconstructions around $\hat{\theta}$, a mode of the regularised DIP objective (obtained using eq. (5)). Section 5.1 describes the construction of a linearised surrogate for the DIP reconstruction. Section 5.2 describes how to compute the surrogate model's error-bars and use them to augment the DIP reconstruction. Section 5.3 discusses how we include the effects of TV regularisation into the surrogate model. Finally, in section 5.4, we describe a strategy to choose the surrogate model's prior hyperparameters using a marginal likelihood objective.

5.1 From a prior over parameters to a prior over images

After training the DIP to an optimal TV-regularised setting $\hat{x} = x(\hat{\theta})$ using eq. (5), we linearise the network around $\hat{\theta}$ by applying eq. (6), and obtain the affine-in-θ function $h(\theta)$. The error-bars obtained from Bayesian

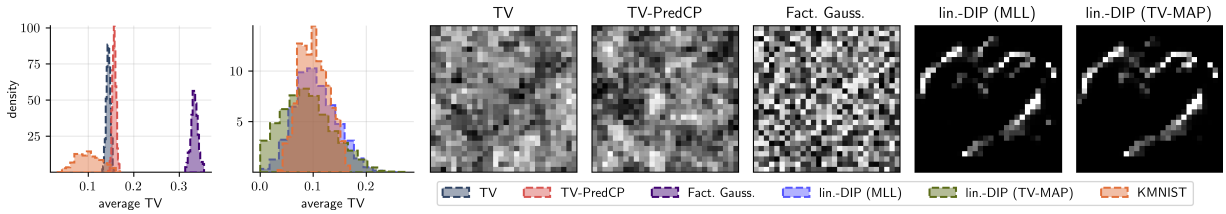

Figure 2: Samples from priors. From left to right. Plot 1 shows a histogram of the average sample TV reporting an overlap between the TV and TV-PredCP priors. The factorised Gaussian prior results in larger TV values. Plot 2 shows an analogous histogram using samples from the linearised DIP (lin.-DIP) fitted to a KMNIST image, where the hyperparameters (ℓ, σ^2) have been optimised both with and without the TV-PredCP term. The TV-PredCP term in the DIP hyperparameter optimisation leads to smoother samples with less artefacts. Plots 3-5 show samples from the TV, TV-PredCP, and factorised Gaussian priors proposed in section 4, drawn using Hamiltonian Monte Carlo (HMC). Plots 6-7 show prior samples from the linearised DIP, which produces samples containing the structure of the KMNIST image used to train the network.

inference with $h(\theta)$ will tell us about the uncertainty in $\hat{x}$. To this end, consider the hierarchical model,

$$y|\theta \sim \mathcal{N}(\mathrm{A}h(\theta), \sigma_y^2 \mathrm{I}), \quad \theta|\ell \sim \mathcal{N}(0, \Sigma_{\theta\theta}(\ell)), \quad \ell \sim p(\ell) \quad \text{with} \quad h(\theta) \coloneqq x(\hat{\theta}) + J(\theta - \hat{\theta}), \tag{12}$$

where we place a Gaussian prior over the parameters θ that, in turn, depends on the lengthscale ℓ. Conditioned on the value of ℓ, this is a conjugate Gaussian-linear model and thus the posterior distribution over θ has a closed Gaussian form. Learning the lengthscale ℓ will allow us to incorporate TV constraints into the computed error-bars, cf. section 5.3. We have introduced the noise variance σ_y^2 as an additional hyperparameter which we will learn using the marginal likelihood (cf. section 5.4).

To provide intuition about the linearised model, we push samples from $\theta \sim \mathcal{N}(\theta; 0, \Sigma_{\theta\theta})$, through h. The resulting reconstruction samples are drawn from a Gaussian distribution with covariance $\Sigma_{xx} \in \mathbb{R}^{d_x \times d_x}$ given by $J\Sigma_{\theta\theta}J^\top$ and are shown in fig. 2. Here, the Jacobian J introduces structure from the NN function around the linearisation point $\hat{\theta}$. It introduces features from the KMNIST character that the DIP was trained on.

5.2 Efficient posterior predictive computation

We augment the DIP reconstruction $\hat{x}$ with Gaussian predictive error-bars computed with the linearised model h described in eq. (12), yielding $\mathcal{N}(x; \hat{x}, \Sigma_{x|y})$. The posterior covariance $\Sigma_{x|y}$ is given by

$$\Sigma_{x|y} = J(\sigma_y^{-2} J^\top \mathrm{A}^\top \mathrm{A} J + \Sigma_{\theta\theta}^{-1})^{-1} J^\top = \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{xy}^\top, \tag{13}$$

which is derived in appendix B. Here, $\Sigma_{xx} = J\Sigma_{\theta\theta}J^\top$, $\Sigma_{xy} = \Sigma_{xx}A^\top$ and $\Sigma_{yy} = A\Sigma_{xx}A^\top + \sigma_y^2 I$. The constant-in-$\theta$ terms in h do not affect the uncertainty estimates, and thus the error-bars match those of the simple linear model $J\theta$. Importantly, eq. (13) depends on the inverse of the observation space covariance Σ_{yy}^{-1}, as opposed to the covariance over reconstructions, or parameters. Equation (13) scales as $\mathcal{O}(d_x d_y^2)$ as opposed to $\mathcal{O}(d_x^3)$ or $\mathcal{O}(d_\theta^3)$ for the more-standard-in-the-literature output (reconstruction) space or parameter space approaches, respectively (Immer et al., 2021b; Daxberger et al., 2021a).

5.3 Incorporating TV-smoothness into our model as a prior

We impose constraints on h's error-bars, such that the model only considers low TV reconstructions as plausible. For this, we place a block-diagonal Matern-1/2 covariance Gaussian prior on the linearised model's weights, similarly to Fortuin et al. (2021). We introduce dependencies between parameters in the same CNN convolutional filter as

$$[\Sigma_{\theta\theta}]_{kij,k'i'j'} = \sigma_d^2 \exp\Big(\frac{-\mathrm{d}(i-i', j-j')}{\ell_d}\Big)\delta_{kk'}, \tag{14}$$

where k indexes the convolutional filters in the CNN, $\delta_{kk'}$ denotes Kronecker symbol, and (i, j) index the spatial locations of specific parameters within a filter. The lengthscale ℓ_d regulates the filter smoothness.

Intuitively, an image generated from convolutions with smoother filters will present lower TV. Indeed, in appendix C we show a bijective relationship between this quantity and the filter lengthscale. The hyperparameter σ_d^2 determines the marginal prior variance. Both parameters are defined per architectural block $d \in \{1, 2, \ldots, D\}$ in the U-Net and we write $\ell = [\ell_1, \ell_2, \ldots, \ell_D]$ and $\sigma^2 = [\sigma_1^2, \sigma_2^2, \ldots, \sigma_D^2]$. The chosen U-Net architecture is fully convolutional and thus eq. (14) applies to all parameters, reducing to a diagonal covariance for 1×1 convolutions. A U-Net diagram highlighting these prior blocks is in fig. 3.

To enforce TV-smoothness, we adopt the strategy given in section 4. Since choosing a large ℓ enforces smoothness in the output, a prior placed over the filter lengthscales ℓ can act as a surrogate for the TV prior.

To make this connection explicit, we construct a TV-PredCP (Nalisnick et al., 2021)

$$p(\ell) = \prod_{d=1}^{D} p(\ell_d) = \prod_{d=1}^{D} \mathrm{Exp}(\kappa_d) \left| \frac{\partial \kappa_d}{\partial \ell_d} \right|, \tag{15}$$

$$\text{with } \kappa_d := \mathbb{E}_{\theta \sim \mathcal{N}(\hat{\theta}_d, \Sigma_{\theta_d \theta_d})} \prod_{i=1, i \neq d}^{D} \delta(\theta_i - \hat{\theta}_i) \left[\lambda \mathrm{TV}(h(\theta)) \right] \tag{16}$$

being the expected TV of the CNN output over the prior uncertainty in the parameters of block d when all other entries of θ are fixed to $\hat{\theta}$.

We relate the expected TV κ_d to the filter lengthscale ℓ_d via the change of variables formula. The independence across blocks of $p(\ell)$ ensures dimensionality preservation, formally needed in changing variables. It follows from the triangle inequality that $\sum_d \kappa_d$ is an upper bound on the expectation under the distribution $\mathbb{E}_{\theta \sim \mathcal{N}(\hat{\theta}, \Sigma_{\theta\theta})}[\mathrm{TV}(h(\theta))]$, cf. appendix C.

Note that eq. (15) can be computed analytically. However, its direct computation is costly and we instead rely on numerical methods described in section 6. In fig. 2 (cf. plot 2 and plots 6-7), we show samples from $\mathcal{N}(x; 0, \Sigma_{xx})$ where ℓ is chosen using the marginal likelihood with TV-PredCP constraints (cf. also section 5.4). Incorporating the TV-PredCP leads to smoother samples with less discontinuities.

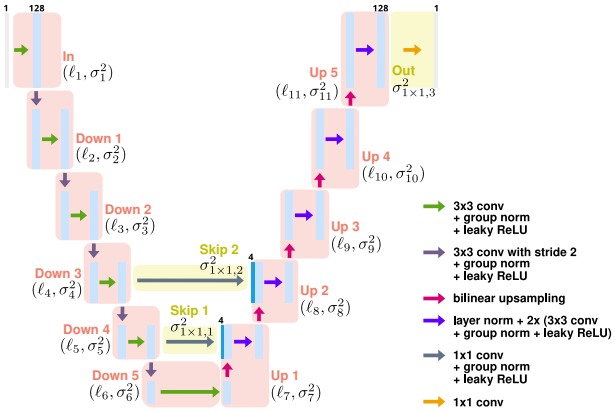

Figure 3: A schematic of the U-Net architecture used in the numerical experiments on Walnut data (see section 7.2). For KMNIST, we use a reduced, 3-scale U-Net without group norm layers (see fig. 20). Each light-blue rectangle corresponds to a multi-channel feature map. We highlight the architectural components corresponding to each block $1, \ldots, D$ for which a separate prior is defined with red and yellow boxes.

5.4 Type-II MAP learning of hyperparameters

The calibration of the predictive Gaussian errorbars depends on the choice of the hyperparameters $(\sigma_y^2, \sigma^2, \ell)$ of the hierarchical model in eq. (12) (Antorán et al., 2022). For a given ℓ, Gaussian-linear conjugacy yields a closed form marginal likelihood objective to learn the hyperparameters. In turn, to learn ℓ, we combine the above objective with the TV-PredCP's log-density, which acts as a regulariser. The resulting expression resembles a Type-II MAP (Rasmussen & Williams, 2005) objective

$$\begin{aligned} &\log p(y|\ell; \sigma_y^2, \sigma^2) + \log p(\ell; \sigma^2) \approx \\ &-\frac{1}{2}\sigma_y^{-2}||y - \mathrm{A}x(\hat{\theta})||_2^2 - \frac{1}{2}\hat{\theta}_h^\top \Sigma_{\theta\theta}^{-1}(\ell, \sigma^2)\hat{\theta}_h - \frac{1}{2}\log|\Sigma_{yy}| - \sum_{d=1}^{D} \kappa_d(\ell, \sigma^2) + \log\left|\frac{\partial \kappa_d(\ell, \sigma^2)}{\partial \ell_d}\right| + B, \end{aligned} \tag{17}$$

where B is independent of $(\sigma_y^2, \sigma^2, \ell)$ and the vector $\hat{\theta}_h \in \mathbb{R}^{d_\theta}$ is the posterior mean of the linear model's parameters. See appendix B for the detailed derivation. The bottleneck in evaluating eq. (17) is the log-determinant $\log|\Sigma_{yy}|$ of Σ_{yy}, which has a cost $\mathcal{O}(d_y^3)$. We go on to describe scalable ways to approximate the log-determinant and other costly quantities required for prediction.

6 Towards scalable computation

In a typical tomography setting, the dimensionality d_x of the image $\hat{x}$ and d_y of the observation y can be large, e.g. $d_x > 1\text{e}5$ and $d_y > 1\text{e}3$. Thus holding the input space covariance matrices (e.g. Σ_{xx} and $\Sigma_{x|y}$) in memory is infeasible. The latter complicates computing $\log|\Sigma_{yy}|$ in eq. (17) (or its gradients), and its inverse in $\Sigma_{x|y}$, cf. eq. (13), which scale as $\mathcal{O}(d_y^3)$ and $\mathcal{O}(d_x d_y^2)$, respectively. To scale the approach, we only access Jacobian and covariance matrices through matrix–vector products (*matvecs*), i.e. products resembling $v_x^\top \Sigma_{xx}$ and $v_y^\top \Sigma_{yy}$ for $v_x \in \mathbb{R}^{d_x}$ and $v_y \in \mathbb{R}^{d_y}$. We compute $v_y \in \mathbb{R}^{d_y}$ through successive matvecs as

$$v_y^\top \Sigma_{yy} = v_y^\top (\mathrm{A} J \Sigma_{\theta\theta} J^\top \mathrm{A}^\top + \sigma_y^2 \mathrm{I}), \tag{18}$$

and we compute $v_x^\top \Sigma_{xx}$ similarly. We compute Jacobian vector products $v_\theta^\top J^\top$ for $v_\theta \in \mathbb{R}^{d_\theta}$ using forward mode automatic differentiation (AD) and $v_x^\top J$ using backward mode AD, both with the `functorch` library (He & Zou, 2021). We compute products with $\Sigma_{\theta\theta}$ by exploiting its block diagonal structure. All these operations can be batched using modern numerical libraries and GPUs.

6.1 Conjugate gradient log-determinant gradients

For the Type-II MAP optimisation in eq. (17), we estimate the gradients of $\log|\Sigma_{yy}|$ with respect to the parameters of interest ϕ using the stochastic trace estimator (Gibbs & MacKay, 1996; Gardner et al., 2018)

$$\frac{\partial \log|\Sigma_{yy}|}{\partial \phi} = \mathrm{Tr}\left(\Sigma_{yy}^{-1}\frac{\partial \Sigma_{yy}}{\partial \phi}\right) = \mathbb{E}_{v\sim\mathcal{N}(0,P)}\left[v^\top \Sigma_{yy}^{-1}\frac{\partial \Sigma_{yy}}{\partial \phi} P^{-1} v\right], \tag{19}$$

where P is a preconditioner matrix. We approximately solve the linear system $v^\top \Sigma_{yy}^{-1}$ for batches of probe vectors v using the `GPyTorch` preconditioned conjugate gradient (PCG) implementation (Dong et al., 2017).

The preconditioner P is constructed using r-rank randomised SVD, by approximating $\mathrm{A} J \Sigma_{\theta\theta} J^\top \mathrm{A}^\top$ as $\tilde{U}\tilde{\Lambda}\tilde{U}^\top$, using a randomised eigendecomposition algorithm (Halko et al., 2011; Martinsson & Tropp, 2020) with $\tilde{U} \in \mathbb{R}^{d_y \times r}$ and $r = 200 \ll d_y$. The algorithm is described in detail in appendix E. Since P depends on the hyperparameters ϕ, we interweave the updates of P with the optimisation of eq. (17).

6.2 Ancestral sampling for TV-PredCP optimisation

For large images, exact evaluation of the expected TV with eq. (16) is intractable. Instead, we estimate the gradient of κ_d with respect to $\phi = (\sigma^2, \ell)$ using a Monte-Carlo approximation

$$\frac{\partial \kappa_d}{\partial \phi} = \mathbb{E}_{\theta_d \sim \mathcal{N}(\hat{\theta}_d, \Sigma_{\theta_d\theta_d})}\left[\frac{\partial \mathrm{TV}(x)}{\partial x} J_d \frac{\partial \theta_d}{\partial \phi}\right], \tag{20}$$

where $J_d = \frac{\partial x(\theta)}{\partial \theta_d}|_{\theta_d=\hat{\theta}_d}$, $\frac{\partial \mathrm{TV}(x)}{\partial x}$ is evaluated at the sample $x{=}J_d\theta_d$ and $\frac{\partial \theta_d}{\partial \phi}$ is the reparametrisation gradient for θ_d, a prior sample of the weights of CNN block d. Since the second derivative of the TV semi-norm is almost everywhere zero, the gradient for the change of variables volume ratio is

$$\frac{\partial^2 \kappa_d}{\partial \phi^2} = \mathbb{E}_{\theta_d \sim \mathcal{N}(\hat{\theta}_d, \Sigma_{\theta_d\theta_d})}\left[\frac{\partial \mathrm{TV}(x)}{\partial \phi} J_d \frac{\partial^2 \theta_d}{\partial \phi^2}\right]. \tag{21}$$

6.3 Posterior covariance matrix estimation by sampling

The covariance matrix $\Sigma_{x|y}$ is too large to fit into memory for high-resolution tomographic reconstructions. Instead, we follow Wilson et al. (2021) in drawing samples from $\mathcal{N}(x; 0, \Sigma_{x|y})$ via Matheron's rule

$$x_{x|y} = x_0 + \Sigma_{xy}\Sigma_{yy}^{-1}(\epsilon - \mathrm{A}x_0); \quad x_0 = J\theta_0; \quad \theta_0 \sim \mathcal{N}(0, \Sigma_{\theta\theta}); \quad \epsilon \sim \mathcal{N}(0, \sigma_y^2 \mathrm{I}). \tag{22}$$

The biggest cost lies in constructing Σ_{yy}, which is achieved by applying eq. (18) to the standard basis vectors $\Sigma_{yy} = [e_1, e_2, \ldots e_{d_y}]^\top \Sigma_{yy}$. We then perform its Cholesky factorisation as an intermediate step towards matrix

inversion, both relatively costly operations. Fortunately, we only have to repeat these once, after which the sampling step in eq. (22) can be evaluated cheaply. Alternatively, as in eq. (19), we can compute the solution of the linear system, $\Sigma_{yy}^{-1} v_y$ for any v_y via PCG, without explicitly assembling (and thus storing in memory) the measurement covariance matrix, or computing its Cholesky factorisation. This approach allows us to scale the sampling operation to large measurement spaces, where the matrix Σ_{yy} may not fit in memory.

Since only nearby pixels of the predictions are expected to be correlated, we estimate cross covariances for patches of only up to 10×10 adjacent pixels. Using larger patches yields no improvements. We use the stabilised formulation of Maddox et al. (2019): $\hat{\Sigma}_{x|y} = \frac{1}{2k}[\sum_{j=1}^{k} \operatorname{diag}(x_j)^2 + x_j x_j^\top]$ for $(x_j)_{j=1}^k$ samples from the posterior predictive distribution over a patch. Note that the samples from eq. (22) are zero mean.

6.4 Faster low-rank Jacobian matvecs

Table 1 shows that the Jacobian matvecs— implemented through forward and backward mode AD— required for sampling from the posterior predictive (that is $2\times$ $v_\theta^\top J^\top$ and $1\times$ $v_x^\top J$) take $\approx 100\,\%$ of this step's computation time (2.4 h). To accelerate sampling, we construct a low-rank approximation of the Jacobian $\tilde{J}$, which we store in memory. We compute $v_\theta^\top \tilde{J}^\top$ and $v_x^\top \tilde{J}$ via matvec, as opposed to AD. This allows for fast approximation of $v_y^\top \Sigma_{yy}$ by substituting $\tilde{J}$ into eq. (18). This brings the time needed for sampling from the posterior predictive down from 2.4 hours to less than a minute. We construct $\tilde{J}$ similarly to the low-rank preconditioner P (see section 6.1 and appendix E). That is, following Halko et al. (2011), we build a rank-r approximation to J, by accessing only to matvecs with J and $J^\top$. While offering a well-calibrated alternative to uncertainty quantification within the DIP framework, it incurs computational overhead (see table 1) when compared to MC dropout, which only require a forward pass through the network to generate a single sample.

Table 1: Wall-clock time on an A100 GPU for the different steps of our algorithm when applied to high-resolution CT (details in section 7.2). Computations reported below the dotted line are in double precision. The time taken by Jacobian matvecs during sampling is given in parenthesis.

	wall-clock time
DIP optim. (after pretraining (Barbano et al., 2022c))	<0.1 h
Hyperparam. optim. (MLL)	26.2 h
Hyperparam. optim. (TV-MAP)	35.4 h
Assemble Σ_{yy}	2.7 h
Draw 4096 posterior samples	2.4 h
- (Evaluate 4096 times $2\times$ $v_\theta^\top J^\top + 1\times$ $v_x^\top J$)	2.4 h
Draw 4096 posterior samples ($\tilde{J}$ & PCG)	0.3 h
- (Evaluate 4096 times $2\times$ $v_\theta^\top \tilde{J}^\top + 1\times$ $v_x^\top \tilde{J}$)	< 0.1 min

Algorithm 1 summarises image reconstruction and uncertainty estimation with the linearised DIP.

Algorithm 1: Linearised deep image prior (lin.-DIP) inference

Inputs: noisy measurements y, a CNN $x(\cdot)$, probabilistic model's hyperparameters, whether to use fast approximate posterior sampling *fast_sampling*

$\hat{\theta} \leftarrow$ `fit_DIP`$(y, x(\theta))$ `// by minimising eq. (5)`

$\hat{\theta}_h \leftarrow$`find_linearised_MAP`$(y, x(\hat{\theta}))$ `// using Algorithm 1 from Antorán et al. (2022)`

$\sigma_y^2, \{\sigma_d^2, \ell_d\}_{d=1}^D \leftarrow$ `optimise_hyperparams`$(y, x(\hat{\theta}), \hat{\theta}_h)$ `// by maximising eq. (17) with estimators eqs. (19) to (21) and solving linear systems with PCG`

if *not* *fast_sampling* **then**

  $\Sigma_{yy} \leftarrow$ `assemble_covariance`$(x(\hat{\theta}), \sigma_y^2, \{\sigma_d^2, \ell_d\}_{d=1}^D)$ `// by applying eq. (18) to rows of` I_{d_y}

  $\hat{\Sigma}_{x|y} \leftarrow$ `posterior_sampling`$(x(\hat{\theta}), \{\sigma_d^2, \ell_d\}_{d=1}^D, \Sigma_{yy})$ `// using eq. (22)`

else

  $\tilde{J} \leftarrow$ `construct_lowrank_Jacobian`$(x(\hat{\theta}))$ `// by randomised SVD, cf. section 6.4`

  $\hat{\Sigma}_{x|y} \leftarrow$ `fast_sampling`$(x(\hat{\theta}), \sigma_y^2, \{\sigma_d^2, \ell_d\}_{d=1}^D, \tilde{J})$ `// using eq. (22) with` $\tilde{J}$ `and PCG`

Output: mean reconstruction $x(\hat{\theta})$, posterior covariance estimate $\hat{\Sigma}_{x|y}$

7 Experiments

Here, we experimentally evaluate: i) the properties of the models and priors discussed in sections 4 and 5, and whether they lead to accurate reconstructions and calibrated uncertainty; ii) the fidelity of the approximations described in section 6; and iii) the performance of the proposed method linearised DIP (lin.-DIP) relative to the previous MC dropout (MCDO) based probabilistic formulation of DIP (Laves et al., 2020). We attempted to include DIP-SGLD (Cheng et al., 2019) in our analysis, but were unable to get the method to produce competitive results on tomographic reconstruction problems. For each individual image to be reconstructed, we employ the following linearised DIP inference procedure: i) optimise the DIP weights via eq. (5), obtaining $\hat{x} = x(\hat{\theta})$; ii) optimise prior hyperparameters (σ_y^2, ℓ, σ^2) via eq. (17); iii) assemble and Cholesky decompose Σ_{yy} with eq. (18) (this step can be accelerated using approximate methods sections 6.3 and 6.4); iv) compute posterior covariance matrices either via eq. (13), or estimate them via eq. (22); cf. Algorithm 1.

7.1 Small scale ablation analysis: reconstruction of KMNIST digits

The initial analysis uses simulated CT data obtained by applying eq. (1) to 50 images from the test set of the Kuzushiji-MNIST (KMNIST) dataset: 28×28 ($d_x = 784$) grayscale images of Hiragana characters (Clanuwat et al., 2018). We choose the noise standard deviation to be either 5% or 10% of the mean of $\mathrm{A}x$, denoted as $\eta(5\%)$ or $\eta(10\%)$. The forward map A is a discrete Radon transform, assembled via `ODL` (Adler et al., 2017). We use a U-Net with 3 scales and 76905 parameters (a down-sized net compared to the one in fig. 3).

7.1.1 Comparing linearised DIP with network-free priors

We first evaluate the priors in section 4, i.e. TV prior, TV-PredCP with a Matern-1/2 kernel, and a factorised Gaussian prior, and perform inference in the setting where the map A collects 5 angles ($d_y = 205$) sampled uniformly from 0° to 180° and is applied to 50 KMNIST test set images. Here, 10% noise is added. This results in a very ill-posed reconstruction problem, maximising the relevance of the prior. We select the σ_y^2 and λ hyperparameters for the factorised Gaussian prior and the TV prior respectively such that the posterior mean's PSNR is maximised across a validation set of 10 images from the KMNIST training set. We keep the choice of σ_y^2 and λ hyperparameters from the first two models for our experiments with the third model: Matern-1/2 with TV-PredCP prior over ℓ. For all priors, we perform inference with the NUTS HMC sampler. We run 5 independent chains for each image. We burn these in for 3×10^3 steps each and then proceed to draw 10^4 samples with a thinning factor of 2.

Table 2: Quantitative results for inference with the different priors introduced in section 4. We report both the PSNR of $\mathbb{E}[x|y]$, which denotes the posterior mean reconstruction, and the PSNR of $\hat{x}$, which denotes the posterior mode found through optimisation.

	log-likelihood	$\mathbb{E}[x\|y]$	$\hat{x}$
Fact. Gauss.	0.30 ± 0.17	16.15 ± 0.38	14.89 ± 0.38
TV	0.49 ± 0.14	16.32 ± 0.38	16.29 ± 0.41
TV-PredCP	$\mathbf{0.65 \pm 0.12}$	$\mathbf{16.55 \pm 0.39}$	$\mathbf{17.48 \pm 0.39}$
lin.-DIP (MLL)	1.63 ± 0.08	–	19.46 ± 0.52
lin.-DIP (TV-MAP)	1.63 ± 0.09	–	19.46 ± 0.52

We evaluate test log-likelihood using Gaussian Kernel Density Estimation (KDE) (Silverman, 1986). The kernel bandwidth is chosen using cross-validation on 10 images from the training set. The results in table 2 show that the TV-PredCP performs best in terms of the test log-likelihood and both posterior mean and posterior mode PSNR, followed by the TV and then the factorised Gaussian. This is somewhat surprising considering that this prior was designed as an approximation to the intractable TV prior. We hypothesise that this may be due to the Matern model allowing for faster transitions in the image than the TV prior, while still capturing local correlations, as shown qualitatively in fig. 2. This property may be well-suited to the KMNIST datasets, where most pixels either present large amplitudes or are close to 0. DIP-based predictions provide 2dB higher PSNR reconstructions than the non-DIP based priors, thus linearised DIP handily obtains a better test log-likelihood than the more-traditional methods.

7.1.2 Comparing calibration with DIP uncertainty quantification baselines

Using KMNIST, we construct test cases of different ill-posedness by simulating the observation y with four different angle sub-sampling settings for the linear operator A: 30 (d_y=1230), 20 (d_y=820), 10 (d_y=410) and 5 (d_y=205) angles are taken uniformly from the range 0° to 180°. We consider two noise configurations by

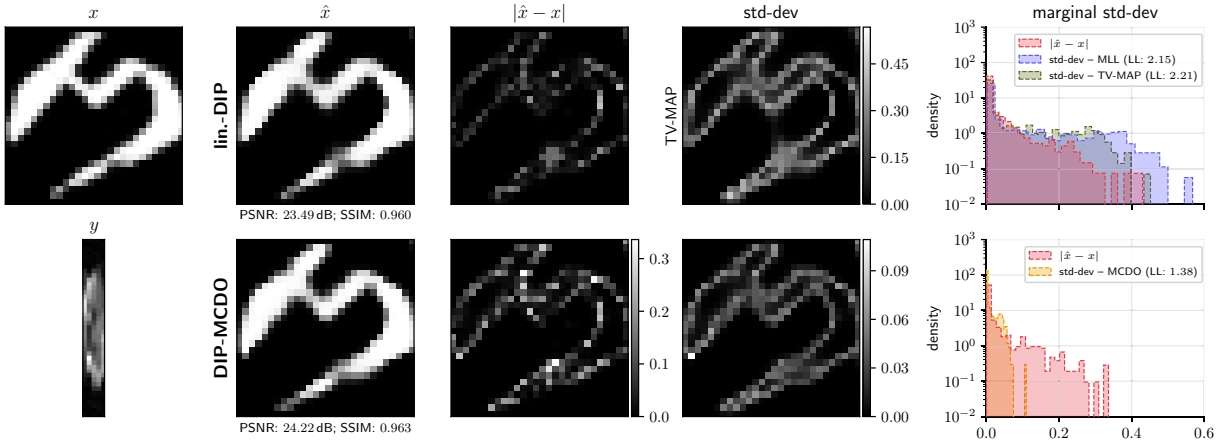

Figure 4: Exemplary character recovered from y (using 5 angles and $\eta(5\%)$) with lin.-DIP and DIP-MCDO along with respective uncertainty estimates. lin.-DIP provides vastly improved uncertainty calibration. For lin.-DIP, the colour-map is shared between $|\hat{x} - x|$ and std-dev, and TV-MAP refers to Type-II MAP optimisation of hyperparameters.

adding either 5% or 10% noise to the exact data $\mathrm{A}x$. We evaluate all DIP-based methods using the same 50 randomly chosen KMNIST test set images. To ensure a best-case showing of the methods, we choose appropriate hyperparameters for each number of angles and white noise percentage setting by applying grid-search cross-validation, using 50 images from the KMNIST training dataset. Specifically, we tune the TV strength λ and the number of optimisation iterations for the DIP. Due to the reduced image size, we apply linearised DIP as in section 5, without approximate computations. As an ablation study, we include additional baselines: linearised DIP without the TV-PredCP prior over hyperparameters (labelled MLL), and DIP reconstruction with a simple Gaussian noise model consisting of the back-projected observation noise $\mathcal{N}(x; \hat{x}, \sigma_{\mathrm{A}}^2 \mathrm{I})$, with $\sigma_{\mathrm{A}}^2 = \sigma_y^2 \mathrm{Tr}((\mathrm{A}^\top \mathrm{A})^\dagger) d_x^{-1}$ where σ_y^2=1 (labelled σ_y^2=1). Note that non-dropout methods share the same DIP parameters $\hat{\theta}$, and thus the same mean reconstruction. Hence, higher values in log-density indicate better uncertainty calibration, i.e. the predictive standard deviation better matches the empirical reconstruction error. DIP-MCDO does not provide an explicit likelihood function over the reconstructed image. We model its uncertainty with a Gaussian predictive distribution with covariance estimated from 2^{14} samples. MNIST images are quantised to 256 bins, but our models make predictions over continuous pixel values. Thus, we simulate a de-quantisation of KMNIST images by adding a noise jitter term of variance approximately matching that of a uniform distribution over the quantisation step (Hoogeboom et al., 2020).

Table 3: Mean and std-err of test log-likelihood computed over 50 KMNIST test images.

η (5%)	#angles: 5	10	20	30
DIP ($\sigma_y^2 = 1$)	0.68 ± 0.14	1.57 ± 0.02	1.85 ± 0.02	2.02 ± 0.02
DIP-MCDO	0.74 ± 0.13	1.60 ± 0.02	1.87 ± 0.02	2.05 ± 0.02
lin.-DIP (MLL)	**1.90 ± 0.14**	**2.57 ± 0.09**	**2.94 ± 0.10**	**3.09 ± 0.12**
lin.-DIP (TV-MAP)	**1.88 ± 0.15**	**2.59 ± 0.10**	**2.96 ± 0.10**	**3.11 ± 0.12**

η (10%)	#angles: 5	10	20	30
DIP ($\sigma_y^2 = 1$)	0.27 ± 0.17	1.31 ± 0.04	1.62 ± 0.03	1.76 ± 0.04
DIP-MCDO	0.42 ± 0.14	1.39 ± 0.04	1.70 ± 0.03	1.85 ± 0.04
lin.-DIP (MLL)	**1.63 ± 0.08**	**2.11 ± 0.07**	**2.43 ± 0.07**	**2.59 ± 0.08**
lin.-DIP (TV-MAP)	**1.63 ± 0.09**	**2.13 ± 0.07**	**2.45 ± 0.08**	**2.61 ± 0.08**

Table 4: PSNR [dB] / SSIM of the reconstruction posterior mean, averaged over 50 KMNIST test images.

η (5%)	#angles: 5	10	20	30
DIP	**21.42**/ **0.890**	27.92/ **0.977**	31.21/ **0.988**	32.93/ **0.991**
DIP-MCDO	20.95/0.882	**28.26**/ **0.977**	**31.65**/0.986	**33.45**/0.990

η (10%)	#angles: 5	10	20	30
DIP	**19.46**/ **0.846**	24.56/ **0.956**	27.27/ **0.974**	28.57/ **0.980**
DIP-MCDO	18.91/0.830	**24.76**/0.953	**27.72**/0.972	**29.09**/0.978

Table 3 shows the test log-likelihood for all the methods and experimental settings under consideration. The peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) index of posterior mean reconstructions are given in table 4. All methods show similar PSNR with the standard DIP (with TV regularisation) obtaining better PSNR in the very ill-posed setting (5 angles) and MCDO obtaining marginally better reconstruction in all others. Despite this, the linearised DIP provides significantly better uncertainty calibration, outperforming all baselines in terms of test log-likelihood in all settings. Figure 4 shows an exemplary character recovered from a simulated observation y (using 20 angles and 5% noise) with both linearised DIP and DIP-MCDO along with their associated uncertainty maps and calibration plots. DIP-MCDO systematically underestimates

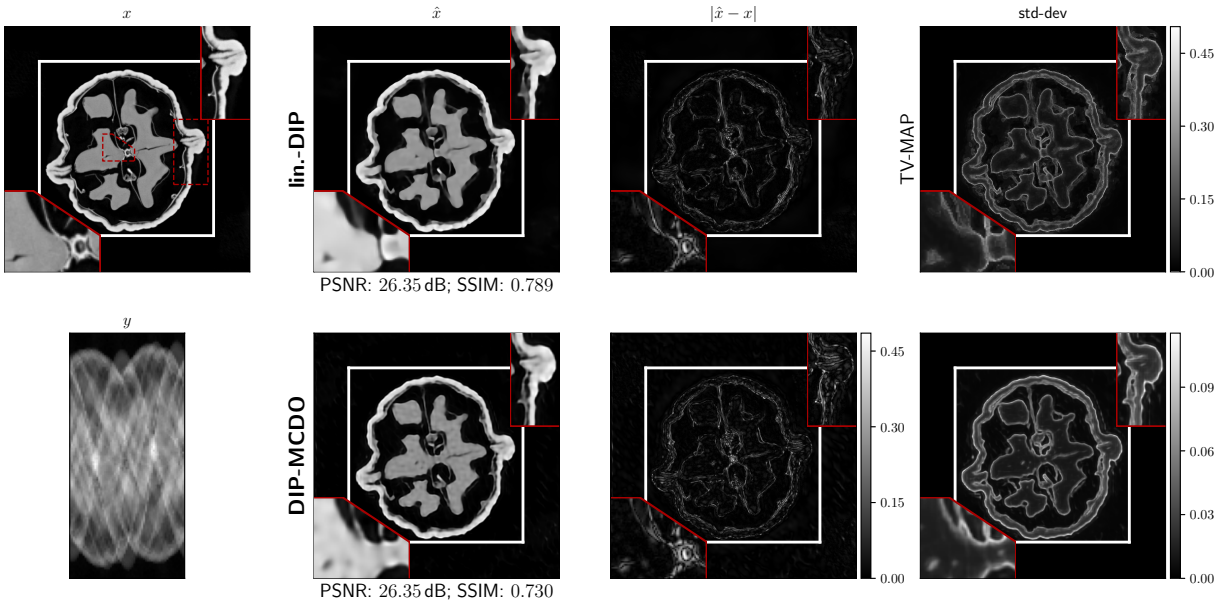

Figure 5: Reconstruction of a $501 \times 501\,\mathrm{px}^2$ slice of a scanned Walnut using lin.-DIP and DIP-MCDO along with their respective uncertainty estimates. The zoomed regions (outlined in red) are given in top-left.

uncertainty for pixels on which the error is large, explaining its poor test log-likelihood. The pixel-wise standard deviation provided by linearised DIP (TV-MAP) better correlates with the reconstruction error.

7.1.3 Evaluating the fidelity of sample-based predictive covariance matrix estimation

We evaluate the accuracy of the sampling, conjugate gradient and low rank approximations to the predictive covariance $\Sigma_{x|y}$ discussed in section 6. We compute the exact predictive covariance with eq. (13) as a reference, which is tractable for KMNIST, and do not use patch-based approximations or stabilised covariance estimators. Table 5 shows that estimating $\Sigma_{x|y}$ using samples does not decrease the performance. Using a low-rank approximation to J and computing linear solves with PCG lose at most 0.32 nats in test log-likelihood with respect to the exact one, but result in almost an order of magnitude speedup at prediction time.

Table 5: Evaluation of our approximate covariance estimation methods in terms of test log-likelihood over 10 KMNIST test images considering the 20 angle ($d_y = 820$) setting and using lin.-DIP (MLL).

η (%)	exact cov. eq. (13)	sampled cov. eq. (22)	sampled cov. ($\tilde{J}$) eq. (22)	sampled cov. ($\tilde{J}$ & PCG) eq. (22)
5	2.80 ± 0.06	2.80 ± 0.06	2.68 ± 0.09	2.62 ± 0.09
10	2.26 ± 0.06	2.26 ± 0.06	2.21 ± 0.06	2.22 ± 0.06

Table 6: Test log-likelihood, PSNR and structural similarity (SSIM) on the Walnut. We compare all lin.-DIP variants with DIP-MCDO.

	1×1	2×2	10×10	PSNR [dB]	SSIM
DIP-MCDO	0.03	1.68	2.47	23.49	0.730
lin.-DIP (MLL)	2.09	2.25	2.43	**26.35**	**0.789**
lin.-DIP (MLL, $\tilde{J}$ & PCG)	1.88	2.05	2.24	–	–
lin.-DIP (TV-MAP)	**2.21**	**2.40**	**2.60**	–	–
lin.-DIP (TV-MAP, $\tilde{J}$ & PCG)	**2.24**	**2.46**	**2.65**	–	–

7.2 Linearised DIP for high-resolution CT

We now demonstrate the approach on real-measured cone-beam μCT data of a walnut (Der Sarkissian et al., 2019). We reconstruct a $501 \times 501\,\mathrm{px}^2$ slice ($d_x = 251\,001$) using a sparse subset of measurements taken from 60 angles and 128 detector rows ($d_y = 7680$), using the U-Net in fig. 3 which has about 3 million parameters. Here, Σ_{xx} is too large to store in memory and Σ_{yy} too expensive to assemble repeatedly, and we use the full suite of approximations in section 6. Since the Walnut data is not quantised, jitter correction is not needed.

During MLL and Type-II MAP optimisation, many layers' prior variance goes to $\sigma_d^2 \approx 0$, cf. appendix D. This phenomenon is known as "automatic relevance determination" (Mackay, 1996; Tipping, 2001), and simplifies our linearised network, preventing uncertainty overestimation. We did not observe this effect when working with KMNIST images and smaller networks. We display the MLL and MAP optimisation profiles for the

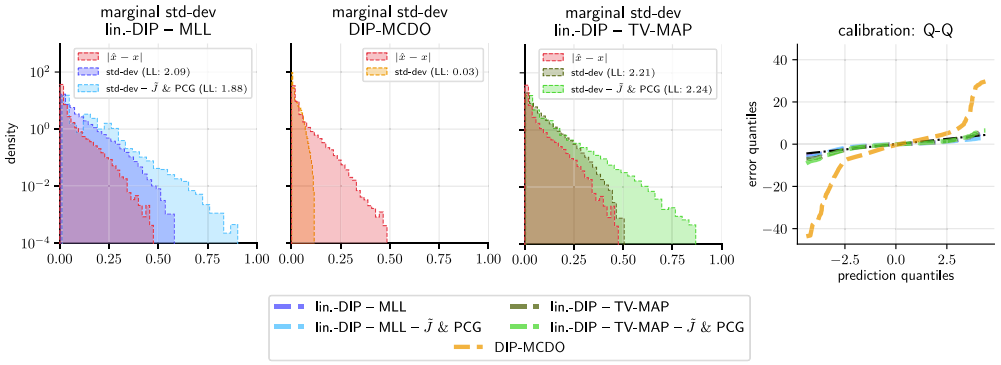

Figure 7: The comparison of uncertainty calibration: the pixel-wise error $|\hat{x} - x|$ overlaps with the uncertainties provided by the lin.-DIP. DIP-MCDO, instead, severely underestimates uncertainty. The scale of the pixel-wise standard deviation (std-dev) obtained including the TV-PredCP matches the absolute error more closely than when the hyperparameters are optimised without. Using $\tilde{J}$ & PCG results in overestimating uncertainty in the tails. LL stands for test log-likelihood.

active layers (i.e. layers with high σ_d^2) in fig. 6. Type-II MAP hyperparameters optimisation drives σ^2 to smaller values, compared to MLL. This restricts the linearised DIP prior, and thus the induced posterior, to functions that are smooth in a TV sense, leading to smaller error-bars, cf. fig. 7. As the optimisation of eq. (17) progresses, ℓ_1, ℓ_{11} fall into basins of new minima corresponding to larger lengthscales. This results in more correlated dimensions in the prior, further simplifying the model.

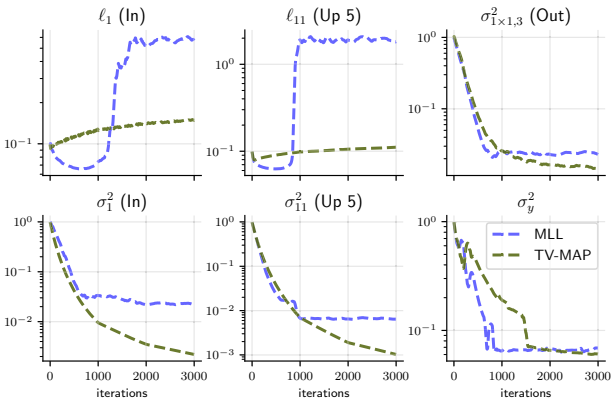

Figure 6: Optimisation trajectories for hyperparameters of the U-Net's first and last 3×3 convolutions $(\ell_1, \sigma_1^2, \ell_{11}, \sigma_{11}^2)$, last 1×1 convolution $(\sigma^2_{1\times1,3})$ and noise variance σ_y^2 for the Walnut data.

Density estimation described in section 6.3, is conducted in double precision (64 bit floating point) since single precision led to numerical instability in the assembly of Σ_{yy}, and also in the estimation of off-diagonal covariance terms for larger patches. In table 6, we report test log-likelihood computed using a Gaussian predictive distribution with covariance patches of sizes 1×1, 2×2 and 10×10 pixels. Mean reconstruction metrics are also reported. Figure 5 displays reconstructed images, uncertainty maps and calibration plots. In this more challenging task, DIP-MCDO performs poorly relative to the standard DIP formulation eq. (5) in terms of PSNR. DIP-MCDO underestimates uncertainty, and its uncertainty map is blurred across large sections of the image, placing large uncertainty in well-reconstructed regions and vice-versa. In contrast, the uncertainty map provided by linearised DIP is fine-grained, concentrating on regions of increased reconstruction error. Linearised DIP provides over 2.06 nats per pixel improvement in terms of test log-likelihood and more calibrated uncertainty estimates, as reflected in the Q-Q plot in fig. 7. Furthermore, the use of TV-PredCP prior for MAP optimisation yields a 0.12 nat per pixel improvement over the MLL approach. Interestingly, using low-rank Jacobians and PCG for sampling provides a small performance boost when using the TV-PredCP prior. Figure 7 reveals that these approximations result in uncertainty overestimation (a known issue (Antoran et al., 2023)) which is compensated by the more restrictive TV-PredCP prior.

8 Conclusion

We have proposed a probabilistic formulation of the deep image prior (DIP) that utilises a linearisation of the DIP network around the mode of the loss and a Gaussian-linear hierarchical prior on the network

parameters mimicking the total variation prior (constructed via the predictive complexity prior framework). The approach yields well-calibrated uncertainty estimates on tomographic reconstruction tasks based on simulated observations and real-measured μCT data. The empirical results suggest that both the DIP reparametrisation and the TV regulariser provide good inductive biases for high-quality reconstructions and well-calibrated uncertainty estimates. The method is shown to provide by far more calibrated uncertainty estimates than existing MC dropout approaches to uncertainty estimation with the DIP. However, this comes at a larger computational cost. Fortunately, since the first appearance of this work, Antoran et al. (2023) have developed techniques that reduce the cost of our linearised DIP inference by two orders of magnitude.

Acknowledgements

The authors would like to thank Shreays Padhy, Marine Schimel, Alexander Terenin, Eric Nalisnick, Erik Daxberger and James Allingham for fruitful discussions. R.B. acknowledges support from the i4health PhD studentship (UK EPSRC EP/S021930/1), and from The Alan Turing Institute (UK EPSRC EP/N510129/1). J.L. was funded by the German Research Foundation (DFG; GRK 2224/1) and by the Federal Ministry of Education and Research via the DELETO project (BMBF, project number 05M20LBB). The work of BJ is partially supported by UK EPSRC grant EP/V026259/1 and a start-up fund from The Chinese University of Hong Kong. JMHL acknowledges support from a Turing AI Fellowship EP/V023756/1 and an EPSRC Prosperity Partnership EP/T005386/1. JA acknowledges support from Microsoft Research, through its PhD Scholarship Programme, and from the EPSRC. This work has been performed using resources provided by the Cambridge Tier-2 system operated by the University of Cambridge Research Computing Service (http://www.hpc.cam.ac.uk) funded by EPSRC Tier-2 capital grant EP/T022159/1.

References

Jonas Adler, Holger Kohr, and Ozan Öktem. Operator discretization library (ODL). *Software available from https://github.com/odlgroup/odl*, 2017.

Javier Antorán. Understanding Uncertainty in Bayesian Neural Networks. Mphil in Machine Learning and Machine Intelligence Thesis, University of Cambridge, 2019.

Javier Antorán, James Allingham, and José Miguel Hernández-Lobato. Depth uncertainty in neural networks. *Advances in Neural Information Processing Systems*, 33:10620–10634, 2020.

Javier Antorán, Umang Bhatt, Tameem Adel, Adrian Weller, and José Miguel Hernández-Lobato. Getting a {clue}: A method for explaining uncertainty estimates. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=XSLF1XFq5h.

Javier Antorán, James Allingham, David Janz, Erik Daxberger, Eric Nalisnick, and José Miguel Hernández-Lobato. Linearised Laplace inference in networks with normalisation layers and the neural g-prior. Fourth Symposium on Advances in Approximate Bayesian Inference, AABI 2022, 2022.

Javier Antorán, David Janz, James Urquhart Allingham, Erik A. Daxberger, Riccardo Barbano, Eric T. Nalisnick, and José Miguel Hernández-Lobato. Adapting the linearised Laplace model evidence for modern deep learning. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvári, Gang Niu, and Sivan Sabato (eds.), *International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 796–821. PMLR, 2022. URL https://proceedings.mlr.press/v162/antoran22a.html.

Javier Antoran, Shreyas Padhy, Riccardo Barbano, Eric Nalisnick, David Janz, and José Miguel Hernández-Lobato. Sampling-based inference for large linear models, with application to linearised laplace. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=aoDyX6vSqsd.

Vegard Antun, Francesco Renna, Clarice Poon, Ben Adcock, and Anders C. Hansen. On instabilities of deep learning in image reconstruction and the potential costs of AI. *Proc. Nat. Acad. Sci.*, 117(48):30088–30095, 2020.

Simon Arridge, Peter Maaß, Ozan Öktem, and Carola-Bibiane Schönlieb. Solving inverse problems using data-driven models. *Acta Numer.*, 28:1–174, 2019.

Arsenii Ashukha, Alexander Lyzhov, Dmitry Molchanov, and Dmitry Vetrov. Pitfalls of in-domain uncertainty estimation and ensembling in deep learning. Preprint, arXiv:2002.06470, 2020.

Daniel Otero Baguer, Johannes Leuschner, and Maximilian Schmidt. Computed tomography reconstruction using deep image prior and learned reconstruction methods. *Inverse Problems*, 36(9):094004, 2020.

Riccardo Barbano, Simon Arridge, Bangti Jin, and Ryutaro Tanno. Uncertainty quantification for medical image synthesis. In *Biomedical Image Synthesis and Simulation: Methods and Applications*, pp. 601–641. Elsevier, 2021.

Riccardo Barbano, Javier Antorán, José Miguel Hernández-Lobato, and Bangti Jin. A probabilistic deep image prior over image space. In *Fourth Symposium on Advances in Approximate Bayesian Inference*, 2022a.

Riccardo Barbano, Johannes Leuschner, Javier Antorán, Bangti Jin, and José Miguel Hernández-Lobato. Bayesian experimental design for computed tomography with the linearised deep image prior. In *ICML2022 Workshop on Adaptive Experimental Design and Active Learning in the Real World*, 2022b.

Riccardo Barbano, Johannes Leuschner, Maximilian Schmidt, Alexander Denker, Andreas Hauptmann, Peter Maaß, and Bangti Jin. An educated warm start for deep image prior-based micro CT reconstruction. *IEEE Trans. Comput. Imag.*, 8:1210–1222, 2022c.

Riccardo Barbano, Javier Antorán, Johannes Leuschner, José Miguel Hernández-Lobato, Željko Kereta, and Bangti Jin. Fast and painless image reconstruction in deep image prior subspaces. Preprint, arXiv:2302.10279, 2023.

Semih Barutcu, Doğa Gürsoy, and Aggelos K. Katsaggelos. Compressive ptychography using deep image and generative priors. Preprint, arXiv:2205.02397, 2022.

Umang Bhatt, Javier Antorán, Yunfeng Zhang, Q. Vera Liao, Prasanna Sattigeri, Riccardo Fogliato, Gabrielle Gauthier Melançon, Ranganath Krishnan, Jason Stanley, Omesh Tickoo, Lama Nachman, Rumi Chunara, Madhulika Srikumar, Adrian Weller, and Alice Xiang. Uncertainty as a form of transparency: Measuring, communicating, and using uncertainty. In Marion Fourcade, Benjamin Kuipers, Seth Lazar, and Deirdre K. Mulligan (eds.), *AIES '21: AAAI/ACM Conference on AI, Ethics, and Society, Virtual Event, USA, May 19-21, 2021*, pp. 401–413. ACM, 2021. URL https://doi.org/10.1145/3461702.3462571.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural networks. In *Proceedings of the 32nd International Conference on Machine Learning, PMLR 37*, pp. 1613–1622, 2015.

Ashish Bora, Ajil Jalal, Eric Price, and Alexandros G. Dimakis. Compressed sensing using generative models. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 537–546, 2017.

Antonin Chambolle, Vicent Caselles, Daniel Cremers, Matteo Novaga, and Thomas Pock. An introduction to total variation for image analysis. In *Theoretical foundations and numerical methods for sparse recovery*, pp. 263–340. de Gruyter, 2010.

Zezhou Cheng, Matheus Gadelha, Subhransu Maji, and Daniel Sheldon. A Bayesian perspective on the deep image prior. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5443–5451. Computer Vision Foundation / IEEE, 2019. URL https://doi.org/10.1109/CVPR.2019.00559.

Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. Deep learning for classical Japanese literature. In *32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Workshop on Machine Learning for Creativity and Design*, 2018.

Jianan Cui, Kuang Gong, Ning Guo, Chenxi Wu, Kyungsang Kim, Huafeng Liu, and Quanzheng Li. Populational and individual information based PET image denoising using conditional unsupervised learning. *Phys. Med. & Biol.*, 66(15):155001, 2021.

Mohammad Zalbagi Darestani and Reinhard Heckel. Accelerated MRI with un-trained neural networks. *IEEE Trans. Comput. Imag.*, 7:724–733, 2021.

Erik Daxberger, Agustinus Kristiadi, Alexander Immer, Runa Eschenhagen, Matthias Bauer, and Philipp Hennig. Laplace redux - effortless Bayesian deep learning. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems 34*, pp. 20089–20103, 2021a. URL https://proceedings.neurips.cc/paper/2021/hash/a7c9585703d275249f30a088cebba0ad-Abstract.html.

Erik Daxberger, Eric Nalisnick, James U. Allingham, Javier Antorán, and Jose Miguel Hernandez-Lobato. Bayesian deep learning via subnetwork inference. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, pp. 2510–2521, 2021b. URL https://proceedings.mlr.press/v139/daxberger21a.html.

Henri Der Sarkissian, Felix Lucka, Maureen van Eijnatten, Giulia Colacicco, Sophia Bethany Coban, and K. Joost Batenburg. Cone-Beam X-Ray CT Data Collection Designed for Machine Learning: Samples 1-8, 2019. URL https://doi.org/10.5281/zenodo.2686726. *Zenodo*.

Kun Dong, David Eriksson, Hannes Nickisch, David Bindel, and Andrew Gordon Wilson. Scalable log determinants for Gaussian process kernel learning. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 30*, pp. 6327–6337, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/976abf49974d4686f87192efa0513ae0-Abstract.html.

Heinz W. Engl, Martin Hanke, and Andreas Neubauer. *Regularization of Inverse Problems*. Kluwer, Dordrecht, 1996. ISBN 0-7923-4157-0.

Andrew Y. K. Foong, David R. Burt, Yingzhen Li, and Richard E. Turner. On the expressiveness of approximate inference in Bayesian neural networks. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin (eds.), *Advances in Neural Information Processing Systems 33*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/b6dfd41875bc090bd31d0b1740eb5b1b-Abstract.html.

Vincent Fortuin, Adrià Garriga-Alonso, Florian Wenzel, Gunnar Ratsch, Richard E. Turner, Mark van der Wilk, and Laurence Aitchison. Bayesian neural network priors revisited. In *Third Symposium on Advances in Approximate Bayesian Inference*, 2021. URL https://openreview.net/forum?id=xaqKWHcoOGP.

Jacob R. Gardner, Geoff Pleiss, Kilian Q. Weinberger, David Bindel, and Andrew Gordon Wilson. GPyTorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 31*, pp. 7587–7597, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/27e8e17134dd7083b050476733207ea1-Abstract.html.

Adrià Garriga-Alonso, Laurence Aitchison, and Carl Edward Rasmussen. Deep convolutional networks as shallow Gaussian processes. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bklfsi0cKm.

Mark N. Gibbs and David J. C. MacKay. Efficient implementation of Gaussian processes for interpolation. http://www.inference.phy.cam.ac.uk/mackay/abstracts/gpros.html, 1996.

Kuang Gong, Ciprian Catana, Jinyi Qi, and Quanzheng Li. PET image reconstruction using deep image prior. *IEEE Trans. Med. Imag.*, 38(7):1655–1665, 2019.

Peter Guttorp and Tilmann Gneiting. On the Whittle-Matérn correlation family. NRCSE Technical Report No. 80, University of Washington, 01 2005.

Nathan Halko, Per-Gunnar Martinsson, and Joel A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.*, 53(2):217–288, 2011. URL https://doi.org/10.1137/090771806.

Horace He and Richard Zou. functorch: JAX-like composable function transforms for PyTorch. https://github.com/pytorch/functorch, 2021.

Reinhard Heckel and Paul Hand. Deep decoder: Concise image representations from untrained non-convolutional networks. In *ICLR*, 2019. URL https://openreview.net/pdf?id=rylV-2C9KQ.

Tapio Helin, Nuutti Hyvönen, and Juha-Pekka Puska. Edge-promoting adaptive Bayesian experimental design for x-ray imaging. *SIAM J. Sci. Comput.*, 44(3):B506–B530, 2022. URL https://doi.org/10.1137/21m1409330.

Emiel Hoogeboom, Taco S. Cohen, and Jakub M. Tomczak. Learning discrete distributions by dequantization. Preprint, arXiv:2001.11235, 2020.

Jiri Hron, Alex Matthews, and Zoubin Ghahramani. Variational Bayesian dropout: pitfalls and fixes. In Jennifer Dy and Andreas Krause (eds.), *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 2019–2028. PMLR, 10–15 Jul 2018.

Alexander Immer, Matthias Bauer, Vincent Fortuin, Gunnar Rätsch, and Mohammad Emtiyaz Khan. Scalable marginal likelihood estimation for model selection in deep learning. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 4563–4573. PMLR, 2021a. URL http://proceedings.mlr.press/v139/immer21a.html.

Alexander Immer, Maciej Korzepa, and Matthias Bauer. Improving predictions of Bayesian neural nets via local linearization. In Arindam Banerjee and Kenji Fukumizu (eds.), *The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 703–711. PMLR, 2021b. URL http://proceedings.mlr.press/v130/immer21a.html.

Kazufumi Ito and Bangti Jin. *Inverse Problems: Tikhonov Theory and Algorithms*, volume 22. World Scientific, 2014.

Arthur Jacot, Clément Hongler, and Franck Gabriel. Neural tangent kernel: Convergence and generalization in neural networks. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 31*, pp. 8580–8589, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/5a4be1fa34e62bb8a6ec6b91d2462f5a-Abstract.html.

Jari Kaipio and Erkki Somersalo. *Statistical and Computational Inverse Problems*. Springer-Verlag, New York, 2005. ISBN 0-387-22073-9.

Mohammad Emtiyaz Khan, Alexander Immer, Ehsan Abedi, and Maciej Korzepa. Approximate inference turns deep networks into Gaussian processes. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32*, pp. 3088–3098, 2019.

Tobias Knopp and Micro Grosser. Warmstart approach for accelerating deep image prior reconstruction in dynamic tomography. Proceedings of Machine Learning Research, Medical Imaging with Deep Learning 2022, 13 pp., 2022.

Benjamin Kompa, Jasper Snoek, and Andrew L. Beam. Second opinion needed: communicating uncertainty in medical machine learning. *NPJ Digital Medicine*, 4(1):1–6, 2021.

Max-Heinrich Laves, Malte Tölle, and Tobias Ortmaier. Uncertainty estimation in medical image denoising with Bayesian deep image prior. In *Uncertainty for Safe Utilization of Machine Learning in Medical Imaging, and Graphs in Biomedical Image Analysis*, pp. 81–96. Springer, 2020.

Jaehoon Lee, Lechao Xiao, Samuel S. Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32*, pp. 8570–8581, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/0d1a9651497a38d8b1c3871c84528bd4-Abstract.html.

F. C. Leone, L. S. Nelson, and R. B. Nottingham. The folded normal distribution. *Technometrics*, 3:543–550, 1961. ISSN 0040-1706. URL https://doi.org/10.2307/1266560.

Jiaming Liu, Yu Sun, Xiaojian Xu, and Ulugbek S Kamilov. Image restoration using total variation regularized deep image prior. In *ICASSP 2019*, 2019. URL https://doi.org/10.1109/ICASSP.2019.8682856.

David J. C. Mackay. Bayesian non-linear modeling for prediction competition. In *Maximum Entropy and Bayesian Methods*, pp. 221–234, 1996.

David John Cameron Mackay. *Bayesian Methods for Adaptive Models*. PhD thesis, California Institute of Technology, California, USA, 1992.

Wesley Maddox, Shuai Tang, Pablo Garcia Moreno, Andrew Gordon Wilson, and Andreas C. Damianou. Fast adaptation with linearized neural networks. In Arindam Banerjee and Kenji Fukumizu (eds.), *The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 2737–2745. PMLR, 2021. URL http://proceedings.mlr.press/v130/maddox21a.html.

Wesley J. Maddox, Pavel Izmailov, Timur Garipov, Dmitry P. Vetrov, and Andrew Gordon Wilson. A simple baseline for Bayesian uncertainty in deep learning. *Advances in Neural Information Processing Systems*, 32, 2019.

Per-Gunnar Martinsson and Joel A. Tropp. Randomized numerical linear algebra: Foundations and algorithms. *Acta Numer.*, 29:403–572, 2020.

Kenneth O. McGraw and S. P. Wong. The descriptive use of absolute differences between pairs of scores with a common mean and variance. *J. Educat. Stat.*, 19(2):103–110, 1994.

Eric T. Nalisnick, Jonathan Gordon, and José Miguel Hernández-Lobato. Predictive complexity priors. In Arindam Banerjee and Kenji Fukumizu (eds.), *The 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 694–702. PMLR, 2021. URL http://proceedings.mlr.press/v130/nalisnick21a.html.

Roman Novak, Lechao Xiao, Yasaman Bahri, Jaehoon Lee, Greg Yang, Jiri Hron, Daniel A. Abolafia, Jeffrey Pennington, and Jascha Sohl-Dickstein. Bayesian deep convolutional networks with many channels are Gaussian processes. In *7th International Conference on Learning Representations*. OpenReview.net, 2019. URL https://openreview.net/forum?id=B1g30j0qF7.

Roman Novak, Lechao Xiao, Jiri Hron, Jaehoon Lee, Alexander A. Alemi, Jascha Sohl-Dickstein, and Samuel S. Schoenholz. Neural tangents: Fast and easy infinite neural networks in Python. In *8th International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=SklD9yrFPS.

Gregory Ongie, Ajil Jalal, Richard G. Baraniuk, Christopher A. Metzler, Alexandros G. Dimakis, and Rebecca Willett. Deep learning techniques for inverse problems in imaging. *IEEE J. Sel. Areas Inform. Theory*, pp. 39–56, 2020.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2005. ISBN 026218253X.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pp. 234–241, 2015.

Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D*, 60(1-4):259–268, 1992.

Walter Rudin. *Fourier Analysis on Groups*. John-Wiley, New York-London, 1990.

Matthias W. Seeger and Hannes Nickisch. Large scale Bayesian inference and experimental design for sparse linear models. *SIAM J. Imaging Sci.*, 4(1):166–199, 2011. URL https://doi.org/10.1137/090758775.

Bernard W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman & Hall, London, 1986. ISBN 0-412-24620-1.

Jasper Snoek, Yaniv Ovadia, Emily Fertig, Balaji Lakshminarayanan, Sebastian Nowozin, D. Sculley, Joshua V. Dillon, Jie Ren, and Zachary Nado. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett (eds.), *Advances in Neural Information Processing Systems 32*, pp. 13969–13980, 2019. URL https://proceedings.neurips.cc/paper/2019/hash/8558cb408c1d76621371888657d2eb1d-Abstract.html.

Andrew M. Stuart. Inverse problems: a Bayesian perspective. *Acta Numer.*, 19:451–559, 2010. ISSN 0962-4929. URL https://doi.org/10.1017/S0962492910000061.

Andrey N. Tikhonov and Vasiliy Y. Arsenin. *Solutions of Ill-posed Problems*. John Wiley & Sons, New York-Toronto, Ont.-London, 1977.

Michael E. Tipping. Sparse Bayesian learning and the relevance vector machine. *J. Mach. Learn. Res.*, pp. 211–244, 2001.

Malte Tölle, Max-Heinrich Laves, and Alexander Schlaefer. A mean-field variational inference approach to deep image prior for inverse problems in medical imaging. In Mattias P. Heinrich, Qi Dou, Marleen de Bruijne, Jan Lellmann, Alexander Schlaefer, and Floris Ernst (eds.), *Medical Imaging with Deep Learning, 7-9 July 2021, Lübeck, Germany*, volume 143 of *Proceedings of Machine Learning Research*, pp. 745–760. PMLR, 2021. URL https://proceedings.mlr.press/v143/tolle21a.html.

Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9446–9454, 2018.

Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. *Int. J. Comput. Vis.*, 128(7): 1867–1888, 2020. ISSN 1573-1405. URL https://doi.org/10.1007/s11263-020-01303-4.

Wim van Aarle, Willem Jan Palenstijn, Jan De Beenhouwer, Thomas Altantzis, Sara Bals, K. Joost Batenburg, and Jan Sijbers. The ASTRA Toolbox: A platform for advanced algorithm development in electron tomography. *Ultramicroscopy*, 157:35–47, 2015. URL https://doi.org/10.1016/j.ultramic.2015.05.002.

Francisca Vasconcelos, Bobby He, Nalini Singh, and Yee Whye Teh. UncertaINR: Uncertainty quantification of end-to-end implicit neural representations for computed tomography. Preprint, arXiv:2202.10847, 2022.

Ge Wang, Jong Chul Ye, and Bruno De Man. Deep learning for tomographic image reconstruction. *Nature Mach. Intell.*, 2(12):737–748, 2020.

Hengkang Wang, Taihui Li, Zhong Zhuang, Tiancong Chen, Hengyue Liang, and Ju Sun. Early stopping for deep image prior. Preprint, arXiv:2112.06074, 2021.

Max Welling and Yee Whye Teh. Bayesian learning via stochastic gradient Langevin dynamics. In Lise Getoor and Tobias Scheffer (eds.), *Proceedings of the 28th International Conference on Machine Learning*, pp. 681–688. Omnipress, 2011. URL https://icml.cc/2011/papers/398_icmlpaper.pdf.

James T. Wilson, Viacheslav Borovitskiy, Alexander Terenin, Peter Mostowsky, and Marc Peter Deisenroth. Pathwise conditioning of Gaussian processes. *J. Mach. Learn. Res.*, 22:105:1–105:47, 2021. URL http://jmlr.org/papers/v22/20-1260.html.

A Designing total variation priors

To develop a probabilistic DIP, we describe first how to design a tractable TV prior. We reinterpret the TV regulariser eq. (2) as a prior over images, favouring those with low ℓ^1 norm gradients

$$p(x) = Z_\lambda^{-1} \exp(-\lambda \mathrm{TV}(x)), \tag{23}$$

where $Z_\lambda = \int \exp(-\lambda \mathrm{TV}(x))\, \mathrm{d}x$. This prior is intractable because Z_λ does not admit a closed form; thus approximations are necessary. We now explore alternatives without this limitation.

A.1 Further discussion on the TV regulariser as a prior

It is tempting to think that we do not need the PredCP machinery in section 5.3 to translate the TV regulariser into the parameter space. Indeed, the Laplace method simply involves a quadratic approximation around a mode of the log posterior, without placing any requirements on the prior used to induce said posterior. Hence, we can decompose the Hessian of the log posterior $\log p(\theta|y)$ into the contributions from the likelihood and the prior as

$$\frac{\partial}{\partial \theta^2} \left(\log p(y|\mathrm{A}x(\theta)) + \log p(x(\theta)) \right) |_{\theta=\hat{\theta}}$$

and realise that the log of the anisotropic TV prior $p(x) \propto \exp(-\lambda \mathrm{TV}(x))$ as in eq. (23) is only once differentiable. Ignoring the origin (where the absolute value function is non-differentiable), we obtain:

$$\frac{\partial}{\partial \theta^2} \log p(x(\theta))|_{\theta=\hat{\theta}} \propto -\frac{\partial}{\partial \theta^2} \mathrm{TV}(x(\theta))|_{\theta=\hat{\theta}} = 0.$$

Thus, a naive application of the Laplace approximation would eliminate the effect of the prior, leaving the posterior ill defined. In practice, one may smooth the non-smooth region around the origin, but the amount of smoothing can significantly influence the behaviour of the Hessian approximation.

A.2 Further discussion on inducing TV-smoothness with Gaussian priors

A standard alternative to enforce local smoothness in an image is to adopt a Gaussian prior $p(x) = \mathcal{N}(x; \mu, \Sigma_{xx})$ with covariance $\Sigma_{xx} \in \mathbb{R}^{d_x \times d_x}$ given by

$$[\Sigma_{xx}]_{ij,i'j'} = \sigma^2 \exp\left(\frac{-\mathrm{d}(i-i', j-j')}{\ell} \right), \tag{24}$$

where i, j index the spatial locations of pixels of x, as in eq. (2), and $\mathrm{d}(a, b) = \sqrt{a^2 + b^2}$ denotes the Euclidean vector norm. Equation (24) is also known as the Matern-$1/2$ kernel and matches the covariance of Brownian motion (Guttorp & Gneiting, 2005). The hyperparameter $\sigma^2 \in \mathbb{R}^+$ informs the pixel amplitude while the lengthscale parameter $\ell \in \mathbb{R}^+$ determines the correlation strength between nearby pixels. The TV in eq. (2) only depends on pixel pairs separated by one pixel ($\mathrm{d} = 1$), allowing analytical computation of the expected TV associated with the Gaussian prior

$$\kappa := \mathbb{E}_{x \sim \mathcal{N}(\mu, \Sigma_{xx})}[\mathrm{TV}(x)] = c\sqrt{\sigma^2(1-\rho)}, \tag{25}$$

with the correlation coefficient $\rho = \exp(-\ell^{-1}) \in (0, 1)$ and $c = {}^{4\sqrt{d_x}(\sqrt{d_x}-1)}/\sqrt{\pi}$ for square images. See appendix A.3 for derivations. Increasing ℓ (for a fixed σ^2) favours x with low TV on average, resulting in smoother images. The prior $\mathcal{N}(x; \mu, \Sigma_{xx})$ is conjugate to the likelihood implied by the least-square fidelity $\mathcal{N}(y; \mathrm{A}x, \sigma_y^2 \mathrm{I})$, leading to a closed form posterior predictive distribution and marginal likelihood objective with costs $\mathcal{O}(d_y^3)$ and $\mathcal{O}(d_y^2 d_x)$, respectively.

A.3 Derivation of the identity eq. (9)

The identity follows from the following result (appendix, (McGraw & Wong, 1994)). The short proof is recalled for the convenience of the reader.

Lemma A.1. *Let X and Y be normal random variables with mean μ, variance σ^2 and correlation coefficient ρ. Let $Z = |X - Y|$. Then*

$$\mathbb{E}[Z] = \frac{2}{\sqrt{\pi}}\sqrt{\sigma^2(1-\rho)}.$$

Proof. Clearly, $X - Y$ follows a Gaussian distribution with mean 0 and variance $2\sigma^2(1-\rho)$. Then the random variable

$$W = \frac{Z^2}{2\sigma^2(1-\rho)} = \Big(\frac{X-Y}{\sqrt{2\sigma^2(1-\rho)}}\Big)^2$$

follows χ^2_1 distribution. Then

$$\mathbb{E}[\sqrt{W}] = \int_0^\infty W^{\frac{1}{2}} \frac{1}{\Gamma(\frac{1}{2})\sqrt{2}} W^{\frac{1}{2}-1} e^{-\frac{W}{2}} \mathrm{d}W = \frac{\sqrt{2}}{\Gamma(\frac{1}{2})} = \frac{\sqrt{2}}{\sqrt{\pi}},$$

where $\Gamma(z)$ denotes the Euler's Gamma function, with $\Gamma(\frac{1}{2}) = \sqrt{\pi}$. Then it follows that

$$\mathbb{E}[Z] = \sqrt{2\sigma^2(1-\rho)}\mathbb{E}[\sqrt{W}] = \frac{2}{\sqrt{\pi}}\sqrt{\sigma^2(1-\rho)}.$$

This shows the assertion in the lemma. □

Now by the marginalisation property of multivariate Gaussians, any two neighbouring pixels of x for $x \sim \mathcal{N}(\mu, \Sigma_{xx})$ satisfy the conditions of Lemma A.1, with $\rho = \exp(-\ell^{-1}) \in (0, 1)$. Thus Lemma A.1 and the trivial fact $d_x = h \times w$ imply

$$\kappa_d = \mathbb{E}_{\mathcal{N}(x;\mu,\Sigma_{xx})}[\mathrm{TV}(x)] = \frac{2[2hw - h - w]}{\sqrt{\pi}}\sqrt{\sigma^2(1-\rho)}.$$

In particular, for a square image, $h = w = \sqrt{d_x}$, we obtain the desired identity in eq. (9).

B Derivation of the linearised deep image prior

B.1 Posterior predictive covariance

We provide an alternative derivation of the posterior predictive covariance of the linearised DIP by reasoning in the parameter space. First we have linearised the neural network $x(\theta)$, turning it into a Bayesian basis function linear model (Khan et al., 2019). The probabilistic model in eq. (12) is thus:

$$y|\theta \sim \mathcal{N}(\mathrm{A}h(\theta), \sigma_y^2\mathrm{I}), \quad \theta|\ell \sim \mathcal{N}\left(0, \Sigma_{\theta\theta}\right),$$

and the linearised Laplace approximate posterior distribution over weights is given by Immer et al. (2021b)

$$p(\theta|y) \approx \mathcal{N}(\theta;\, \hat{\theta}, \Sigma_{\theta|y}) \quad \text{with} \quad \Sigma_{\theta|y} = \left(\sigma_y^{-2} J^\top \mathrm{A}^\top \mathrm{A} J + \Sigma_{\theta\theta}^{-1}\right)^{-1}. \tag{26}$$

In this work we exploit the equivalence between basis function linear models and Gaussian Processes (GP), and perform inference using the dual GP formulation. This is advantageous due to its lower computational cost when $d_\theta >> d_y$, which is common in tomographic reconstruction.

We switch to the dual formulation using the SMW matrix inversion identity, we have

$$\Sigma_{\theta|y} = \left(\sigma_y^{-2} J^\top \mathrm{A}^\top \mathrm{A} J + \Sigma_{\theta\theta}^{-1}\right)^{-1} = \Sigma_{\theta\theta} - \Sigma_{\theta\theta} J^\top \mathrm{A}^\top (\sigma_y^2 \mathrm{I} + \mathrm{A} J \Sigma_{\theta\theta} J^\top \mathrm{A}^\top)^{-1} \mathrm{A} J \Sigma_{\theta\theta} \tag{27}$$

The predictive distribution over images can be built by marginalising the NN parameters in the conditional likelihood $p(x|y) = \int p(x|\theta)p(\theta|y)\,\mathrm{d}\theta$. Since $h(\cdot)$ is a deterministic function, we have $p(x|\theta) = \delta(x - h(\theta))$ and

$$\int p(x|\theta)p(\theta|y)\,\mathrm{d}\theta = \int \delta(x - h(\theta))\mathcal{N}(\theta; \hat{\theta}, \Sigma_{\theta|y})\,\mathrm{d}\theta = \mathcal{N}(x; \hat{x}, J\Sigma_{\theta|y}J^\top).$$

Note that this assumes $\hat{\theta}$ to be a mode of the DIP training loss eq. (5). In practise, this will not be satisfied and thus the posterior mean of the linear model $\hat{\theta}_h$, which is given as the minima of the linear model's loss introduced in section 5.4, will not match that of the NN, that is, $\hat{\theta}$. Using the linear model's exact mode is only necessary for the purpose of constructing the marginal likelihood objective (Antorán et al., 2022; Antorán et al., 2022) (see also appendix B.2). However, for the purpose of making predictions, assuming $\hat{\theta}$ to be the mode allows us to keep the DIP reconstruction $\hat{x}$ as the predictive mean.

B.2 Laplace marginal likelihood and Type-II MAP in eq. (17)

For the purpose of uncertainty estimation, we tune the hyperparameters of our linear model using the marginal likelihood of the conditional-on-ℓ Gaussian-linear model introduced in eq. (6). The posterior mode of the TV-regularised linearised model is given by $\hat{\theta}_h = \text{argmin}_{\theta_h} \sigma_y^{-2}\|\text{A}h(\theta_h) - y\| + \lambda\text{TV}(h(\theta_h))$. However, we substitute the TV with a multivariate Gaussian surrogate $p(\theta|\ell)$. Now we derive the marginal log-likelihood (MLL) for the linearised model conditional on ℓ following Antorán et al. (2022). In Bayes rule

$$\log p(\theta|y, \ell; \sigma_y^2, \sigma^2) = \log p(y|\theta; \sigma_y^2) + \log p(\theta|\ell; \sigma^2) - \log\, p(y|\ell; \sigma_y^2, \sigma^2),$$

we isolate the MLL $\log\, p(y|\ell; \sigma_y^2, \sigma^2)$, evaluate at the linear model's posterior mode $\theta = \hat{\theta}_h$ and obtain

$$\log p(y|\ell; \sigma_y^2, \sigma^2) = \log p(y|\theta{=}\hat{\theta}_h; \sigma_y^2) + \log p(\theta{=}\hat{\theta}_h|\ell; \sigma^2) - \log p(\theta{=}\hat{\theta}_h|y, \ell; \sigma_y^2, \sigma^2). \tag{28}$$

The log-density $\log p(y|\theta = \hat{\theta}_h; \sigma_y^2)$ quantifies the quality of the model's fit to the data y, and is given by

$$\log p(y|\theta = \hat{\theta}_h; \sigma_y^2) = -\frac{d_y}{2}\log(2\pi) - \frac{1}{2}\log|\sigma_y^2\text{I}| - \frac{1}{2\sigma_y^2}\|y - Ah(\hat{\theta}_h)\|_2^2.$$

However, since our predictive mode is given by the DIP reconstruction and not the linear model's reconstruction, we depart from the exact expression for the linear model's MLL and use $-\frac{d_y}{2}\log(2\pi) - \frac{1}{2}\log|\sigma_y^2\text{I}| - \frac{1}{2\sigma_y^2}\|y - \text{A}x(\hat{\theta})\|_2^2$ as the data fit term instead. The weight-mode log prior density $\log p(\theta{=}\hat{\theta}_h|\ell, \sigma^2)$ is given by

$$\log p(\theta{=}\hat{\theta}_h|\ell, \sigma^2) = -\frac{d_\theta}{2}\log(2\pi) - \frac{1}{2}\log|\Sigma_{\theta\theta|}| - \frac{1}{2}\hat{\theta}_h^\top\Sigma_{\theta\theta}^{-1}\hat{\theta}_h.$$

Evaluating the Gaussian posterior log density over θ at its mode $\hat{\theta}_h$ cancels the exponent of the Gaussian and leaves us with just the normalising constant

$$\log p(\theta{=}\hat{\theta}_h|y, \ell; \sigma_y^2, \sigma^2) = -\frac{1}{2}\log|\Sigma_{\theta|y}| - \frac{d_\theta}{2}\log(2\pi)$$

By the matrix determinant lemma, the determinant $|\Sigma_{\theta|y}|$ is given by

$$|\Sigma_{\theta|y}| = |\sigma_y^{-2}J^\top\text{A}^\top\text{A}J + \Sigma_{\theta\theta}^{-1}|^{-1} = |\text{A}J\Sigma_{\theta\theta}J^\top\text{A}^\top + \sigma_y^2\text{I}|^{-1}|\Sigma_{\theta\theta}||\sigma_y^2\text{I}|. \tag{29}$$

Thus, the linearised Laplace marginal likelihood is given by

$$\begin{aligned}
\log p(y|\ell; \sigma_y^2, \sigma^2) =& -\frac{1}{2}\log|\sigma_y^2\text{I}| - \frac{1}{2\sigma_y^2}\|y - \text{A}x(\hat{\theta})\|_2^2 - \frac{1}{2}\log|\Sigma_{\theta\theta}| - \frac{1}{2}\hat{\theta}_h^\top\Sigma_{\theta\theta}^{-1}\hat{\theta}_h \\
&- \frac{1}{2}\log|\text{A}J\Sigma_{\theta\theta}J^\top\text{A}^\top + \sigma_y^2\text{I}| + \frac{1}{2}\log|\Sigma_{\theta\theta}| + \frac{1}{2}\log|\sigma_y^2\text{I}| + C \\
=& -\frac{1}{2\sigma_y^2}||y - \text{A}x(\hat{\theta})||_2^2 - \frac{1}{2}\hat{\theta}_h^\top\Sigma_{\theta\theta}^{-1}\hat{\theta}_h - \frac{1}{2}\log|\text{A}J\Sigma_{\theta\theta}J^\top\text{A}^\top + \sigma_y^2\text{I}| + C
\end{aligned} \tag{30}$$

where C captures all terms constant in $(\sigma_y^2, \ell, \sigma^2)$. Recall that $\Sigma_{yy} = \text{A}J\Sigma_{\theta\theta}J^\top\text{A}^\top + \sigma_y^2\text{I}$. Next we turn to the TV-PredCP prior over ℓ

$$\log p(\ell; \sigma^2) = -\sum_{d=1}^{D}\kappa_d + \log\left|\frac{\partial\kappa_d}{\partial\ell_d}\right|, \text{ with } \kappa_d := \mathbb{E}_{\mathcal{N}(\hat{\theta}_d, \Sigma_{\theta_d\theta_d})\prod_{i=1, i\neq d}^{D}\delta(\theta_i - \hat{\theta}_i)}\left[\lambda\text{TV}(h(\theta))\right].$$

Hence we obtain the following Type-II maximum a posteriori (MAP)-style objective:

$$\begin{aligned}&\log p(y, \ell; \sigma_y^2, \sigma^2) \approx \log \mathcal{N}(y; 0, \Sigma_{yy}) + \log p(\ell; \sigma^2) \\ =&\frac{1}{2}\left(-\sigma_y^{-2}||y - \mathrm{A}x(\hat{\theta})||_2^2 - \hat{\theta}_h^\top \Sigma_{\theta\theta}^{-1}\hat{\theta}_h - \log|\Sigma_{yy}|\right) - \sum_{d=1}^{D}\kappa_d + \log\left|\frac{\partial \kappa_d}{\partial \ell_d}\right| + C.\end{aligned}$$

C Additional details on our TV-PredCP

C.1 Correspondence to the formulation of Nalisnick et al. (2021)

The original formulation of the TV-PredCP (Nalisnick et al., 2021) defines a base model $q(x) = p(x|a = a_0)$ and an extended model $p(x) = p(x|a = \tau)$. The (hyper)parameter τ determines how much the predictions of the two models vary. A divergence $\mathcal{D}(p(x|a = a_0)||p(x|a = \tau))$ is placed between the two distributions and a prior placed over the divergence. This divergence is mapped back to the parameter τ using the change of variables formula. To see how our approach eq. (10) falls within this setup, take $p(x|a = \tau)$ to be $p(x) = \mathcal{N}(x; \mu, \Sigma_{xx}(\sigma^2, \ell))$, where the lengthscale ℓ takes the place of τ. The base model sets the lengthscale to be infinite, or equivalently the correlation coefficient ρ to be 1, $q(x) = \mathcal{N}(x; \mu, \Sigma_{xx}(\sigma_x^2, \infty))$. As a divergence, we choose $\mathcal{D}(p, q) = \mathbb{E}_p[\mathrm{TV}(x)] - \mathbb{E}_q[\mathrm{TV}(x)]$. We have defined our base model to be one in which all pixels are perfectly correlated and thus have the same value. This results in the expected TV for this distribution taking a value of 0. We end up with our divergence simply matching the expected TV under the extended model $\mathbb{E}_{\mathcal{N}(\mu, \Sigma_{xx})}[\mathrm{TV}(x)]$. Even when an expected TV of 0 is not attainable for any value of ℓ, as is the case when using the DIP eq. (15), there still exists a base model which will be constant with respect to our parameters of interest and can be safely ignored.

C.2 An upper bound on the expected TV

To ensure dimensionality preservation, we define our prior over ℓ in eq. (15) as a product of TV-PredCP priors, one defined for every convolutional block in the CNN, indexed by d,

$$p(\ell) = p(\ell_1)p(\ell_2)\dots p(\ell_D) = \prod_{d=1}^{D}\pi(\kappa_d)\left|\frac{\partial \kappa_d}{\partial \ell_d}\right|, \text{with } \kappa_d := \mathbb{E}_{\mathcal{N}(\hat{\theta}_d, \Sigma_{\theta_d\theta_d})\prod_{i=1, i\neq d}^{D}\delta(\theta_i - \hat{\theta}_i)}\left[\mathrm{TV}(h(\theta))\right].$$

This formula differs from the expected TV in eq. (9), which doesn't discriminate by blocks $\kappa := \mathbb{E}_{\mathcal{N}(\hat{\theta}, \Sigma_{\theta\theta})}\left[\mathrm{TV}(h(\theta))\right]$. By the triangle inequality, $\sum_d \kappa_d$ upper bounds the expectation under $\mathcal{N}(\hat{\theta}, \Sigma_{\theta\theta})$:

$$\mathbb{E}_{\mathcal{N}(\hat{\theta}, \Sigma_{\theta\theta})}\left[\mathrm{TV}(h(\theta))\right] = \sum_{(i,j)\in\mathcal{S}}\mathbb{E}_{\mathcal{N}(\hat{\theta}, \Sigma_{\theta\theta})}\left[|(J_i\theta - J_j\theta)|\right] = \sum_{(i,j)\in\mathcal{S}}\mathbb{E}_{\mathcal{N}(\hat{\theta}, \Sigma_{\theta\theta})}\left[|\sum_d (J_{id} - J_{jd})\theta_d)|\right]$$

$$\leq \sum_{(i,j)\in\mathcal{S}}\sum_d \mathbb{E}_{\mathcal{N}(\hat{\theta}_d, \Sigma_{\theta_d\theta_d})}\left[|(J_{id} - J_{jd})\theta_d|\right] = \sum_d \mathbb{E}_{\mathcal{N}(\hat{\theta}_d, \Sigma_{\theta_d\theta_d})\prod_{c=1, c\neq d}^{D}\delta(\theta_c - \hat{\theta}_c)}\left[\sum_{(i,j)\in\mathcal{S}}|(J_i - J_j)\theta|\right] = \sum_d \kappa_d,$$

where $\mathcal{S}$ is the set of all adjacent pixel pairs. Thus, the separable form of the TV prior as a regulariser for MAP ensures that the expectated TV under the joint distribution of parameters is also regularised.

C.3 Discussing monotonicity of the TV in the prior lengthscales

In order to apply the change of variables formula in eq. (15), we require bijectivity between ℓ_d and κ_d. In the simplest setting, both variables are one-dimensional, making this constraint easier to satisfy. In fact, it suffices to show monotonicity between the two.

In practice, we use the linearised model in eq. (6) for inference. In fig. 8, we show very compelling numerical evidence for the monotonicity. We observe that κ increases in ℓ since large values for ℓ lead to an increased marginal variance σ^2 over images. After fixing the marginal variance to 1, the lengthscales have a monotonically

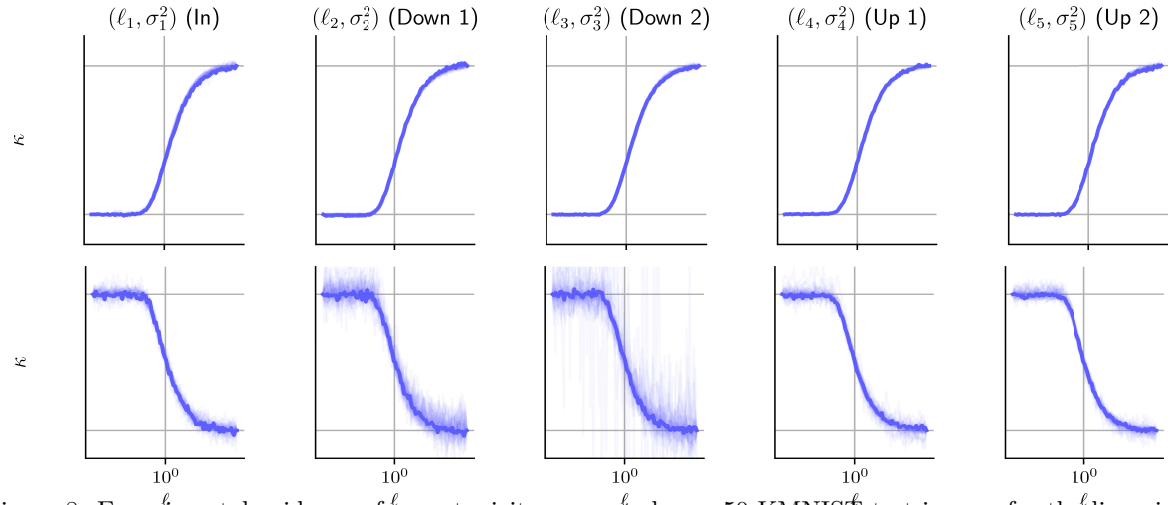

Figure 8: Experimental evidence of monotonicity computed over 50 KMNIST test images for the linearised network used in the KMNIST experiments. Horizontal axis represents lengthscale $\ell \in [0.01, 100]$. κ is estimated with 10k Monte Carlo samples. In the bottom row we fix the marginal variances of $J\Sigma_{yy}J^\top$ in image space to be 1. This allows us to observe the smoothing effect from ℓ. We use the first and last value to normalise over different KMNIST sample. The monotonicity implies the desired invertibility of the mappings ℓ and κ. We draw 500 samples to estimate k.

decreasing relationship with the expected TV. However, analytically studying the monotonicity is delicate. We investigate the issue in the linear setting to she insights (which also matches our experimental setup):

$$\kappa_d = \mathbb{E}_{\mathcal{N}(\hat{\theta},\Sigma_{\theta\theta})\prod_{j=1,j\neq d}^{D}\delta(\theta_j-\hat{\theta}_j)}[\mathrm{TV}(h(\theta))] = \mathbb{E}_{\mathcal{N}(\hat{\theta},\Sigma_{\theta\theta})\prod_{j=1,j\neq d}^{D}\delta(\theta_j-\hat{\theta}_j)}\Big[\sum_i |h(\theta)_i - h(\theta)_{i+1}|\Big], \tag{31}$$

assuming that the output is a 1D signal so there is only one derivative to simplify the discussion. First we derive the distribution of $h(\theta)_i - h(\theta)_{i+1}$. Note that $h(\theta)$ can be written as $h(\theta) = h_0 + J(\theta - \hat{\theta})$, by slightly abusing the notation h_0 to denote the vectors constant with respect to ℓ_d and i indices an entry of the vector $(J\theta) \in \mathbb{R}^{d_x}$. Note that the constant vector h_0 depends on the choice of the based point $\theta = 0$ (or equally plausible $\theta = \hat{\theta}$), but it does not play a role in $\mathrm{TV}(h(\theta))$, since it cancels out from the definition of $\mathrm{TV}(h(\theta))$. Then, we can rewrite it as an inner product between two vectors

$$h(\theta)_i - h(\theta)_{i+1} = (J\theta)_i - (J\theta)_{i+1} = (J_i - J_{i+1})\theta_d = v_i\theta_d,$$

where $J_i \in \mathbb{R}^{1\times d_{\theta_d}}$ denotes our NN's Jacobian for a single output pixel i (i.e. the ith row of the Jacobian matrix J, corresponding to the block parameters $\theta_d \in \mathbb{R}^{1\times d_{\theta_d}}$) and $v_i = J_i - J_{i+1} \in \mathbb{R}^{1\times d_{\theta_d}}$, $i = 1, \ldots, d_x - 1$. Now, the block parameters θ_d is distributed as $\theta_d \sim \mathcal{N}(0, \Sigma_{\theta_d\theta_d})$, in the expectation in eq. (31), whereas the remaining parameters are fixed at the mode $\hat{\theta}_j$, $j \neq d$, i.e. $\prod_{j=1,j\neq d}^{D}\delta(\theta_j - \hat{\theta}_j)$. Let $V_d \in \mathbb{R}^{(d_x-1)\times d_{\theta_d}}$ correspond to the stacking of the vectors $v_i \in \mathbb{R}^{1\times d_{\theta_d}}$, i.e. the Jacobian of the network output with respect to the weights in convolutional group d. Since the affine transformation of a Gaussian distribution remains Gaussian, $V_d\theta_d$ is distributed according to $V_d\theta_d \sim \mathcal{N}(0, V_d\Sigma_{\theta_d\theta_d}V_d^\top)$. Note that the matrix $V_d\Sigma_{\theta_d\theta_d}V_d^\top$ is not necessarily invertible, and if not, as usual, the inverse covariance should be interpreted in the sense of pseudo-inverse. Let $a =: V_d\theta_d \in \mathbb{R}^{d_x-1}$. Then

$$\kappa_d = \mathbb{E}_{a\sim\mathcal{N}(0,V_d\Sigma_{\theta_d\theta_d}V_d^\top)}\Big[\sum_i |a_i|\Big] = \sum_i \mathbb{E}_{a_i\sim\mathcal{N}(0,v_i\Sigma_{\theta_d\theta_d}v_i^\top)}[|a_i|].$$

The distribution of $|a_i|$ follows a half-normal distribution, and there holds (cf. eq. (3) of Leone et al. (1961))

$$\mathbb{E}_{a_i\sim\mathcal{N}(0,v_i\Sigma_{\theta_d\theta_d}v_i^\top)}[|a_i|] = \sqrt{\frac{2}{\pi}}(v_i\Sigma_{\theta_d\theta_d}v_i^\top)^{\frac{1}{2}}.$$

Consequently,

$$\kappa_d = \sqrt{\frac{2}{\pi}} \sum_i (v_i \Sigma_{\theta_d \theta_d} v_i^\top)^{\frac{1}{2}} \quad \text{and} \quad \frac{\partial \kappa_d}{\partial \ell_d} = \sqrt{\frac{1}{2\pi}} \sum_i (v_i \Sigma_{\theta_d \theta_d} v_i^\top)^{-\frac{1}{2}} v_i \frac{\partial}{\partial \ell_d} \Sigma_{\theta_d \theta_d} v_i^\top. \tag{32}$$

It remains to examine the monotonicity of $v_i \Sigma_{\theta_d \theta_d} v_i^\top$ in ℓ_d. Indeed, by the definition of Σ_d, we have

$$\frac{\partial}{\partial \ell_d} [\Sigma_{\theta_d \theta_d}(\ell_d)]_{j,j'} = \frac{\partial}{\partial \ell_d} \sigma_d^2 \exp\Big(-\frac{\mathrm{d}(j,j')}{\ell_d}\Big) = \frac{\sigma_d^2 \mathrm{d}(j,j')}{\ell_d^2} \exp\Big(-\frac{\mathrm{d}(j,j')}{\ell_d}\Big),$$

and thus

$$\frac{\partial}{\partial \ell_d} v_i \Sigma_{\theta_d \theta_d} v_i^\top = \frac{\sigma_d^2}{\ell_d^2} \sum_j \sum_{j'} v_{i,j} \mathrm{d}(j,j') \exp\Big(-\frac{\mathrm{d}(j,j')}{\ell_d}\Big) v_{i,j'}.$$

Then it follows that if the vectors v_i were arbitrary, the monotonicity issue would rest on the positive definiteness of the associated derivative kernel. For example, for a Gaussian kernel $e^{-\frac{(x-y)^2}{\ell_d}}$ (i.e. d is the squared Euclidean distance), the associated kernel $k(x,y)$ is given by $(x-y)^2 e^{-\frac{(x-y)^2}{\ell_d}}$. This issue seems generally challenging to verify directly, since $(x-y)^2$ is not a positive semidefinite kernel by itself on $\mathbb{R}$, even though the Gaussian kernel $e^{-\frac{(x-y)^2}{\ell_d}}$ is indeed positive semidefinite. Thus, one cannot use the standard Schur product theorem to conclude the monotonicity. Alternatively, one can also compute the Fourier transform of the kernel $k(x) = x^2 e^{-x^2}$ directly, which is given by

$$\mathcal{F}[k(x)](\omega) = \frac{2-\omega^2}{4} \frac{1}{\sqrt{2}} e^{-\frac{\omega^2}{4}}.$$

see the proposition below for the detailed derivation. Clearly, the Fourier transform of the kernel $x^2 e^{-x^2}$ is not positive over the whole real line $\mathbb{R}$. By Bochner's theorem (see e.g. p. 19 of Rudin (1990)), this kernel is actually not positive. The fact that the kernel is no longer positive definite makes the analytical analysis challenging. This observation holds also for the Matern-1/2 kernel, see the proposition below. These observations clearly indicate the risk for a potential non-monotonicity in ℓ. Nonetheless, we emphasise that this condition is only sufficient, but not necessary, since the kernel is only evaluated at lattice points (instead of arbitrary scattered points). We leave a full investigation of the monotonicity to a future work, given the compelling empirical evidence for monotonicity in both the NN and linearised settings.

Now we give Fourier transforms of the associated kernel for the Gaussian and Matern-1/2 kernels.

Proposition 1. *The Fourier transforms of the functions $x^2 e^{-x^2}$ and $|x| e^{-|x|}$ are given by*

$$\mathcal{F}[x^2 e^{-x^2}](\omega) = \frac{2-\omega^2}{4\sqrt{2}} e^{-\frac{x^2}{4}} \quad \textit{and} \quad \mathcal{F}[|x| e^{-|x|}](\omega) = \frac{2(1-\omega^2)}{\sqrt{2\pi}(1+\omega^2)^2}.$$

Proof. Recall that the Fourier transform $\mathcal{F}[e^{-x^2}]$ of the Gaussian kernel e^{-x^2} is given by

$$\mathcal{F}[e^{-x^2}](\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} e^{-x^2} e^{-\mathrm{i}\omega x} \mathrm{d}x = \frac{1}{\sqrt{2}} e^{-\frac{\omega^2}{4}}.$$

Direct computation shows

$$k''(x) = 4x^2 e^{-x^2} - 2e^{-x^2} = 4x^2 e^{-x^2} - 2k(x).$$

Taking Fourier transform on both sides and using the identity $\mathcal{F}[k''(x)](\omega) = -\omega^2 \mathcal{F}[f(x)](\omega)$, we obtain

$$-\omega^2 \mathcal{F}[f(x)](\omega) = 4\mathcal{F}[x^2 e^{-x^2}](\omega) - 2\mathcal{F}[f(x)](\omega),$$

which upon rearrangement gives the desired expression for $\mathcal{F}[x^2 f(x)]$. Next we compute $\mathcal{F}[|x| e^{-|x|}](\omega)$:

$$\begin{aligned} &\mathcal{F}[|x| e^{-|x|}](\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} |x| e^{-|x|} e^{-\mathrm{i}\omega x} \mathrm{d}x \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} |x| e^{-|x|} (\cos \omega x - \mathrm{i} \sin \omega x) \mathrm{d}x = \frac{2}{\sqrt{2\pi}} \int_0^{\infty} x e^{-x} \cos \omega x \mathrm{d}x, \end{aligned}$$

since $\sin \omega x$ is odd and the corresponding integral vanishes. Integration by parts twice gives

$$\begin{aligned}\int_0^\infty xe^{-x}\cos\omega x\mathrm{d}x &= -xe^{-x}\cos\omega x|_{x=0}^\infty + \int_0^\infty e^{-x}(\cos\omega x - \omega x\sin\omega x)\mathrm{d}x \\ &= \int_0^\infty e^{-x}\cos\omega x\mathrm{d}x - \int_0^\infty \omega x e^{-x}\sin\omega x\mathrm{d}x \\ &= \int_0^\infty e^{-x}\cos\omega x\mathrm{d}x + \omega x e^{-x}\sin\omega x|_{x=0}^\infty - \int_0^\infty e^{-x}(\omega\sin\omega x + \omega^2 x\cos\omega x)\mathrm{d}x.\end{aligned}$$

Rearranging the identity gives

$$\int_0^\infty xe^{-x}\cos\omega x\mathrm{d}x = \frac{1}{\omega^2+1}\int_0^\infty e^{-x}\cos\omega x\mathrm{d}x - \frac{\omega}{\omega^2+1}\int_0^\infty e^{-x}\sin\omega x\mathrm{d}x$$

This and the identities

$$\int_0^\infty e^{-x}\cos\omega x\mathrm{d}x = \frac{1}{1+\omega^2} \quad \text{and} \quad \int_0^\infty e^{-x}\sin\omega x\mathrm{d}x = \frac{\omega}{1+\omega^2},$$

immediately imply

$$\mathcal{F}[|x|e^{-|x|}](\omega) = \frac{2}{\sqrt{2\pi}}\int_0^\infty xe^{-x}\cos\omega x\mathrm{d}x = \frac{2(1-\omega^2)}{\sqrt{2\pi}(1+\omega^2)^2}.$$

This shows the second identity. □

D Additional experimental discussion

In this section, we provide additional empirical evaluation of the uncertainty estimates obtained with the linearised DIP. Validating the accuracy of the uncertainty estimates is crucial for their reliable integration into downstream tasks and computer human interaction workflows, as discussed by Antorán et al. (2021), Bhatt et al. (2021), and Barbano et al. (2021).

D.1 Evaluating approximate computations

We validate the accuracy of our approximate computation presented in section 6 on the KMNIST dataset. KMNIST is the perfect ground for this evaluation due to the fact that the low-dimensionality of d_x and d_y guarantees computational tractability of the inference problem, allowing us to benchmark the approximations we introduce in section 6, against exact computation. In this section, if not stated otherwise, we carry out our investigations with the setting where the forward operator A, comprises 20 angles, and we add 5% noise to Ax. We repeat the analysis on 10 characters taken from the test set of the KMNIST dataset. We assess the suitability of the Hutchinson trace estimator for the gradient of the log-determinant (section 6.1), and the ancestral sampling for the TV-PredCP gradients (section 6.2). Figure 9 and fig. 10 show hyperparameter optimisation $(\sigma_y^2, \sigma^2, \ell)$ using exact and estimated gradients. The hyperparameters trajectories match closely; we only observe tiny oscillations when using estimated gradients. The log-determinant gradients $\frac{\partial \log|\Sigma_{yy}|}{\partial \phi}$ are estimated using 10 samples, $v \sim \mathcal{N}(0, P)$. The PCG for solving $v^\top \Sigma_{yy}^{-1}$ uses a maximum of 50 iterations (with a early stopping criterion in place if a tolerance of 1.0 is met). We use a randomised SVD-based preconditioner P (cf. 6.1), where the rank, r, is chosen to be 200, and P is updated every 100 steps. The TV-PredCP gradients are estimated using 500 samples.

We assess the approximations introduced in section 6.3; the accuracy of the estimation of the posterior covariance matrix, but most importantly, the estimation of the test log-likelihood. For large image sizes (e.g. the Walnut cf. section 7.2), it is infeasible to store the posterior predictive covariance matrix $\Sigma_{x|y} \in \mathbb{R}^{d_x \times d_x}$, which in single precision would require 250 GB of memory. However, it can be made computationally cheaper if we consider smaller image patches of pixels, neglecting the inter-patch-dependencies. This assumes the covariance matrix $\Sigma_{x|y}$ to be block diagonal. Figure 11 shows the effect of neglecting inter-patch-dependencies. The log-likelihood increases with increasing patch-size (i.e. with more inter-dependencies being taken into

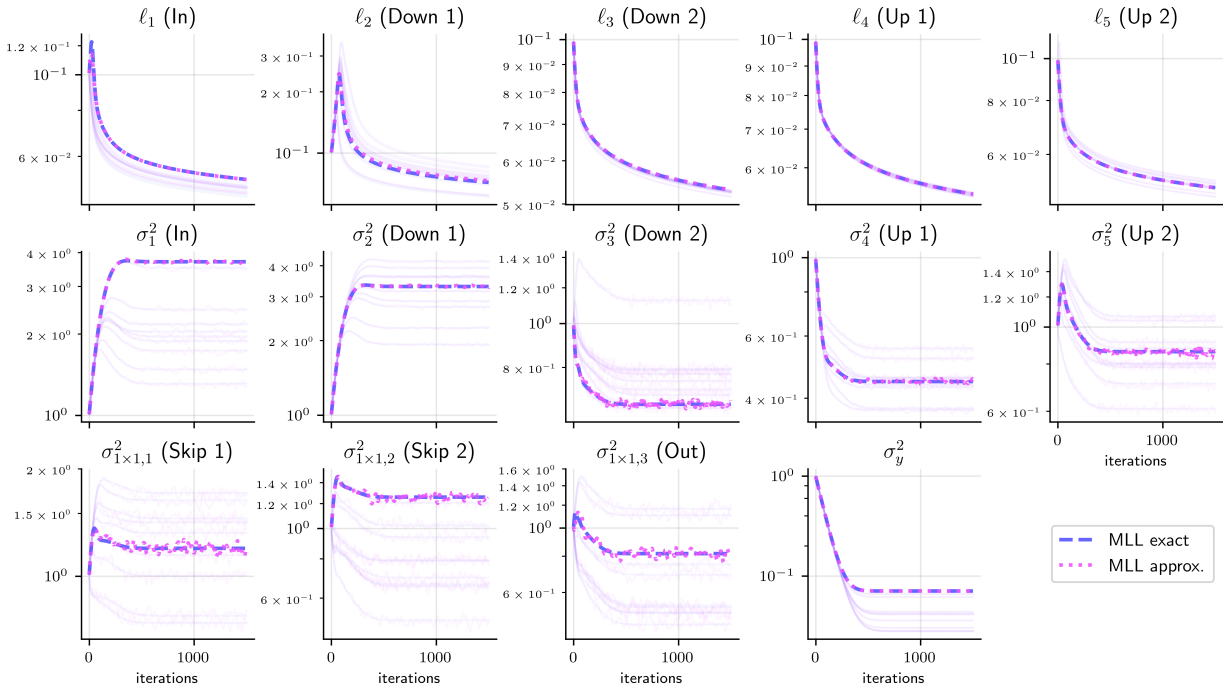

Figure 9: Hyperparameters' optimisation in eq. (17) for lin.-DIP excluding PredCP (MLL), computing exact gradients as well as resorting to the approximate numerical methods discussed in section 6.1 (i.e. PCG-based log-determinant gradients) on 10 KMNIST images.

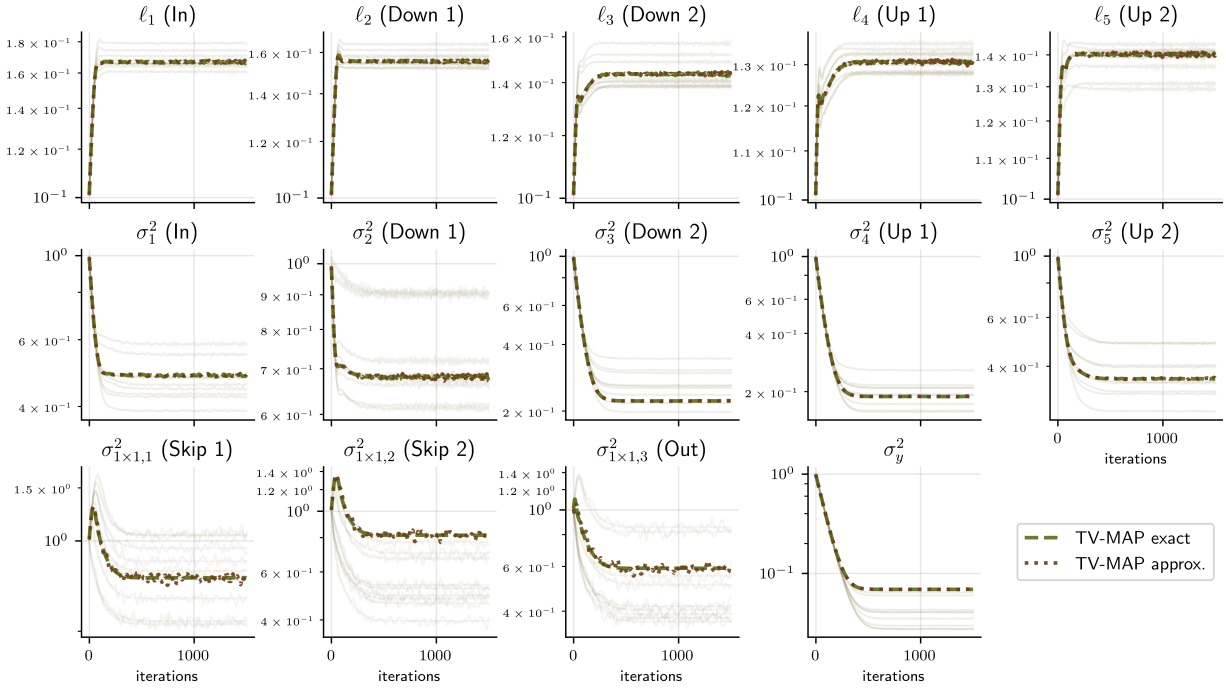

Figure 10: Hyperparameters' optimisation in eq. (17) for lin.-DIP including TV-PredCP (TV-MAP), computing exact gradients as well as resorting to the approximate numerical methods discussed in section 6.1 (i.e. PCG-based log-determinant gradients) and section 6.2 (i.e. ancestral sampling for TV-PredCP term) on 10 KMNIST images.

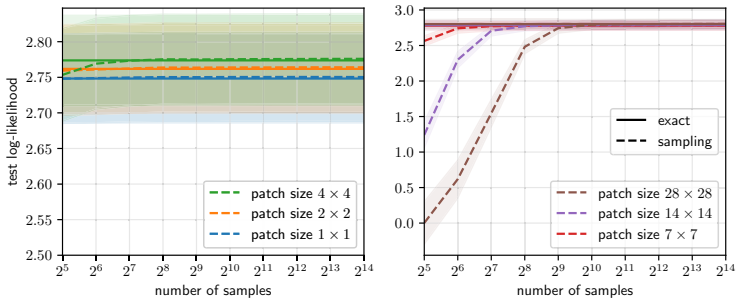

Figure 11: Test log-likelihood computed with posterior predictive covariance matrices estimated via eq. (22), and compared to the one obtained with exact methods (i.e. using exact posterior predictive covariance matrices via eq. (14)). The log-likelihood is overall well approximated. As we would expect, we observe that larger patches require more samples. We conduct our investigation over 10 KMNIST images, and show mean and standard error. $(\sigma_y^2, \sigma^2, \ell)$ are obtained using MLL.

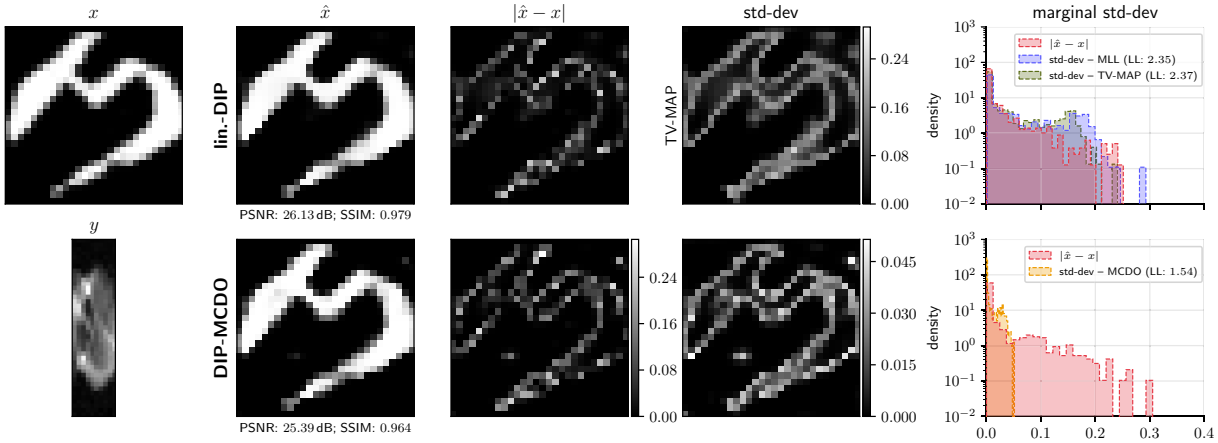

Figure 12: KMNIST character recovered from a simulated observation y (using 10 angles and $\eta(5\%)$) with lin.-DIP, DIP-MCDO and along with their uncertainty estimates and histogram plots.

account). Figure 11 shows how well the test log-likelihood is approximated when resorting to posterior predictive covariance matrices estimated via sampling using eq. (22), while sweeping across different numbers of samples and patch-sizes. As expected, estimating the log-likelihood for larger patch-sizes requires more samples. On KMNIST, 1024 samples are sufficient for almost perfect approximation of the test log-likelihood, when approximating the posterior predictive covariance matrix with patch-size of 28×28. Note that a patch-size of 28×28 on KMNIST implies that no inter-patch-dependencies are neglected.

D.2 Further discussion on KMNIST

We include additional experimental figures to support the discussion about the experiments in section 7.1.2. Figure 12, fig. 13, fig. 14, and fig. 15 are analogous to fig. 4, yet show a KMNIST character for four different problem settings: 10 angles and 20 angles, and the two noise regimes.

Figure 16 and fig. 17 show the hyperparameters' optimisation via Type-II MAP and MLL outlined in section 5.4. The use of our TV-PredCP prior leads to smaller marginal variances and larger lengthscales. This restricts our prior over reconstructions to smooth functions. The TV-PredCP introduces additional constraints into the model by encouraging the prior to contract (stronger parameter correlations and smaller posterior predictive marginal variances. In turn, this results in a more contracted posterior, which we observe as a larger Hessian determinant.

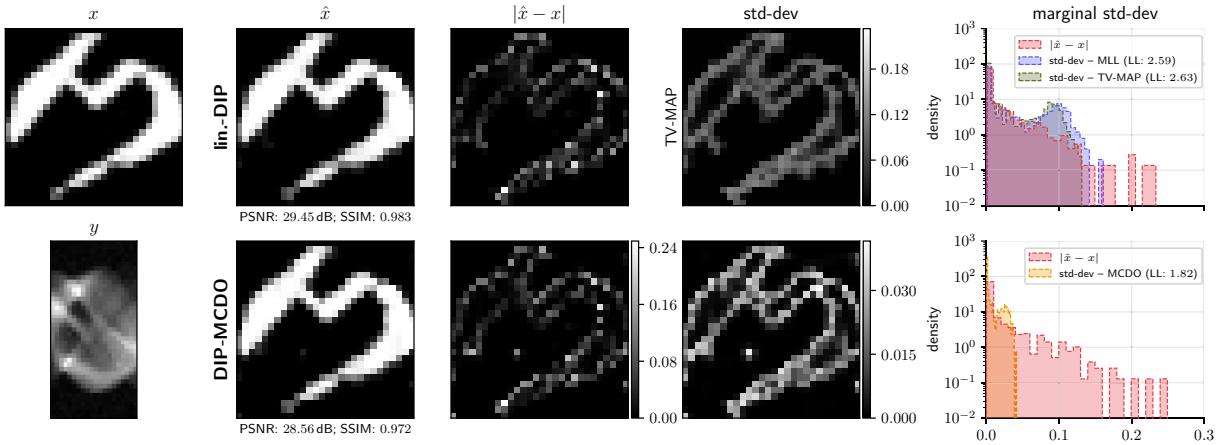

Figure 13: KMNIST character recovered from a simulated observation y (using 20 angles and $\eta(5\%)$) with lin.-DIP, DIP-MCDO along with their uncertainty estimates and histogram plots.

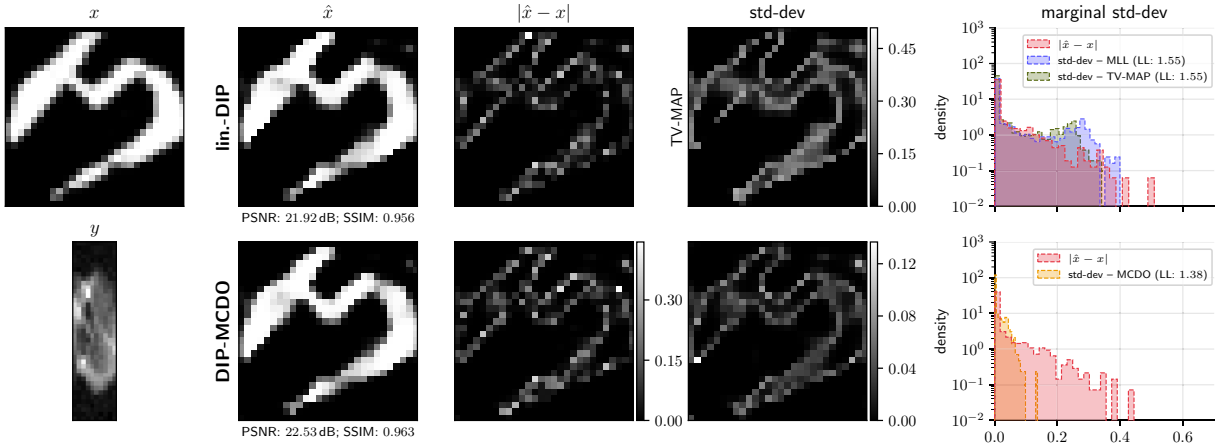

Figure 14: KMNIST character recovered from a simulated observation y (using 10 angles and $\eta(10\%)$) with lin.-DIP, DIP-MCDO along with their uncertainty estimates and histogram plots.

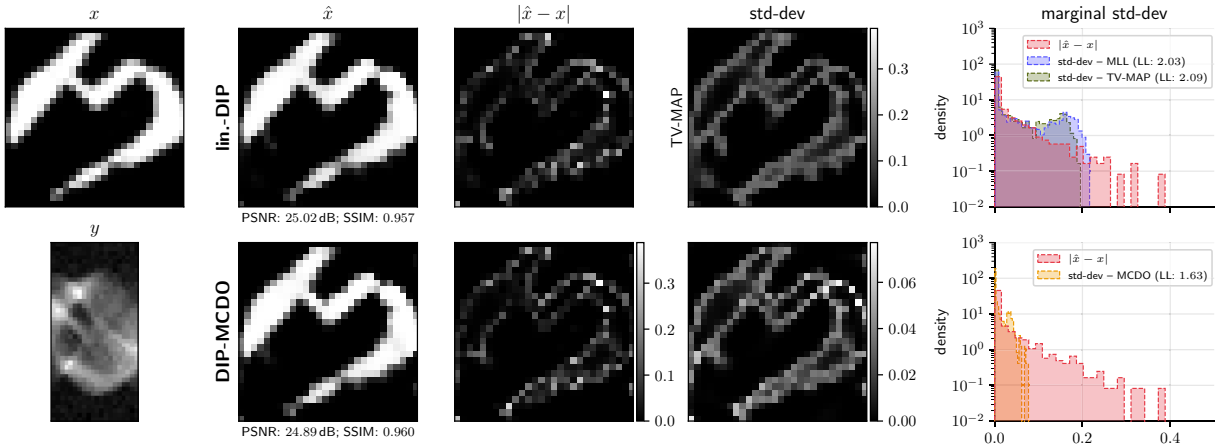

Figure 15: KMNIST character recovered from a simulated observation y (using 20 angles and $\eta(10\%)$) with lin.-DIP, DIP-MCDO along with their uncertainty estimates and calibration plots.

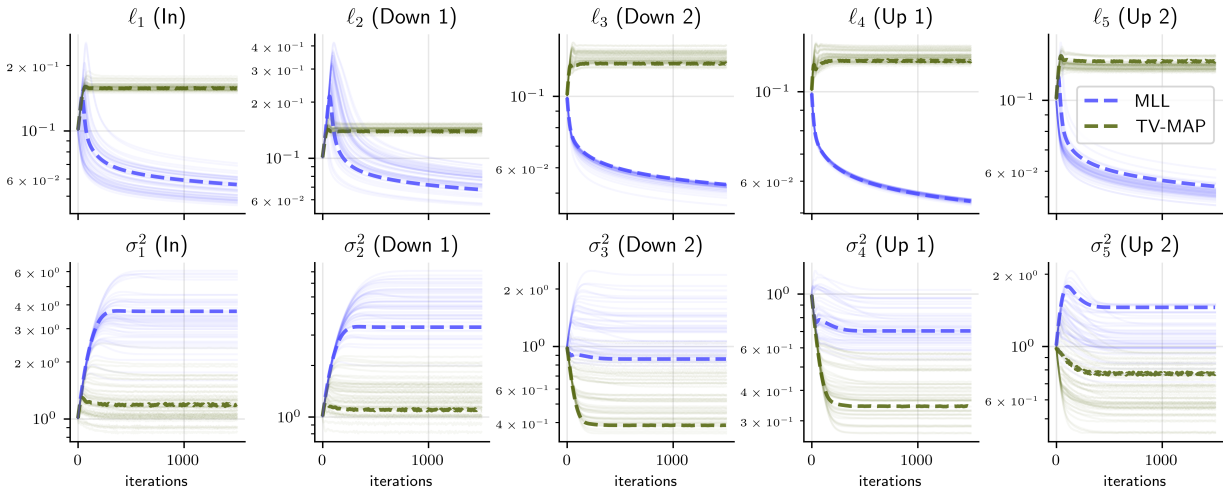

Figure 16: Optimisation of (ℓ, σ^2) via MLL and Type-II MAP for 3×3 convolution layers belonging to the small U-Net used for KMNIST. Thicker dotted lines refer to the optimisation of the exemplary reconstruction shown in fig. 4 while transparent lines correspond to other KMNIST images. The TV-PredCP leads to larger prior lengthscales ℓ and lower variances σ^2.

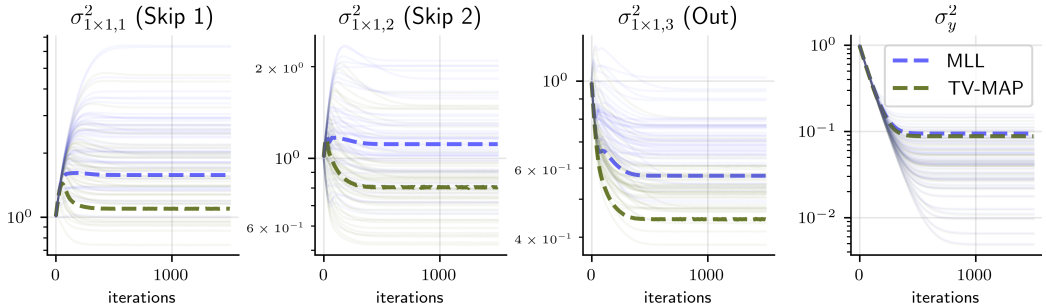

Figure 17: Hyperparameters' optimisation via MLL and Type-II MAP for 1×1 convolution layers belonging to the small U-Net used for KMNIST, along with σ_y^2. Thicker dotted lines refer to the optimisation of the KMNIST image shown in fig. 4, while transparent lines correspond to other KMNIST images.

For the KMIST dataset, one may question whether TV is an ideal regulariser. The TV regulariser enforces sparsity in the local image gradients. A TV regulariser is highly recommended when we observe sparsity in the edges present in an image, especially when the edges constitute a small fraction of the overall image pixels. That is often the case in high-resolution medical images or natural images. Intuitively, the higher the resolution of the image is, the higher the sparsity level of the edges is. However, in the KMIST dataset, due to the low resolution of the images, the edges constitute a considerable fraction of the total pixels. Therefore, a TV regulariser could be sub-optimal. In the KMNIST dataset, it is difficult to distinguish (in TV sense) what is part of the image structure and what is part of the background. The stroke is only a few pixels wide, and ground-truth pixel values are generated through interpolation (Clanuwat et al., 2018). Indeed we observe a larger gain from selecting hyperparameters using Type-II MAP (instead of MLL) for the real-measured high-resolution Walnut data than for KMNIST.

Furthermore, some KMNIST images present spurious high valued pixels away from the region containing the handwritten character. This contradicts the modelling assumption in eq. (1) which assumes x is noiseless. Our likelihood function from eq. (12) is defined over the space of observations y and thus can not account for noise in x. We translate the uncertainty induced by the observation noise to the space of images by computing the conditional log-likelihood Hessian with respect to x: $-\frac{\partial^2 \log p(y|x)}{\partial x^2} = \sigma_y^{-2} \mathrm{A}^\top \mathrm{A} \in \mathbb{R}^{d_x \times d_x}$. This matrix is of rank at most d_y, which potentially can be much smaller than d_x due to the ill-conditioning of the reconstruction problem, and therefore cannot act as a proper Gaussian precision matrix on its own. We incorporate the noise uncertainty from the observation subspace into the image space by adding the mean of the diagonal of the pseudoinverse $\sigma_y^2(\mathrm{A}^\top \mathrm{A})^\dagger$ to the marginal variances of the predictive distribution. This can also be seen as placing a Gaussian likelihood over reconstruction space, which can be marginalised to recover the predictive distribution $p(x|y) = \int \mathcal{N}(x; \hat{x}, \sigma_y^2 \mathrm{Tr}((\mathrm{A}^\top \mathrm{A})^\dagger) d_x^{-1} \mathrm{I}) \mathcal{N}(\theta; \hat{\theta}, \Sigma_{\theta|y})\, \mathrm{d}\theta = \mathcal{N}(x; \hat{x}, J \Sigma_{\theta|y} J^\top + \sigma_y^2 \mathrm{Tr}((\mathrm{A}^\top \mathrm{A})^\dagger) d_x^{-1} \mathrm{I})$.

D.3 Further discussions on Walnut data

We include additional figures to support the discussion in section 7.2. We evaluate the effect of the TV-PredCP prior for hyperparameter optimisation. We observe that this prior leads to a slightly less heavy tailed standard deviation histogram. It presents slightly better agreement with the empirical reconstruction error, resulting in a lager log-likelihood. Figure 18 and fig. 19 show the optimisation of the hyperparameters (σ_y^2, ℓ, $\boldsymbol{\sigma}_\theta^2$) using the method in section 5.4 and approximate computations in section 6. For both MLL and Type2-MAP learning, the marginal variance for all CNN blocks except the two closest to the output goes to $\approx$0. This is due to the representations from these last layer being able to explain the data well on their own. The our hyperparameter objectives are thus able to eliminate previous layers from our probabilistic model, simplifying it without sacrificing reconstruction quality. We did not observe this for KMNIST data, possibly because of our use of a smaller, less overparametrised network without any spare capacity.

E Additional experimental setup details

E.1 Setup for KMNIST experiments

We use a down-sized version of U-Net (Ronneberger et al., 2015), cf. fig. 20, as the reduced output dimension d_x and the simplicity of the problem allow us to employ a shallow architecture without compromising the reconstruction quality. This problem is computationally tractable removing the need for the approximations described in section 6. We reduce the U-Net architecture in fig. 3 to 3 scales and 32 channels at each scale, remove group-normalisation layers and use a sigmoid activation for the output. A filtered back-projection reconstruction from y is used as the network input.

Table 7 lists the hyperparameters of DIP optimisation for each setting. These values were found by grid-search on 50 KMNIST training images. The dropout rate p of DIP-MCDO is set to 0.05.

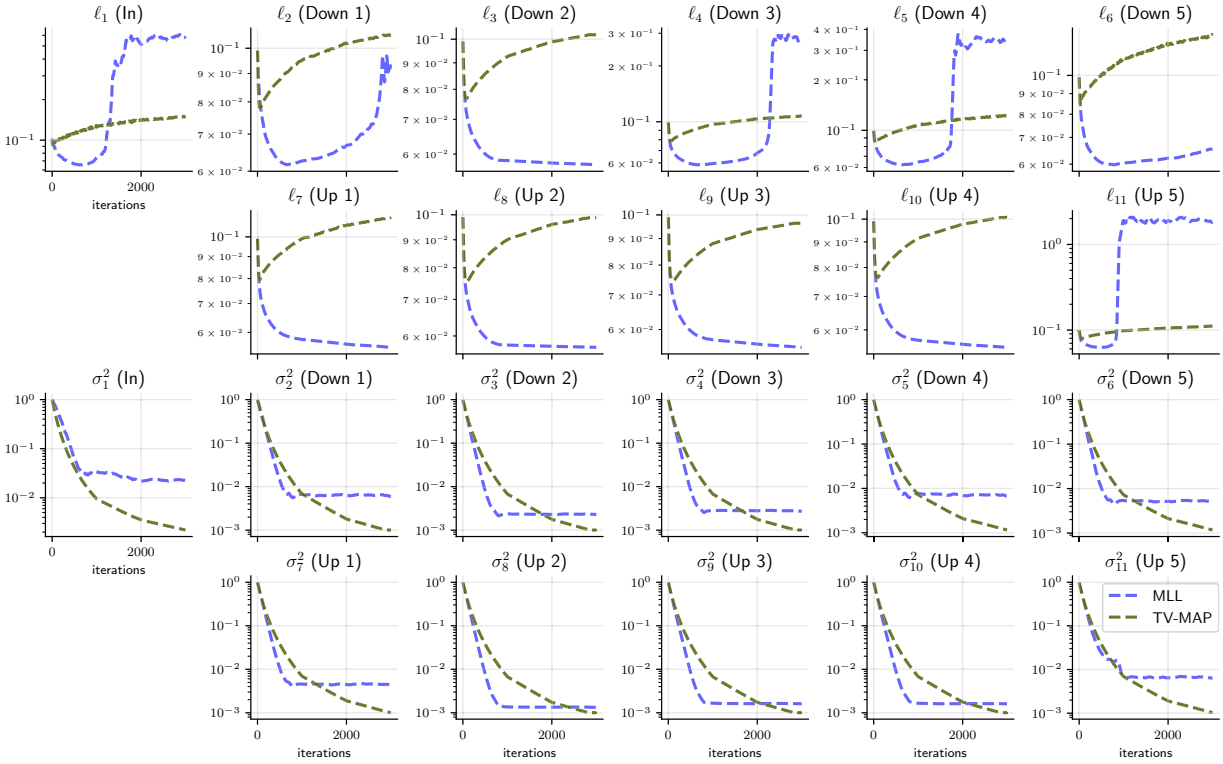

Figure 18: Optimisation of (ℓ, σ^2) via MLL and Type-II MAP for 3×3 convolution layers for the Walnut data described in section 7.2. As in fig. 16, the TV-PredCP leads to larger prior lengthscales ℓ and lower variances σ^2.

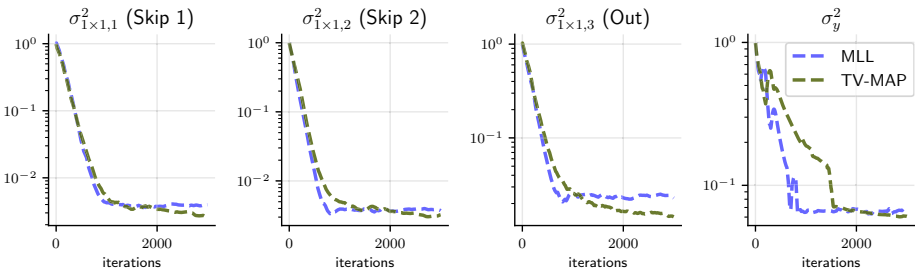

Figure 19: Optimisation of (σ_y^2, σ^2) via MLL and Type-II MAP for 1×1 convolutions.

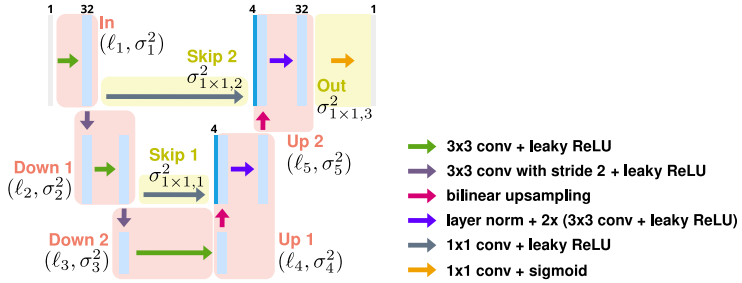

Figure 20: A schematic illustration of the reduced U-Net architecture used in the numerical experiments on KMNIST data. It has 3 scales and does not include group norm layers. Each light-blue rectangle corresponds to a multi-channel feature map. We highlight the architectural components corresponding to each block for which a separate prior is defined with red boxes.

Table 7: Hyperparameters of DIP optimisation selected using 50 randomly chosen images from the KMNIST training set. The λ values refer to our implementation of eq. (5) in which $\|\cdot\|^2$ is replaced with mean squared error (or the regularisation term is up-scaled by d_x).

	5% noise				10% noise			
#angles	5	10	20	30	5	10	20	30
TV scaling for DIP: λ	1e−5	3e−5	1e−4	1e−4	3e−5	1e−4	3e−4	3e−4
DIP iterations	14 000	29 000	41 000	50 000	7400	13 000	17 000	22 000

E.2 Computing the preconditioner for conjugate gradients

For our preconditioner P, we approximate $\mathrm{A}J\Sigma_{\theta\theta}J^\top\mathrm{A}^\top$ —for simplicity denoted as $H \in \mathbb{R}^{d_y\times d_y}$— as $\tilde{U}\tilde{\Lambda}\tilde{U}^\top$, using a randomised eigendecomposition algorithm (Halko et al., 2011; Martinsson & Tropp, 2020) with $\tilde{U} \in \mathbb{R}^{d_y\times r}$ and $r \ll d_y$. The approach first computes an orthonormal basis capturing the space spanned by H's columns. The idea is to obtain a matrix Q with r orthonormal columns, that approximates the range of H. This is done by constructing a standard normal test matrix $\Omega \in \mathbb{R}^{d_y\times r}$, and computing the (thin) QR decomposition of $H\Omega$. Once Q is computed, we solve for a symmetric matrix $B \in \mathbb{R}^{r\times r}$ (much smaller than H) such that B approximately satisfies $B(Q^\top\Omega) \approx Q^\top H\Omega$. We then compute the eigendecomposition of B, $V\Lambda V^\top$, and recover $\tilde{U} = QV$. This method requires $\mathcal{O}(r)$ matvecs resembling Hv to construct not only an approximate basis but also its complete factorisation. Finally, the preconditioner P is defined as $\tilde{U}\tilde{\Lambda}\tilde{U}^\top + \sigma_y^2\mathrm{I}$. To compute $P^{-1}v$ efficiently, we make use of the Woodbury identity.

E.3 Setup for X-ray Walnut data experiments

In (Der Sarkissian et al., 2019) projection data sets obtained with three different source positions are provided for 42 walnuts, as well as high-quality reconstructions of size 501^3 px^3 obtained via iterative reconstruction using the measurements from all three source positions. We consider the task of reconstructing a single slice of size 501^2 of the first walnut from a sub-sampled set of measurements using the second source position, which corresponds to a sparse fan-beam-like geometry. From the original 1200 projections (equally distributed over 360°) of size 972×768 we first select the appropriate detector row matching the slice position (which varies for different detector columns and angles due to a tilt in the setup), yielding measurement data of size $1200 \cdot 768$. We then sub-sample in both angle and column dimensions by factors of 20 and 6, respectively, leaving $d_y = 60 \cdot 128 = 7680$ measurements. For evaluation metrics, we take the corresponding slice from the provided high-quality reconstruction as the reference ground truth image x. The sparse operator matrix A is assembled by calling the forward projection routine of the `ASTRA` toolbox (van Aarle et al., 2015) for every standard basis vector, $\mathrm{A} = \mathrm{A}[e_1, e_2, \ldots e_{d_x}]$. While especially for large data dimensions it would be favourable to directly use the matrix-free implementations from the toolbox, we also need to evaluate the transposed operation $v_y^\top\mathrm{A}$, which would be only approximately matched by the back-projection routine (especially for the tilted 2D sub-geometry, which would require padding). Therefore, we resort to the sparse matrix multiplication via PyTorch.

The network architecture is shown in fig. 3. Following Barbano et al. (2022c), we pretrain the network to perform post-processing of filtered back-projection (FBP) reconstructions on synthetic data. The dataset consists of pairs of images containing random ellipses, and corresponding FBPs from observations simulated according to eq. (1) with 5% noise. The supervised pretraining accelerates the convergence of the subsequent unsupervised DIP reconstruction from y. In the DIP phase, the FBP of y is used as the network input. Table 8 lists the hyperparameters of DIP optimisation. The dropout rate p of DIP-MCDO is set to 0.05.

After DIP optimisation, following Antorán et al. (2022) the network weights are refined for the linearised model (eq. (6)). We optimise the same loss function as for DIP, but with the linear model eq. (6) instead of the network model, for 1000 steps. This yields network weights that fit better the subsequent MLL / Type-II MAP optimisation eq. (17), which employs the linear model.

Table 8: Hyperparameters of DIP optimisation used for the walnut data. The λ value refers to our implementation of eq. (5) in which $\|\cdot\|^2$ is replaced with mean squared error (or the regularisation term is upscaled by d_x).

TV scaling for DIP: λ	6.5e−6
DIP iterations (after pretraining)	1500

In MLL / Type-II MAP optimisation eq. (17), we use 10 probes to estimate the gradients of the log-determinant $\log|\Sigma_{yy}|$ eq. (19), employing the PCG method for solving $v^\top \Sigma_{yy}^{-1}$ using a maximum of 50 steps with a randomised SVD-based preconditioner P of rank 200 that is updated every 100 steps. The TV-PredCP gradients eqs. (20) and (21) are estimated using 20 samples. The MLL / Type-II MAP optimisation is run for 3000 iterations.

The posterior predictive covariance matrices for all methods are estimated by drawing 4096 zero-mean samples and computing empirical posterior predictive covariance matrix. The latter is done for patch-sizes from 1×1 up to 10×10 image patches. We use a stabilising heuristic for the estimated covariance matrices, inspired by Maddox et al. (2019): by letting $\tilde{\Sigma}_{x|y} \leftarrow \alpha\Sigma_{x|y} + (1-\alpha)\mathrm{diag}(\mathrm{diag}(\Sigma_{x|y}))$, $\alpha = \frac{1}{2}$, the impact of the off-diagonal entries is reduced. Note that our Gaussian assumption is correct in the case of linearised DIP but not for MCDO. However, MCDO does not provide a closed form density over the reconstructed image, only samples. The dimensionality of the reconstruction is too large for exact density estimation on real-measured data. We thus compute the log-likelihood in the same way as for the linearised DIP, i.e. via a Gaussian distribution with mean and posterior predictive covariance matrices estimated from samples. The accelerated sampling method via $\tilde{J}$ & PCG uses a randomised SVD-based 500-rank approximation $\tilde{J}$ of the Jacobian, and PCG for solving $v^\top \Sigma_{yy}^{-1}$ with a maximum of 50 steps along with a randomised SVD-based preconditioner of rank 400. This sampling variant can be performed in single precision (32 bit floating point). Thus constructing $\tilde{J}$ is actually much faster than reported in table 1 (0.5 min instead of 0.2 h).

Paper 6

Bayesian experimental design for computed tomography with the linearised deep image prior

ICML2022 Workshop on Adaptive Experimental Design and Active Learning in the Real World

Bayesian Experimental Design for Computed Tomography with the Linearised Deep Image Prior

Riccardo Barbano* RICCARDO.BARBANO.19@UCL.AC.UK
Department of Computer Science, University College London

Johannes Leuschner* JLEUSCHN@UNI-BREMEN.DE
Center for Industrial Mathematics, University of Bremen

Javier Antorán* JA666@CAM.AC.UK
Department of Engineering, University of Cambridge

Bangti Jin B.JIN@UCL.AC.UK
Department of Computer Science, University College London

José Miguel Hernández-Lobato JMH233@CAM.AC.UK
Department of Engineering, University of Cambridge

Abstract

We investigate adaptive design based on a single sparse pilot scan for generating effective scanning strategies for computed tomography reconstruction. We propose a novel approach using the linearised deep image prior. It allows incorporating information from the pilot measurements into the angle selection criteria, while maintaining the tractability of a conjugate Gaussian-linear model. On a synthetically generated dataset with preferential directions, linearised DIP design allows reducing the number of scans by up to 30% relative to an equidistant angle baseline.

1. Introduction and related work

Linear inverse problems in imaging aim to recover an unknown image $x \in \mathbb{R}^{d_x}$ from measurements $y \in \mathbb{R}^{d_y}$, which are often described by the application of a forward operator $\mathrm{A} \in \mathbb{R}^{d_y \times d_x}$, and the addition of Gaussian noise $\epsilon \sim \mathcal{N}(0, \sigma_y^2 \, \mathrm{I}_{d_y})$ as

$$y = \mathrm{A}x + \epsilon. \tag{1}$$

This acquisition model is ubiquitous in machine vision, computed tomography (CT), and magnetic resonance imaging among other applications. Due to the inherent ill-posedness of the task (e.g. $d_y \ll d_x$), suitable regularisation or prior assumptions are crucial for the stable and accurate recovery of x (Tikhonov and Arsenin, 1977; Ito and Jin, 2014). In this work, we focus on X-ray imaging, a setting with application to both medical and industrial settings (Buzug, 2011).

In CT, an emitter sends X-ray quanta through the object being scanned. The quanta are captured by d_p detector elements placed opposite the emitter. Each row of A tells us about which regions (pixels) the X-ray quanta will pass through before reaching a detector element (cf. fig. 1). The number of X-ray quanta measured by a detector pixel conveys information about the attenuation coefficient of the material present along the quanta's path. This procedure is repeated at $d_{\mathcal{B}}$ angles, yielding a measurement of dimension $d_y = d_p \cdot d_{\mathcal{B}}$.

∗. Authors contributed equally. Our code is at github.com/educating-dip/bayesian_experimental_design

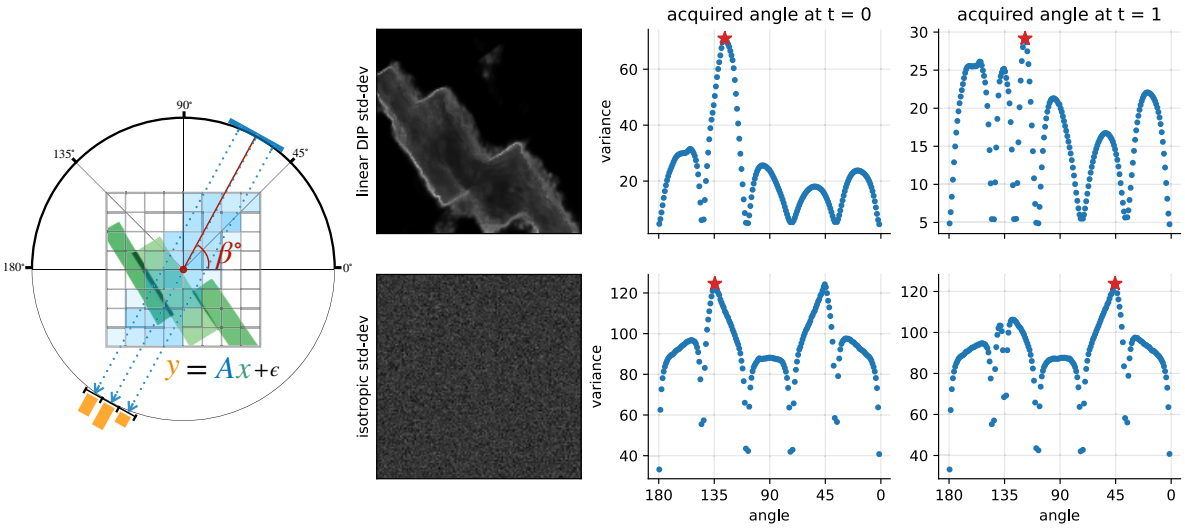

Figure 1: Left: A schematic diagram of 2D parallel beam CT geometry, used in the experiments. Top row: the linearised DIP assigns prior variance to pixels where edges are present, guiding angle selection so that X-ray quanta cover these pixels. Bottom row: the isotropic linear model's variance does not depend on the measurements. Angles $45, 135$ are chosen since they are oblique and maximise quanta path-length in the image.

In CT, Bayesian experimental design employs prior assumptions to select scanning angles which are aimed to yield the highest fidelity reconstruction. Adaptive design further incorporates information gained at previous angles to inform subsequent angle selections (Chaloner and Verdinelli, 1995). These methods are of great practical interest since they promise to reduce radiation dosages and scanning times. Alas, existing CT design methods often struggle to improve over equidistant angle choice (Shen et al., 2022). Furthermore, the requisite of additional computations before subsequent scans makes adaptive methods impractical for many applications.

Critically important to experimental design is the choice of prior (Feng, 2015; Foster, 2021). Linear models allow for tractable computation of quantities of interest for design, but their predictive uncertainty is independent of previously measured values, disallowing adaptive design (Burger et al., 2021). More complex model choices make inference difficult, necessitating approximations which can degrade performance (Helin et al., 2022; Shen et al., 2022).

This work aims to make adaptive design practical by considering a setting where the CT scan is performed in two phases. First, a sparse pilot scan is performed to provide data with which to fit adaptive methods. These are then used to select angles for a full scan. We demonstrate this procedure with a synthetic dataset where a different "preferential" angle is most informative for each image. Preferential directions appear commonly in industrial CT for material science and in medical CT for medical implant assessment. We use the linearised Deep Image Prior (DIP) (Barbano et al., 2022) as a data-dependent prior for adaptive design which preserves the tractability of conjugate Gaussian-linear models. Unlike simple linear models, the linearised DIP outperforms the equidistant angle baseline. Finally, we show that designs obtained with the linearised DIP perform well under traditional (non DIP-based) regularised-reconstruction.

2. Regularised reconstruction and deep image prior

Total Variation (TV) is the most popular regulariser for CT reconstruction (Rudin et al., 1992; Chambolle et al., 2010). The anisotropic TV semi-norm of an image vector $x \in \mathbb{R}^{d_x}$ is given by

$$\mathrm{TV}(x) = \sum_{i,j} |X_{i,j} - X_{i+1,j}| + \sum_{i,j} |X_{i,j} - X_{i,j+1}|, \tag{2}$$

where $X \in \mathbb{R}^{h \times w}$ denotes the vector $x \in \mathbb{R}^{d_x}$ reshaped into an image of height h by width w, and $d_x = h \cdot w$. The corresponding regularised reconstruction is obtained by

$$x^\star \in \operatorname*{argmin}_{x \in \mathbb{R}^{d_x}} \|\mathrm{A}x - y\|^2 + \lambda \mathrm{TV}(x), \tag{3}$$

where the hyperparameter $\lambda > 0$ determines the strength of regularisation.

The DIP (Ulyanov et al., 2020; Baguer et al., 2020) reparametrises the reconstruction x as the output of a U-net $x(\theta)$ (Ronneberger et al., 2015) with a fixed input, which we omit for clarity, and parameters $\theta \in \mathbb{R}^{d_\theta}$. The resulting reconstruction problem reads

$$\theta^\star \in \operatorname*{argmin}_{\theta \in \mathbb{R}^{d_\theta}} \|\mathrm{A}x(\theta) - y\|^2 + \lambda \mathrm{TV}(x(\theta)) \quad \text{and} \quad x^\star = x(\theta^\star). \tag{4}$$

We follow Barbano et al. (2021) in accelerating optimisation of eq. (4) using pre-trained U-nets.

3. Linear(ised) models for CT experimental design

Let $\mathcal{B}_a$ be the set of *all* possible angles at which we can scan. The task is to choose the subset of angles $\mathcal{B} \subset \mathcal{B}_a$ which produces the highest-fidelity reconstruction. We shall add angles sequentially over T steps. The set $\mathcal{B}^{(t)}$ denotes the chosen angles up to step $t < T$, and $\bar{\mathcal{B}}^{(t)} = \mathcal{B}_a \setminus \mathcal{B}^{(t)}$ the angles left to choose from. $\mathcal{B}^{(0)}$ denotes the set of angles used in the initial pilot scan, and $\mathcal{B} = \mathcal{B}^{(T)}$ the full design. We incorporate a decision to scan at angle $\beta \in \bar{\mathcal{B}}^{(t)}$ by concatenating the matrix $\mathrm{A}^\beta \in \mathbb{R}^{d_p \times d_x}$, which contains a row for each detector pixel at angle β, to the operator. After step t, the operator $\mathrm{A}^{(t)} \in \mathbb{R}^{d_p \cdot d_{\mathcal{B}^{(t)}} \times d_x}$ stacks $d_{\mathcal{B}^{(t)}}$ of these matrices, with $d_{\mathcal{B}^{(t)}} = |\mathcal{B}^{(t)}|$. $\bar{\mathrm{A}}^{(t)} \in \mathbb{R}^{d_p \cdot d_{\bar{\mathcal{B}}^{(t)}} \times d_x}$ denotes the forward operator for the angles left to choose from.

For design, we place a multivariate Gaussian prior on x with zero mean and covariance matrix $\Sigma_{xx} \in \mathbb{R}^{d_x \times d_x}$. Together with the Gaussian noise model in eq. (1), this gives a conjugate Gaussian-linear model. The vector $y^{(t)} \in \mathbb{R}^{d_p \cdot d_{\mathcal{B}^{(t)}}}$ of all measurements at step t is distributed as

$$y^{(t)}|x \sim \mathcal{N}(\mathrm{A}^{(t)}x,\, \sigma_y^2 \mathrm{I}_{d_y}) \quad \text{with} \quad x \sim \mathcal{N}(0, \Sigma_{xx}).$$

Thus, $\Sigma_{yy}^{(t)} = \mathrm{A}^{(t)}\Sigma_{xx}(\mathrm{A}^{(t)})^\top + \sigma_y^2 \mathrm{I}$ is the measurement covariance and the posterior over x is

$$x|y^{(t)} \sim \mathcal{N}(\mu_{x|y^{(t)}}, \Sigma_{x|y^{(t)}}), \quad \text{with}$$
$$\mu_{x|y^{(t)}} = \Sigma_{xx}(\mathrm{A}^{(t)})^\top (\Sigma_{yy}^{(t)})^{-1} y^{(t)}, \quad \text{and} \quad \Sigma_{x|y^{(t)}} = \Sigma_{xx} - \Sigma_{xx}(\mathrm{A}^{(t)})^\top (\Sigma_{yy}^{(t)})^{-1} \mathrm{A}^{(t)} \Sigma_{xx}. \tag{5}$$

The predictive covariance $\Sigma_{x|y^{(t)}}$ completely characterises the uncertainty of the reconstruction at step t and is the building block for the angle selection criteria in section 3.1. Note that natural images often exhibit heavy-tailed non-Gaussian statistics (Seeger and Nickisch, 2011). Additionally, by eq. (5), $\Sigma_{x|y^{(t)}}$ depends on the choice of angles through $\mathrm{A}^{(t)}$, but not on the measurements made at said angles $y^{(t)}$, precluding adaptive design. In section 3.2, we construct Σ_{xx} with correlations between nearby pixels, imitating the effects of the TV regulariser eq. (2), and with dependence on previous measurements, recovering adaptive design capability. In the experiments, we use linear models for angle selection and afterwards we discard the predictive mean $\mu_{x|y}$ and employ the regularised approaches from section 2 for reconstruction.

3.1 Experimental design with linear models

Acquisition objectives. Since the linear design task is submodular (Seeger, 2009), we greedily add one single angle per acquisition step [1]. We consider two popular acquisition objectives.

The first objective, *expected information gain* (EIG) (Mackay, 1992a), is the expected reduction in the posterior entropy $H(x|y)$ from scanning at angle β. At step t, it is given by

$$\text{EIG} := \text{H}(x|y^{(t)}) - \mathbb{E}_{p(y^\beta|y^{(t)})}[\text{H}(x|y^{(t)}, y^\beta)] = \text{logdet}(\sigma_y^2 \text{I}_{d_{\mathcal{B}^{(t)}}} + \text{A}^\beta \Sigma_{x|y^{(t)}} (\text{A}^\beta)^\top) + C \tag{6}$$

where the constant $C = -\text{logdet}(\sigma_y^2 \text{I})$ is independent of the angle choice. We give a derivation in appendix A for completeness. Intuitively, the determinant of the matrix $\text{A}^\beta \Sigma_{x|y^{(t)}} (\text{A}^\beta)^\top \in \mathbb{R}^{d_p \times d_p}$ penalises angles for which different detector elements make correlated measurements and the log term encourages the measurements from all detector pixels to be similarly informative.

The second objective, which we find to perform better empirically, is to choose the angles for which our prediction has the largest *expected squared error* (ESE) in measurement space

$$\text{ESE} := \mathbb{E}_{p(y^\beta,\, x|y^{(t)})}[(y^\beta - \text{A}^\beta x)^\top (y^\beta - \text{A}^\beta x)] = \text{Tr}(\text{A}^\beta \Sigma_{x|y^{(t)}} (\text{A}^\beta)^\top) + C. \tag{7}$$

This objective is equivalent to EIG in the setting where our detector has a single pixel.

Efficient acquisition. Constructing the matrix $\text{A}^\beta \Sigma_{x|y^{(t)}} (\text{A}^\beta)^\top$ repeatedly for each candidate angle $\beta \in \bar{\mathcal{B}}^{(t)}$ requires $\mathcal{O}(d_p \cdot d_{\bar{\mathcal{B}}^{(t)}})$ matrix vector products, which is very costly even for moderate size scanners. Instead, we estimate the matrix for every angle simultaneously by drawing K samples from $\mathcal{N}(0, \bar{\text{A}}^{(t)} \Sigma_{x|y^{(t)}} (\bar{\text{A}}^{(t)})^\top)$ with $\mathcal{O}(K)$ matrix vector products. That is, we sample $\mathbb{R}^{d_p \cdot d_{\bar{\mathcal{B}}^{(t)}}}$ sized vectors built by concatenating the "pseudo measurements" for each unused angle $\beta \in \bar{\mathcal{B}}^{(t)}$. We use Matheron's rule (Hoffman and Ribak, 1991; Wilson et al., 2021)

$$\begin{aligned} \bigoplus_{\beta \in \bar{\mathcal{B}}^{(t)}} y_k^\beta = \bar{\text{A}}^{(t)} \Big(x_k - \Sigma_{xx} (\text{A}^{(t)})^\top \Sigma_{yy}^{-1} (\eta_k + \text{A}^{(t)} x_k) \Big) \quad \text{with} \\ x_k \sim \mathcal{N}(0, \Sigma_{xx}) \quad \text{and} \quad \eta_k \sim \mathcal{N}(0, \sigma_y^2 \text{I}), . \end{aligned} \tag{8}$$

Here, $k \in \{1, ..., K\}$ indexes different samples and $\bigoplus$ denotes vector concatenation. We compute

$$\text{A}^\beta \Sigma_{x|y^{(t)}} (\text{A}^\beta)^\top \approx K^{-1} \textstyle\sum_{k=1}^K y_k^\beta (y_k^\beta)^\top,$$

which is then used to estimate the acquisition objective eq. (6) or eq. (7). The log term makes EIG estimates only asymptotically unbiased (i.e. as $K \to \infty$) but we find the bias to be insignificant. Once the angle β that maximises eq. (6) or eq. (7) is chosen, we update $\Sigma_{yy}^{(t+1)}$ as

$$\Sigma_{yy}^{(t+1)} = \begin{bmatrix} \Sigma_{yy}^{(t)} & \text{A}^{(t)} \Sigma_{xx} (\text{A}^{(t+1)})^\top \\ \text{A}^{(t+1)} \Sigma_{xx} (\text{A}^{(t)})^\top & \text{A}^{(t+1)} \Sigma_{xx} (\text{A}^{(t+1)})^\top \end{bmatrix}, \tag{9}$$

and repeat the procedure, i.e. return to eq. (8).

1. Submodularity guarantees this procedure obtains a score within a $(1 - 1/e)$ factor of the optimal strategy.

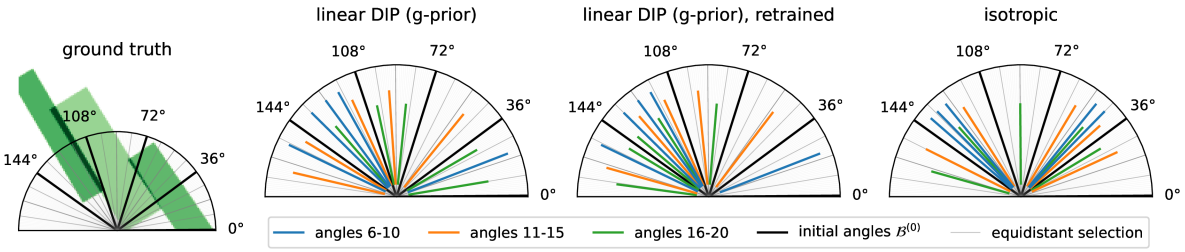

Figure 2: First 20 angles selected by each method under consideration for an example image.

3.2 Construction of the prior covariance Σ_{xx}

Now we describe the construction of the Gaussian prior covariance $\Sigma_{xx} \in \mathbb{R}^{d_x \times d_x}$ over reconstructions. We consider a range of models, building from very simple models to flexible data-driven ones that allows for adaptive design.

Isotropic model. The simple choice $\Sigma_{xx} = \sigma_x^2 I_{d_x}$ assumes uncorrelated pixels, and it implies a ridge regulariser for the reconstruction, which is known to perform poorly in imaging.

Matern-1/2 Process. Antorán et al. (2022) employ the Matern-1/2 covariance $[\Sigma_{xx}]_{ij,i'j'} = \sigma_x^2 \exp(-\ell^{-1}\sqrt{(i-i')^2 + (j-j')^2})$, where i, j index the pixel locations in the image x, as a surrogate for TV. With the hyperparameters σ_x^2 and ℓ properly chosen, the prior samples and posterior inferences closely match those obtained with an intractable TV prior.

Linearised deep image prior (Barbano et al., 2022; Antorán et al., 2022). This data-driven prior is constructed by first fitting a DIP model on the measurements taken during the pilot scan with eq. (4), and then adopting a linear model on the basis expansion given by the Jacobian of the trained U-net $x(\cdot)$ with respect to θ evaluated at the optimal point $\theta^\star$, i.e. $\nabla_\theta x(\theta)_{|\theta=\theta^\star} =: \mathrm{J} \in \mathbb{R}^{d_x \times d_\theta}$ (Immer et al., 2021b). The resulting prior over x is given by

$$x = \mathrm{J}\theta, \quad \theta \sim \mathcal{N}(0, \Sigma_\theta) \quad \text{and thus} \quad x \sim \mathcal{N}(0, \mathrm{J}\Sigma_\theta \mathrm{J}^\top).$$

The covariance $\Sigma_{xx} = \mathrm{J}\Sigma_\theta \mathrm{J}^\top$ incorporates information about the pilot measurements through the features J. It assigns higher prior variance being near the edges in the reconstruction, cf. fig. 1, which are most sensitive to a change in U-net parameters. The covariance $\Sigma_\theta \in \mathbb{R}^{d_\theta \times d_\theta}$ weights different Jacobian entries. We consider two different structures for Σ_θ.

- The filter-wise block-diagonal matrix of Antorán et al. (2022) uses a separate prior for every block in the U-net (cf. appendix D.2). This choice uses a large number of hyperparameters. It risks overfitting to the pilot scan measurements resulting in uncertainty underestimation.
- The neural g-prior (Zellner, 1986; Antoran et al., 2022) is a maximally uninformative diagonal Gaussian prior with covariance matching the diagonal of U-net's inverse Fisher information matrix, denoted s^{-1}, scaled by a constant g (see appendix C for extended discussion). That is

$$\Sigma_\theta = g \cdot s^{-1}\mathrm{I}, \quad s = d_{y^{(t)}}^{-1} \sum_{i=1}^{d_y} ([\mathrm{A}^{(t)}\mathrm{J}]_i)^2 \in \mathbb{R}^{d_\theta}, \quad \text{and we choose} \quad g = (d_{y^{(0)}} d_\theta)^{-1} \sum_{i=1}^{d_y} ((y_i^{(0)})^2 - \sigma_y^2),$$

where $[\mathrm{AJ}]_i$ refers to the ith row of the matrix AJ. Computing s does not require measurement values and we update it every 5 acquired angles. We compute g once using the measurements from the pilot scan. Our choice of s ensures that the Jacobian entries corresponding to all U-net weights contribute equally to the marginal prior variance over measurements. Our choice of g ensures this marginal variance is equal to the empirical second moment of pilot measurements.

All models discussed have a number of free parameters $\sigma_y^2, \sigma_x^2, \ell, \Sigma_\theta$, which we choose to maximise the model evidence given the pilot scan measurements. See appendix B for details.

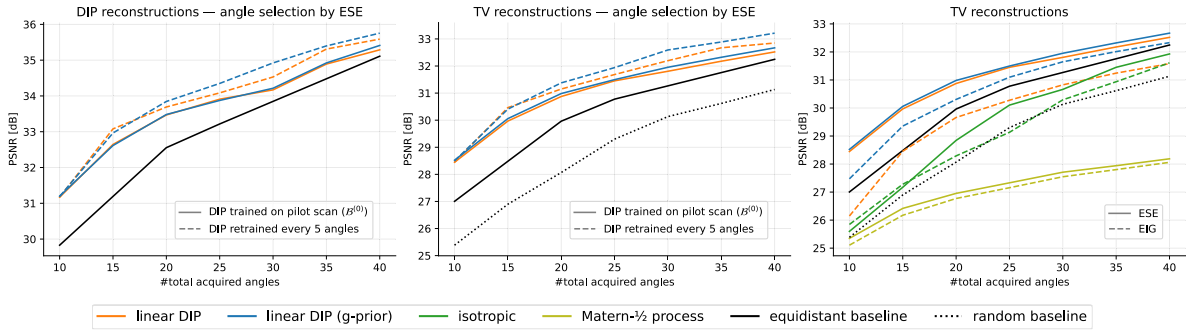

Figure 3: Reconstruction PSNR vs n. angles scanned, averaged across 30 images (5% noise).

4. Experiments and analysis

We simulate CT measurements y from 128×128 pixel images displaying rectangles of random proportions aligned along a randomly chosen "preferential" direction (see fig. 2 and fig. 4). The forward operator A is the discrete Radon transform, and either 5% or 10% white noise is added to the measurement y. We divide the range $[0°, 180°)$ into 200 selectable angles (i.e. $|\mathcal{B}_a| = 200$). The pilot scan measures at 5 equidistant angles, on which we fit all models' hyperparameters and the linearised DIP's U-net (see appendix B). Then, we apply the methods in section 3.1 to produce designs consisting of 35 additional angles. For every 5 acquired angles, we evaluate reconstruction quality using both the DIP (i.e. eq. (4)), and the traditional TV regularised approach (i.e. eq. (3)). We include equidistant and random angle selection as baselines. On an NVIDIA A100 GPU, a full linearised DIP acquisition step with $K = 3000$ samples takes 9 seconds and the full design takes 5 minutes. Appendix D contains full experimental details.

For the **linearised DIP**, we consider training our U-net and prior hyperparameters only on the pilot scan, and also retraining every 5 angles. Figure 2 shows both approaches can identify and prioritise the preferential direction, leading to reconstructions that *outperform the equidistant angle baseline by over 1.5 dB* in the range of $[10, 15]$ angles (see fig. 3). During this initial stage, the linearised DIP requires roughly *30% less scanned angles* to match the equidistant baseline's performance. The performance gap decreases as we select more angles, although linearised DIP remains more efficient even after 40 angles. Retraining the U-net provides most benefits in the large angle regime. It increases focus on preferential directions and consistently provides gains >0.5dB after 20 angles. All gains over the equidistant baseline are obtained with both DIP (i.e. eq. (4)) and traditional TV regularised reconstruction (i.e. eq. (3)). In the high 10% noise setting, gains from experimental design are smaller, but still significant (see appendix E).

The **isotropic and Matern-1/2** models' uncertainty estimates are independent of the pilot measurements. These models prioritise clustered sets of oblique angles which maximise the length of quanta trajectories in the image. They perform similar to or worse than random. We explore this negative result in appendix E, finding it due to overfitting of hyperparameters.

ESE outperforms EIG across models. For the linearised DIP, this gap is smaller when using the g-prior. We hypothesise that model misspecification and hyperparameter overfitting may result in poor measurement covariance estimates, in turn degrading EIG estimates.

5. Conclusion and future work

Our results suggest that dependence on the measurement data, i.e. adaptivity, is key to outperforming equidistant angle selection in CT reconstruction, a notoriously difficult task (Shen

et al., 2022; Helin et al., 2022). Distinctly from previous work, our methods only necessitate a pilot scan instead of being fully online, increasing applicability. We observe the largest gains in the 10 to 20 angle regime, where our designs reduce the angle requirement by roughly 30% without loss of reconstruction quality. This is true for both traditional TV-regularised and DIP reconstructions. In future, we aim to apply linearised DIP designs to real measurements.

Acknowledgments

We would like to thank Eric Nalisnick and Mark van der Wilk for helpful discussions. R.B. acknowledges support from the i4health PhD studentship (UK EPSRC EP/S021930/1), and from The Alan Turing Institute (UK EPSRC EP/N510129/1). The work of B.J. is partially supported by UK EPSRC grants EP/T000864/1 and EP/V026259/1. J.L. is funded by the German Research Foundation (DFG; GRK 2224/1), and additionally acknowledges support from the DELETO project funded by the Federal Ministry of Education and Research (BMBF, project number 05M20LBB). J.A. acknowledges support from Microsoft Research, through its PhD Scholarship Programme, and from the EPSRC. J.A. also acknowledges travel support from ELISE (GA no 951847). This work has been performed using resources provided by the Cambridge Tier-2 system operated by the University of Cambridge Research Computing Service (http://www.hpc.cam.ac.uk) funded by EPSRC Tier-2 capital grant EP/T022159/1.

References

J. Antoran and A. Miguel. Disentangling and learning robust representations with natural clustering. In *2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, pages 694–699, 2019.

Javier Antorán, James Urquhart Allingham, and José Miguel Hernández-Lobato. Depth uncertainty in neural networks. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/781877bda0783aac5f1cf765c128b437-Abstract.html.

Javier Antorán, Umang Bhatt, Tameem Adel, Adrian Weller, and José Miguel Hernández-Lobato. Getting a CLUE: A method for explaining uncertainty estimates. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=XSLF1XFq5h.

Javier Antoran, James Urquhart Allingham, David Janz, Erik Daxberger, Eric Nalisnick, and José Miguel Hernández-Lobato. Linearised laplace inference in networks with normalisation layers and the neural g-prior. In *Fourth Symposium on Advances in Approximate Bayesian Inference*, 2022. URL https://openreview.net/forum?id=uUH8x-h9zdB.

Javier Antorán, Riccardo Barbano, Johannes Leuschner, José Miguel Hernández-Lobato, and Bangti Jin. A probabilistic deep image prior for computational tomography. Preprint, arXiv:2203.00479, 2022.

Javier Antorán, David Janz, James Urquhart Allingham, Erik A. Daxberger, Riccardo Barbano, Eric T. Nalisnick, and José Miguel Hernández-Lobato. Adapting the linearised laplace model evidence for modern deep learning. *CoRR*, abs/2206.08900, 2022. doi: 10.48550/arXiv.2206.08900. URL https://doi.org/10.48550/arXiv.2206.08900.

Daniel Otero Baguer, Johannes Leuschner, and Maximilian Schmidt. Computed tomography reconstruction using deep image prior and learned reconstruction methods. *Inverse Problems*, 36(9):094004, 2020.

Riccardo Barbano, Johannes Leuschner, Maximilian Schmidt, Alexander Denker, Andreas Hauptmann, Peter Maaß, and Bangti Jin. Is deep image prior in need of a good education? *arXiv preprint arXiv:2111.11926*, 2021.

Riccardo Barbano, Javier Antorén, José Miguel Hernández-Lobato, and Bangti Jin. A probabilistic deep image prior over image space. In *Fourth Symposium on Advances in Approximate Bayesian Inference*, 2022. URL https://openreview.net/forum?id=qtFPfwJWowM.

Martin Burger, Andreas Hauptmann, Tapio Helin, Nutti Hyvönen, and Juha-Pekka Puska. Sequentially optimized projections in x-ray imaging. *Inverse Problems*, 37(7):075006, 2021. doi: 10.1088/1361-6420/ac01a4.

Thorsten M Buzug. Computed tomography. In *Springer handbook of medical technology*, pages 311–342. Springer, 2011.

Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, 10(3):273 – 304, 1995. doi: 10.1214/ss/1177009939. URL https://doi.org/10.1214/ss/1177009939.

Antonin Chambolle, Vicent Caselles, Daniel Cremers, Matteo Novaga, and Thomas Pock. An introduction to total variation for image analysis. In *Theoretical foundations and numerical methods for sparse recovery*, pages 263–340. de Gruyter, 2010.

Erik A. Daxberger, Eric T. Nalisnick, James Urquhart Allingham, Javier Antorán, and José Miguel Hernández-Lobato. Bayesian deep learning via subnetwork inference. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 2510–2521. PMLR, 2021. URL http://proceedings.mlr.press/v139/daxberger21a.html.

V. V. Fedorov. *Theory of Optimal Experiments*. Academic Press New York, 1972. ISBN 0122507509.

Chi Feng. *Optimal Bayesian experimental design in the presence of model error*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 2015.

Adam Evan Foster. *Variational, Monte Carlo and Policy-Based Approaches to Bayesian Experimental Design*. PhD thesis, University of Oxford, 2021.

Tapio Helin, Nuutti Hyvönen, and Juha-Pekka Puska. Edge-promoting adaptive Bayesian experimental design for X-ray imaging. *SIAM J. Sci. Comput.*, 44(3):B506–B530, 2022. ISSN 1064-8275. doi: 10.1137/21M1409330.

Yehuda Hoffman and Erez Ribak. Constrained realizations of Gaussian fields: a simple algorithm. *Astrophys. J. Lett.*, 380:L5–L8, 1991. doi: 10.1086/186160.

Alexander Immer, Matthias Bauer, Vincent Fortuin, Gunnar Rätsch, and Mohammad Emtiyaz Khan. Scalable marginal likelihood estimation for model selection in deep learning. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 4563–4573. PMLR, 2021a. URL http://proceedings.mlr.press/v139/immer21a.html.

Alexander Immer, Maciej Korzepa, and Matthias Bauer. Improving predictions of bayesian neural nets via local linearization. In Arindam Banerjee and Kenji Fukumizu, editors, *The 24th International Conference on Artificial Intelligence and Statistics, AISTATS 2021, April 13-15, 2021, Virtual Event*, volume 130 of *Proceedings of Machine Learning Research*, pages 703–711. PMLR, 2021b. URL http://proceedings.mlr.press/v130/immer21a.html.

Kazufumi Ito and Bangti Jin. *Inverse problems: Tikhonov theory and algorithms*, volume 22. World Scientific, 2014.

Jaehoon Lee, Samuel Schoenholz, Jeffrey Pennington, Ben Adlam, Lechao Xiao, Roman Novak, and Jascha Sohl-Dickstein. Finite versus infinite neural networks: an empirical study. *Advances in Neural Information Processing Systems*, 33:15156–15172, 2020.

David J. C. Mackay. Information-based objective functions for active data selection. *Neural Computation*, 4:590–604, 1992a.

David John Cameron Mackay. *Bayesian Methods for Adaptive Models*. PhD thesis, USA, 1992b. UMI Order No. GAX92-32200.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241. Springer, 2015.

Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268, 1992.

Matthias W. Seeger. On the submodularity of linear experimental design. Technical report, 2009. URL http://infoscience.epfl.ch/record/175483.

Matthias W. Seeger and Hannes Nickisch. Large scale bayesian inference and experimental design for sparse linear models. *SIAM J. Imaging Sci.*, 4(1):166–199, 2011. doi: 10.1137/090758775.

Ziju Shen, Yufei Wang, Dufan Wu, Xu Yang, and Bin Dong. Learning to scan: A deep reinforcement learning approach for personalized scanning in ct imaging. *Inverse Problems and Imaging*, 16(1):179–195, 2022.

Andrey N. Tikhonov and Vasiliy Y. Arsenin. *Solutions of ill-posed problems*. V. H. Winston & Sons, Washington, D.C.: John Wiley & Sons, New York, 1977. Translated from the Russian, Preface by translation editor Fritz John, Scripta Series in Mathematics.

Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. *Int. J. Comput. Vis.*, 128(7):1867–1888, 2020. ISSN 1573-1405. doi: 10.1007/s11263-020-01303-4.

James T. Wilson, Viacheslav Borovitskiy, Alexander Terenin, Peter Mostowsky, and Marc Peter Deisenroth. Pathwise conditioning of gaussian processes. *J. Mach. Learn. Res.*, 22:105:1–105:47, 2021. URL http://jmlr.org/papers/v22/20-1260.html.

Arnold Zellner. *On assessing prior distributions and Bayesian regression analysis with g prior distributions*, volume 6 of *Studies in Bayesian Econometrics and Statistics*, pages 233–243. Elsevier, 1986.

Appendix A. Discussion on acquisition objectives

The descriptions that follow are well known within the experimental design community but may be of interest to readers with a background in CT. Thus we describe these facts for the convenience of readers, and refer readers to Fedorov (1972); Chaloner and Verdinelli (1995) for a comprehensive introduction to experimental design and to Mackay (1992a) for a Bayesian perspective on experimental design.

EIG quantifies the information (in nats) we expect to gain by observing the detector elements' measurements for an angle or set of angles (Mackay, 1992a). Since our experiments employ greedy angle selection, we derive EIG for measurements at a single angle β. The generalisation to the multi-angle setting is straightforward. EIG is the expected decrease in posterior entropy from observing the detector elements' measurements at β:

$$\mathrm{EIG} = \mathrm{H}(x|y^{(t-1)}) - \mathbb{E}_{p(y^\beta|y^{(t-1)})}[\mathrm{H}(x|y^{(t-1)}, y^\beta)],$$

where we take an expectation over the new measurement y^β, since EIG is computed before the measurement y^β is made. For this, we use the posterior predictive distribution $p(y^\beta|y^{(t-1)})$ given our previous measurements $y^{(t-1)}$

$$p(y^\beta|y^{(t-1)}) = \int p(y^\beta|x)p(x|y^{(t-1)})\,\mathrm{d}x.$$

For the linear-Gaussian case, this integral can be evaluated in closed form, although this will not be necessary for our purposes.

EIG is also equal to the mutual information $MI(x, y^\beta|y^{(t-1)})$ between the reconstruction x and the new measurement y^β conditional on the previous measurements $y^{(t-1)}$, giving an interpretation as aiming to select the angle β most informative towards the reconstruction. For fixed model hyperparameters, EIG is always greater or equal than 0 since making additional measurements cannot increase the uncertainty in the reconstruction.

The entropy of a multivariate Gaussian $\mathcal{N}(\mu, \Sigma)$ is $\mathrm{H} = \frac{1}{2}\mathrm{logdet}(\Sigma) + \frac{d}{2}(\log(2\pi) + 1)$. For a fixed dimensionality d, the second term is constant across design steps and thus we only need to focus on the log determinant. The entropy does not depend on the distribution mean but only its covariance. Thus, taking $y^{(t)} = [y^{(t-1)}, y^\beta]$, we can write

$$\mathrm{EIG} = \mathrm{logdet}(\Sigma_{x|y^{(t-1)}}) - \mathrm{logdet}(\Sigma_{x|y^{(t)}}).$$

Since the covariance $\Sigma_{x|y^{(t-1)}}$ does not depend on the new angle choice β, maximising EIG is equivalent to choosing the angle which minimises the updated covariance log-determinant $\mathrm{logdet}(\Sigma_{x|y^{(t)}})$. Hence, the EIG objective for linear models is also known as the D(eterminant)-optimal criterion.

We can obtain a more convenient expression for EIG by noting the sequential nature of Bayesian learning; when data is observed, the prior is updated to a posterior. This posterior represents the updated beliefs and, as such, acts as a prior distribution for further inferences

$$p(x|y^{(t)}) = \frac{p(y^\beta|x)p(x|y^{(t-1)})}{p(y^\beta|y^{(t-1)})}.$$

For conjugate Gaussian-linear models, we can apply this principle to obtain the posterior covariance at time t from the covariance at time $t-1$ using the matrix determinant lemma

$$\operatorname{logdet}(\Sigma_{x|y^{(t)}}) = -\operatorname{logdet}(\Sigma^{-1}_{x|y^{(t-1)}}) - \operatorname{logdet}(\sigma_y^{-2}\mathrm{I}) - \operatorname{logdet}(\sigma_y^2\mathrm{I} + \mathrm{A}_0^{(t)}\Sigma_{x|y^{(t-1)}}\mathrm{A}_0^{\top,(t)}).$$

Thus, we have

$$\begin{aligned}
\mathrm{EIG} &= \operatorname{logdet}(\Sigma_{x|y^{(t-1)}}) - \operatorname{logdet}(\Sigma_{x|y^{(t)}}) \\
&= \operatorname{logdet}(\Sigma_{x|y^{(t-1)}}) - [-\operatorname{logdet}(\Sigma^{-1}_{x|y^{(t-1)}}) - \operatorname{logdet}(\sigma_y^{-2}\mathrm{I}) - \operatorname{logdet}(\sigma_y^2\mathrm{I} + \mathrm{A}^\beta\Sigma_{x|y^{(t-1)}}(\mathrm{A}^\beta)^\top)] \\
&= -\operatorname{logdet}(\sigma_y^2\mathrm{I}) + \operatorname{logdet}(\sigma_y^2\mathrm{I} + \mathrm{A}^\beta\Sigma_{x|y^{(t-1)}}(\mathrm{A}^\beta)^\top) \\
&= \operatorname{logdet}(\sigma_y^2\mathrm{I} + \mathrm{A}^\beta\Sigma_{x|y^{(t-1)}}(\mathrm{A}^\beta)^\top) + C
\end{aligned}$$

where the constant $C = -\operatorname{logdet}(\sigma_y^2\mathrm{I})$ is independent of angle choice, yielding the objective we use for angle selection in practise.

The ESE objective in eq. (7) aims to minimise the squared prediction error in measurement space. Objectives of this kind are commonly known as (A)verage-optimal. However, ESE is A-optimal over measurement space y, not over image space x. ESE is crucially different from minimising the arguably more relevant expected squared reconstruction error, a more computationally expensive criterion. ESE can be understood as a naive simplification of EIG, by discarding correlations between detector pixels, making $\operatorname{logdet}(\mathrm{A}^\beta\Sigma_{x|y^{(t-1)}}(\mathrm{A}^\beta)^\top)$ match $\sum_{i<d_p}\log[\mathrm{A}^\beta\Sigma_{x|y^{(t-1)}}(\mathrm{A}^\beta)^\top]_{ii}$. Then, the order of log and sum are switched, something that will only be true if every element under the sum is the same. Having reached this point, since the log function is monotonic, it does not affect angle selection and the criterion matches the trace of $\mathrm{A}^\beta\Sigma_{x|y^{(t-1)}}(\mathrm{A}^\beta)^\top$.

Appendix B. Hyperparameter selection via model evidence maximisation

For the conjugate linear-Gaussian model, the model evidence can be computed in closed form

$$\log p(y) = \log\mathcal{N}(y; 0, \Sigma_{yy}) = -\frac{1}{2}\left(y^\top\Sigma_{yy}^{-1}y + \operatorname{logdet}(\Sigma_{yy})\right) + C$$
$$\text{with} \quad \Sigma_{yy} = \mathrm{A}\Sigma_{xx}\mathrm{A}^\top + \sigma_y^2 I_{d_y}$$

and $C = {}^{-d_y}/_2\log 2\pi$. This expression is straightforward to compute for the isotropic and Matern-${}^1/_2$ models. The linear solve against Σ_{yy} and log-determiant operations, while costly, are tractable to perform when the dimensionality of y is low. This is the case in our experimental setup, where we use the measurements from our 5 angle pilot scan $y^{(0)}$, specifically, $d_y = d_p \cdot d_{\mathcal{B}} = 5 \cdot 183 = 915$. We refer to Antorán et al. (2022) for discussion of efficient computation of the model evidence for the linearised DIP. For additional discussion on the motivation for the model evidence objective, its applications and pitfalls, we refer to Mackay (1992b); Immer et al. (2021a); Antorán et al. (2022).

Selecting prior hyperparameters with the model evidence is often claimed to be immune from overfitting due to the flexibility of the prior model being relatively low. However, when the number of measurements is small, e.g. after performing the pilot scan, overfitting is still possible. Indeed we observe the Matern-${}^1/_2$ model suffers due to this issue in our experiments. We further discuss this in appendix E.

The risk of overfitting is also high for the linearised DIP model of Barbano et al. (2022); Antorán et al. (2022). Here, the basis expansion is selected by training a U-net on the pilot measurements and the number of hyperparameters is twice the number of U-net blocks, making this prior class very flexible. This has motivated the use of the neural g-prior (Antoran et al., 2022), discussed in the following section.

Appendix C. Discussion on the neural g-prior

The neural g-prior $\Sigma_\theta = g \cdot s^{-1}\mathrm{I}$ was introduced by Antoran et al. (2022) as an approach to "normalise" the second moment of the Jacobian feature expansion analogously to standard data normalisation. This normalisation ensures that the Jacobian entries corresponding to all network weights contribute equally to the predictions at the train points, or in our case, to the predictions at the already measured angles. We refer to Antoran et al. (2022) for a full derivation.

Antoran et al. (2022) learn the variance scale g with the model evidence objective. However, it is well known that this procedure can overfit in the small-data regime. To prevent overfitting, in this work we choose g using the heuristic

$$g = (d_y d_\theta)^{-1} \sum_{i=1}^{d_y} ((y_i)^2 - \sigma_y^2).$$

This choice is made so that the marginal predictive variance averaged across measurement locations matches the empirical second moment of the observed targets, which we will denote $\mathbb{E}[y^2] = d_y^{-1} \sum_{i=1}^{d_y} y_i^2$. In other words, when using this prior over weights, our prior over measurements will have roughly the "right" variance. To see this, first recall

$$s = d_y^{-1} \sum_{i=1}^{d_y} ([\mathrm{AJ}]_i)^2$$

where $[\mathrm{AJ}]_i$ refers to the ith row of the matrix AJ and we will use $[\mathrm{AJ}]_{ij}$ to index each scalar entry of this matrix. We now expand the average marginal variance across measurements when using the neural g-prior

$$\begin{aligned}
d_y^{-1} \sum_{i=1}^{d_y} [\Sigma_{yy}]_{ii} &= d_y^{-1} \sum_{i=1}^{d_y} [\mathrm{AJ}(gs^{-1}I)\mathrm{A}^\top \mathrm{J}^\top]_{ii} + \sigma_y^2 \\
&= g d_y^{-1} \sum_{i=1}^{d_y} s_i^{-1} \sum_{j}^{d_\theta} [\mathrm{AJ}]_{ij}^2 + \sigma_y^2 \\
&= (\mathbb{E}[y^2] - \sigma_y^2) d_\theta^{-1} \sum_{i=1}^{d_y} \sum_{j=1}^{d_\theta} \frac{[\mathrm{AJ}]_{ij}^2}{\sum_{k=1}^{d_y} [\mathrm{AJ}]_{kj}^2} + \sigma_y^2 \\
&= (\mathbb{E}[y^2] - \sigma_y^2) d_\theta^{-1} \sum_{j=1}^{d_\theta} \frac{\sum_{i=1}^{d_y} [\mathrm{AJ}]_{ij}^2}{\sum_{k=1}^{d_y} [\mathrm{AJ}]_{kj}^2} + \sigma_y^2 \\
&= (\mathbb{E}[y^2] - \sigma_y^2) \left(d_\theta^{-1} \sum_{j=1}^{d_\theta} 1 \right) + \sigma_y^2 = \mathbb{E}[y^2],
\end{aligned}$$

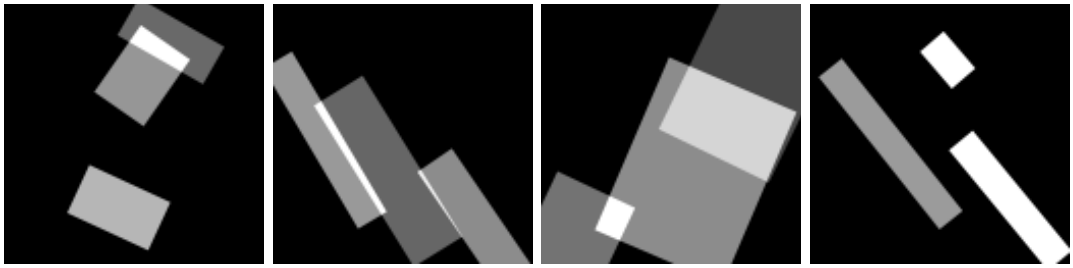

Figure 4: Examples of synthetic images.

showing the property.

For Antorán et al. (2022), model evidence optimisation is the most computationally costly step of inference. Avoiding model evidence optimisation speeds up inference and thus angle selection, making our proposed procedure more attractive for a real deployment.

Additionally, in fig. 3, we observe that the EIG objective performs best when combined with the neural g-prior. Arguably, EIG is a better motivated selection criterion than ESE but performs worse than equidistant selection when combined with all models except the g-prior linearised DIP. We hypothesise that model misspecification introduces error in our estimates of relative marginal variances and covariances across detector pixel measurements, in turn degrading the performance of EIG. ESE is less sensitive to these, as discussed in appendix A. Since the neural g-prior is a maximally uninformative prior, it somewhat mitigates model misspecification, improving the performance of EIG acquisition.

Appendix D. Full experimental setup

D.1 Dataset generation

We use a synthetic dataset comprising images of rectangles with randomised shape, orientation and intensity values, and simulate CT measurements by applying the forward operator $\mathrm{A} \in \mathbb{R}^{d_y \times d_x}$ and adding Gaussian noise with standard deviation of 5 % or 10 % of the average absolute value of the noiseless measurements $\mathrm{A}x$. Each image has resolution 128×128 px^2 and shows 3 superimposed rectangles, whose orientation is sampled from a single normal distribution with zero mean and standard deviation 2.86°. Thus, images in this class contain edges in roughly two perpendicular directions. Figure 4 shows example images from the dataset.

D.2 Implementation details for the linearised DIP

The key step of efficiently implementing the linearised DIP is the computation and Cholesky decomposition of the measurement covariance matrix Σ_{yy}. We describe this step in the following paragraphs and refer to Antorán et al. (2022), which we have followed in our implementation, for a complete set of details.

Computing the measurement covariance matrix $\Sigma_{yy}^{(t)}$ To assemble or multiply with $\Sigma_{yy}^{(t)}$, we employ matrix-free methods. Our workhorses are the matrix vector products $v_x^\top \Sigma_{xx}$ and $v_y^\top \Sigma_{yy}^{(t)}$ for $v_x \in \mathbb{R}^{d_x}$ and $v_y \in \mathbb{R}^{d_y}$. We efficiently compute these products through successive

	5 % noise				10 % noise			
#angles:	5	10-15	20-30	35-40	5	10-15	20-30	35-40
TV strength λ	$1e{-2}$	$3e{-3}$	$3e{-3}$	$3e{-3}$	$1e{-2}$	$1e{-2}$	$1e{-2}$	$3e{-3}$
iterations	60 000	30 000	10 000	10 000	60 000	30 000	10 000	10 000

Table 1: Hyperparameters for TV reconstruction. The values for λ are found by grid search on 10 validation images using 5, 10, 20 and 40 angles, and the numbers of iterations are chosen such that convergence is observed.

matrix vector product with the components of either Σ_{xx}, or $\Sigma_{yy}^{(t)}$, respectively. For instance,

$$v_y^\top \Sigma_{yy}^{(t)} = v_y^\top \left(\mathrm{A}^{(t)} \mathrm{J} \Sigma_\theta \mathrm{J}^\top \left(\mathrm{A}^{(t)} \right)^\top + \sigma_y^2 \mathrm{I} \right).$$

For any vector v_θ of appropriate size, we compute Jacobian vector products $v_\theta^\top \mathrm{J}^\top$ using forward mode automatic differentiation (AD) and $v_\theta^\top \mathrm{J}$ using backward mode AD. For the non g-prior model, we efficiently compute products with Σ_θ by exploiting its block diagonal structure. Since the g-prior covariance matrix is diagonal, computing products with it is straightforward.

Numerically stable sample generation with Matheron's rule eq. (8) Numerical instabilities can arise during the sample generation with the Matheron's rule due to the inversion of Σ_{yy}, updated via eq. (9). We resort to a simple regularisation strategy, which consists in adding to $\Sigma_{yy}^{(t)}$ a small diagonal element $\epsilon \mathrm{I}$, where ϵ is chosen from 1% to 10% of the diagonal mean, similarly to Lee et al. (2020).

D.3 Hyperparameters for TV and DIP reconstruction

The TV strength (i.e. λ) used in the DIP optimisation and the TV regularised objective, reported in table 1 and table 2, are found by grid search on 10 validation images. The DIP reconstruction quality from some images degrades when using many iterations Baguer et al. (2020), so an early stopping would be beneficial. For the PSNR evaluations of DIP reconstructions, we iterate for 30 000 steps and select the maximum PSNR for each image; this resembles the ideal early stopping by using the (in practice unknown) ground truth image, and is done in order to exclude the complexity of the stopping mechanism from our evaluations. For the DIP optimisations used for angle selection (i.e. the initial DIP on $\mathcal{B}^{(0)}$ and the DIPs retrained every 5 angles), the numbers of iterations in table 2 are used, which were found by grid search on 10 validation images.

	5 % noise				10 % noise			
#angles:	5	10-15	20-30	35-40	5	10-15	20-30	35-40
TV strength λ	$3e{-3}$	$3e{-3}$	$3e{-3}$	$1e{-3}$	$1e{-2}$	$1e{-2}$	$3e{-3}$	$3e{-3}$
iterations	19 000	9400	12 000	13 000	11 000	7500	12 000	7100

Table 2: Hyperparameters for DIP reconstruction (including TV regularisation). The values are found by grid search on 10 validation images using 5, 10, 20 and 40 angles.

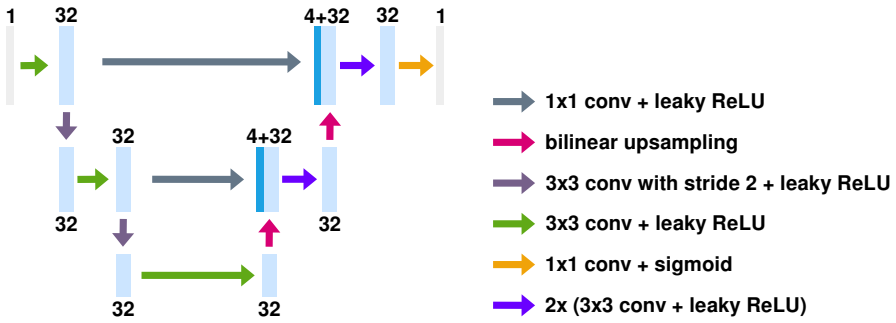

Figure 5: U-net architecture. Each light-blue box corresponds to a multi-channel feature map. The number of channels is set to 32 at every scale. The arrows denote the different operations.

Appendix E. Additional experimental results and analysis

In this section, we include additional experimental figures and discuss them.

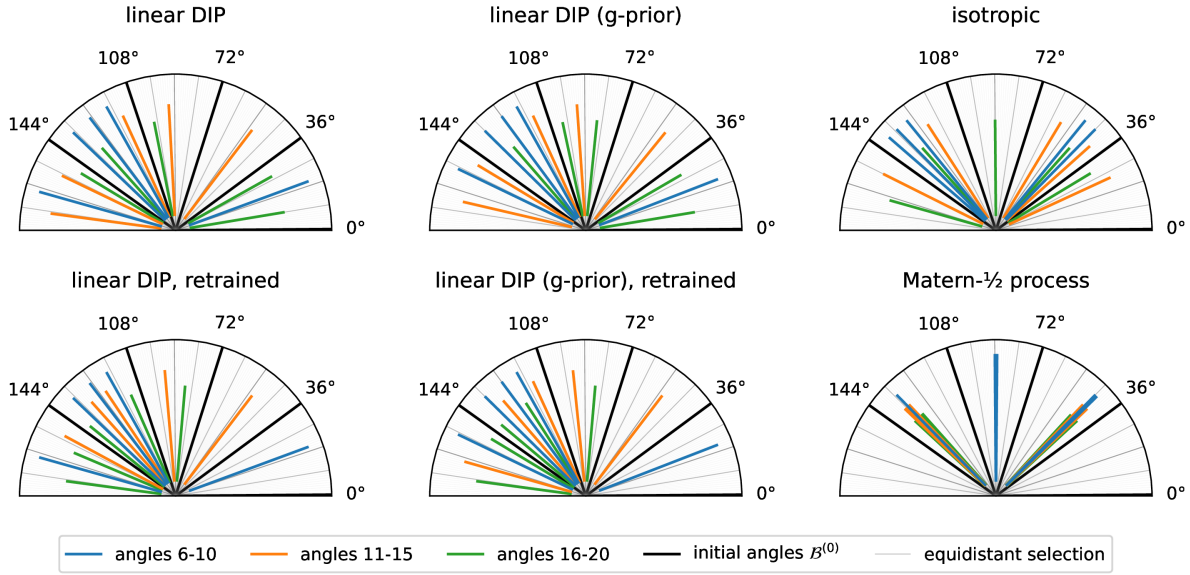

Figure 6: First 20 angles selected by each method under consideration for the example image shown in fig. 1

Figure 6 completes fig. 2 by showing the angles selected by all methods under consideration. Both linear DIP and linear DIP with g-prior choose very similar angles, with the g-prior resulting in a very slightly more diverse angle set. Retraining the linearised DIP every 5 angles to update the basis expansion results in a stronger focus on angles close to the preferential direction. As expected, the differences with the non-retrained DIP are more pronounced for later selected angles (i.e. angles 16-20).

The Matern-1/2 model concentrates its selection on oblique angles much more strongly than the isotropic model. This results in a very non-diverse angle set which achieves very poor performance. To understand why this happens we first remark that the Matern-1/2 model generalises the isotropic model and the two are equal when the lengthscale is set to $\ell = 0$. We investigate the hyperparameters chosen by the model evidence for the Matern-1/2 model and find that for all images the lengthscale is in the range [40-70]. This value is very large relative to the size of the image (128×128) and represents an assumption that the reconstructed image has only 2 or 3 regions with different pixel intensity values. Under this assumption, only taking measurements at 3 different angles is justified.

We verify this explanation by examining the ESE scores assigned by the isotropic and Matern-1/2 models to the first 8 angles chosen in fig. 7 and fig. 8 respectively. The isotropic model chooses oblique angles. After each new angle is included in the updated operator $\mathrm{A}^{(t)}$, the predictive variance in a region spanning roughly 10° around the chosen angles decreases. This is the span of the detector elements. The uncertainty at other angles remains unchanged because the model assumes reconstruction pixels to be uncorrelated. By modelling correlations among detector

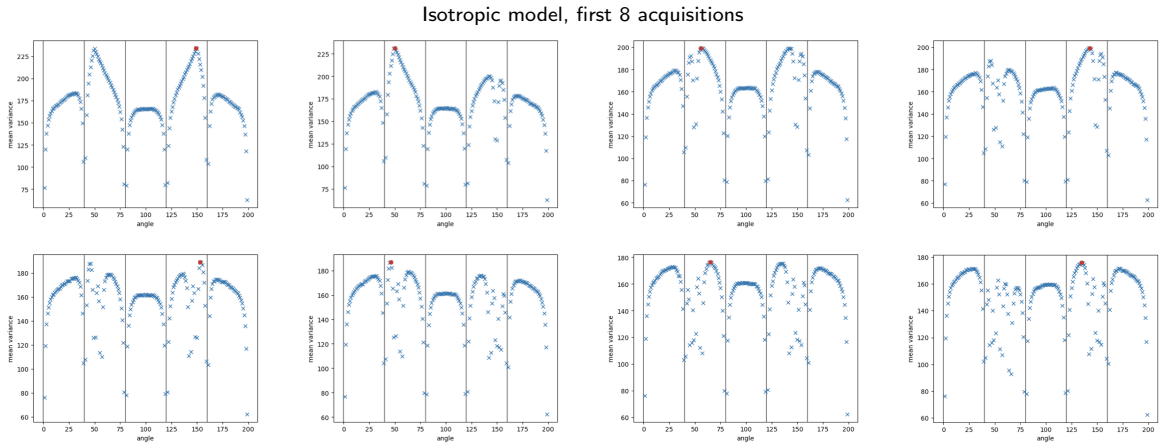

Figure 7: Variance assigned to each candidate angle during the first 8 design steps by our Isotropic model.

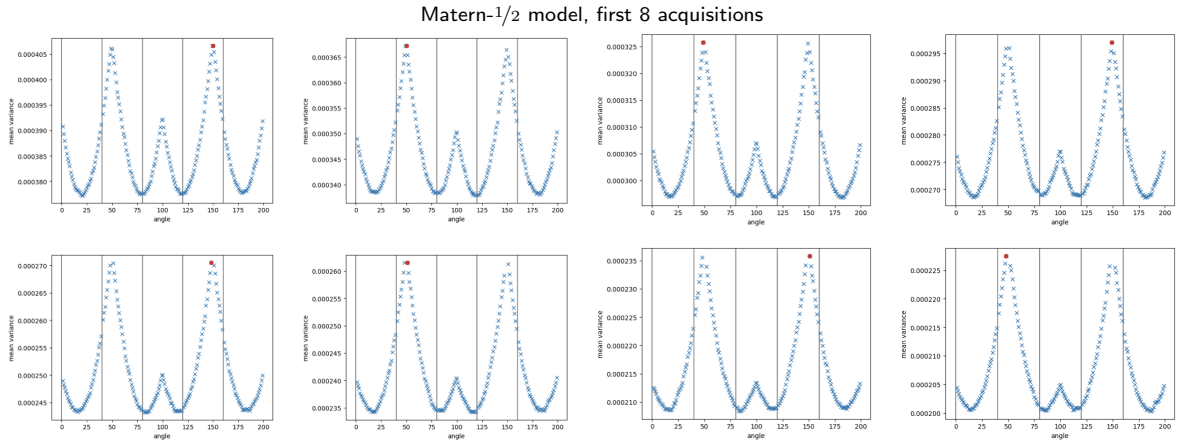

Figure 8: Variance assigned to each candidate angle during the first 8 design steps by our Matern-$1/2$ model.

pixels, each additional angle should reduce the Matern-$1/2$ model's uncertainty in a larger angle range (set via the lengthscale), promoting exploration. However, because the lengthcale, which has overfit the pilot measurements, is very large, each new angle introduced into the operator reduces the predictive variance of every angle almost equally. As a result, the relative assignment of predictive variance in angle space remains roughly constant throughout design steps, and all of the chosen angles become very similar to each other.

Although, it is well known that experimental design is very sensitive to the choice of prior (Feng, 2015; Foster, 2021), the ease with which the relatively very simple Matern-$1/2$ model can

overfit the degree to which this degrades performance was unexpected to us. In future work we will investigate alternative methods for setting model hyperparameters.

Figure 9 and fig. 10 show the variance assigned to each angle in the first 8 acquidition steps on an example image (first image from fig. 4) for the linearised DIP and the linearised DIP with g-prior, respectively. Although the angles selected by the two models are different, both prioritise similar angle regions.

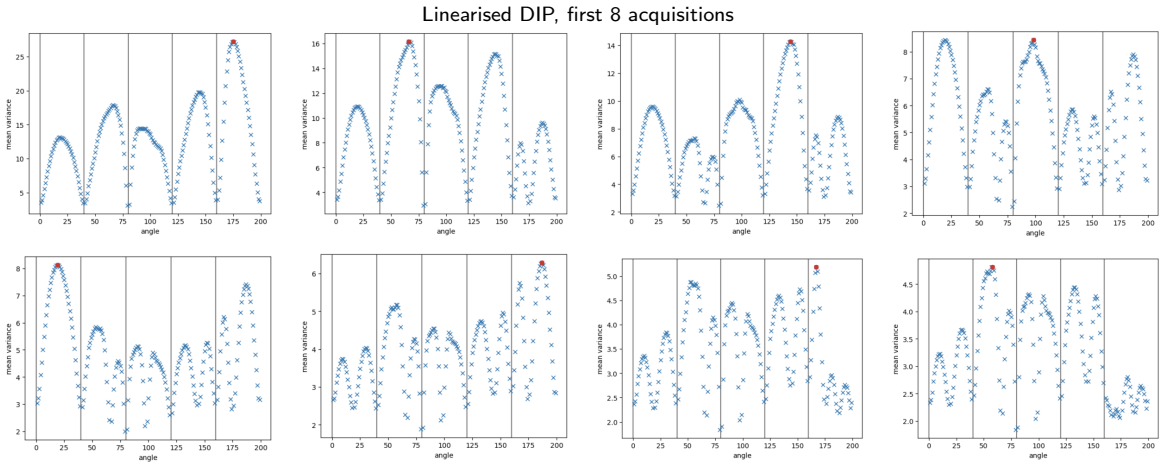

Figure 9: Variance assigned to each candidate angle during the first 8 design steps by our linearised DIP model.

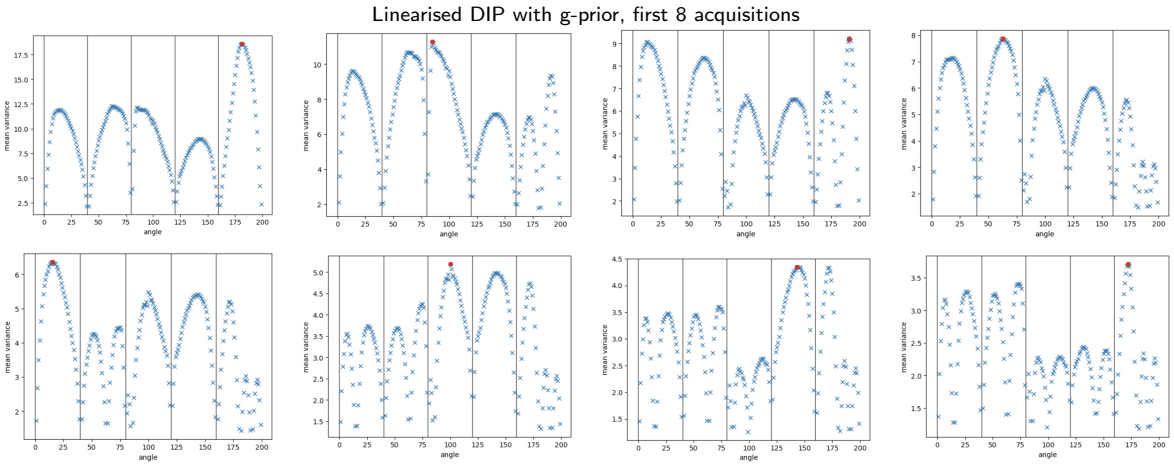

Figure 10: Variance assigned to each candidate angle during the first 8 design steps by our linearised DIP model with the g-prior.

Figure 11 is a more complete version of fig. 3, including the standard error. Given our 30 image runs, we can conclude that the linearised DIP provides a statistically significant improve-

ment over the equidistant baseline up to 20 selected angles. By retraining the DIP Jacobians every 5 angles, we can extend the significant improvements up to 35 scanned angles. In future we aim to make these statements stronger by running more experiments.

Figure 12 shows our findings on measurement data simulated adding 10% noise. The gains from experimental design are slightly reduced in the noisier setting, although the conclusions remain the same. From the EIG expression eq. (6), we can see that noisier measurements should push our score assignment to be more uniform across angles and thus closer to the equidistant baseline.

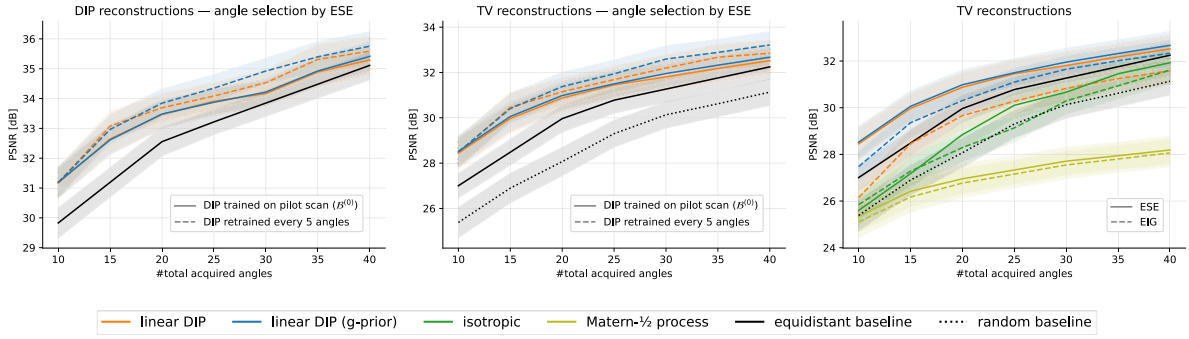

Figure 11: Reconstruction PSNR vs n. angles scanned, averaged across 30 images (5% noise).

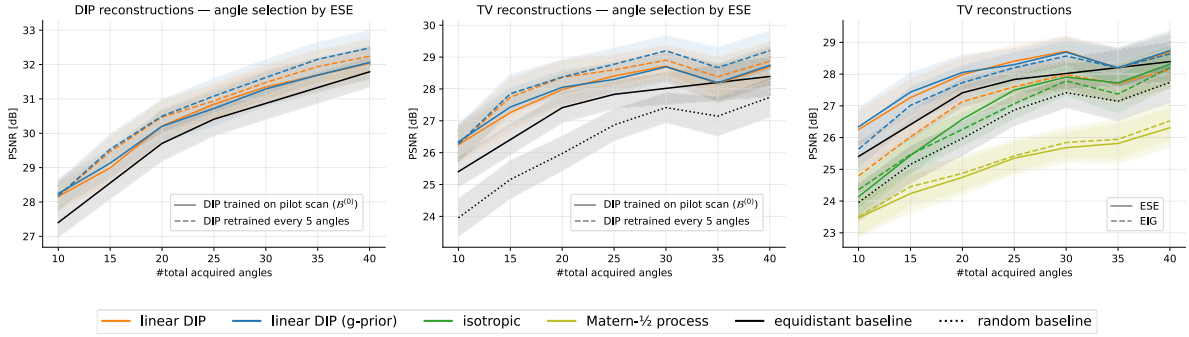

Figure 12: Reconstruction PSNR vs n. angles scanned, averaged across 30 images (10% noise).