Multi-Scale Learned Iterative Reconstruction

Andreas Hauptmann*, Jonas Adler*, Simon Arridge, and Ozan Öktem

***Abstract*—Model-based learned iterative reconstruction methods have recently been shown to outperform classical reconstruction methods. Applicability of these methods to large scale inverse problems is however limited by the available memory for training and extensive training times. As a possible solution to these restrictions we propose a multi-scale learned iterative reconstruction algorithm that computes iterates on discretisations of increasing resolution. This procedure does not only reduce memory requirements, it also considerably speeds up reconstruction and training times, but most importantly is scalable to large scale inverse problems, like those that arise in 3D tomographic imaging. Feasibility of the proposed method to speed up training and computation times in comparison to established learned reconstruction methods in 2D is demonstrated for low dose computed tomography (CT), for which we utilise the data base of abdominal CT scans provided for the 2016 AAPM low-dose CT grand challenge.**

I. Introduction

Computed tomography (CT) is an imaging technology where the interior anatomy of a subject is computed from a series of X-ray radiographs acquired by radiating the subject from different directions. CT has had a profound impact on medical practice and it is now an indispensable technology in a wide spectrum of clinical applications. It has also been essential for advancing our understanding of disease in medical research.

There are however risks that come with CT imaging, especially when used for screening. CT relies on repeatedly exposing a patient to ionising X-rays and hence there is an ongoing effort to minimise the total dose delivered to a patient during a CT scan. For that purpose, low dose CT protocols could be employed, which imply that less X-ray photons are measured and consequently implies that acquired data has a lower Signal-to-Noise ratio. Standard reconstruction techniques used in clinical practice, like filtered backprojection, are based on sampling theory and as such are not properly adapted to account for the statistical characteristics of measured data with high noise level. Hence, applying these schemes on low-dose CT data will produce sub-optimal images and consequently prevents low dose protocols to be widely adapted.

Over the years, a wide range of reconstruction methods have been developed that better account for the aforementioned statistical properties in low-dose CT scans. Among these, the most powerful and flexible have been variational model-based methods [1], [2], [3]. These offer a plug-and-play architecture for reconstruction where a user provides a model for how data is generated in absence of noise (forward operator), a statistical model for noise in data, and a prior model for desired reconstructions. The forward operator together with the statistical model for data ensures consistency against measured data, whereas the prior mainly prevents over-fitting by penalising images that have 'irregular' behaviour. These methods can be also understood in a statistical setting, where one wants to determine the posterior density under a given measurement [4], [5], [6].

Variational model-based reconstruction methods are however computationally demanding since one needs to solve a large-scale optimisation problem. This becomes prohibitive in time-critical applications in large-scale CT, and especially so when the prior model is sparsity promoting. Yet another challenge lies in choosing an appropriate prior. [7], [8], [9], [10], [11]. Motivated by these shortcomings, recently there have been several efforts in using methods from Deep Learning for reconstruction, in which one can approximate the conditional mean image by training a deep neural network against supervised data using a squared ℓ^2-loss [12]. When properly adapted, these data driven approaches considerably outperform purely model based reconstruction techniques regarding *both* reconstruction quality and reconstruction speed.

One natural approach is to use Deep Learning to directly learn the mapping from data to image [13]. Such an approach scales poorly, it requires re-training when data acquisition changes, and it relies on access to huge amounts of training data. Hence, this is not a feasible approach for clinical CT where high quality training data is scarce. Another approach is to use Deep Learning as a post-processing tool to improve upon an initial reconstruction. This is computationally feasible as shown in [14], [15], [16], but such an approach is essentially limited by the information content of the initial reconstruction and the richness of a-priori information learned from training data, which potentially increases bias in the reconstruction.

Learned iterative reconstruction methods seek to overcome these drawbacks by combining Deep Learning with a model-based approach. More precisely, the idea is to use a deep neural network architecture for reconstruction that incorporates an explicit handcrafted forward operator and the adjoint of its derivative [12], [17], [18], [19], [20]. This can be done by unrolling a suitable fixed-point iteration that defines a reconstruction operator from a model-based approach. This yields further improvements to reconstruction quality as compared to direct or post-processing approaches mentioned before. Furthermore, including an explicit forward operator improves robustness and generalisability [21]. Additionally, it also reduces the amount

*Equal contribution.

A. Hauptmann is with the Research Unit of Mathematical Sciences; University of Oulu, Oulu, Finland and with the Department of Computer Science; University College London, London, United Kingdom.

J. Adler did this work at Elekta, Stockholm, Sweden and KTH – Royal Institute of Technology, Stockolm, Sweden. He is currently with DeepMind, London, UK.

S. Arridge is with the Department of Computer Science; University College London, London, United Kingdom.

O. Öktem is with is with the Department of Mathematics, KTH - Royal Institute of Technology KTH

of training data, since networks tend to have less parameters and the forward operator encodes a major portion of the relations in data that come from the acquisition geometry.

As already indicated, learned iterative reconstruction methods are typically trained in an end-to-end manner. Hence, the entire unrolled fixed-point scheme is treated as a single network and all its parameters are trained jointly. This provides an optimal set of network parameters under suitable optimisation procedures, but it also comes with two challenges. First, the memory footprint of storing and manipulating the network is too large for most single GPU configurations. Furthermore, during training the loss function is evaluated several times. Each of these involves evaluating the forward operator and its adjoint, or the adjoint of its derivative, which quickly leads to unreasonable training times. Hence, current learned iterative reconstruction algorithms do not scale well to large-scale and higher dimensions, such as fully 3D CT.

One possible solution to address these challenges is to adopt a greedy approach for training. Here each unrolled iteration in the network is trained separately [18]. In this way, training of each unrolled iterate and evaluation of the forward operator can be separated, thus rendering a training procedure feasible. On the other hand, such an approach clearly does not represent an optimal selection of network parameters as compared to jointly optimising over all network parameters for all unrolled iterates. Therefore, such a greedy approach renders a trained network for reconstruction that falls short in reconstruction quality compared to end-to-end schemes. Additionally, reconstruction times are still comparably slow due to multiple applications of the forward operator. The issue of computation times can be tackled by using faster approximate models [22], if available, but memory footprint remains an issue.

This paper proposes another approach for training learned iterative reconstruction methods that scales to demanding large-scale tomographic imaging problems. It is a multi-scale scheme that is motivated by the fact that the continuum forward operator can be discretised on various scales. In fact, the ray transform is known to be scale invariant [23], which defines the forward operator in CT, and this consistency across scales can be utilised for reconstruction [24], [25]. In particular, in our case each unrolled iterate in the network involves discretising the ray transform on a voxalised grid and the discretisation becomes increasingly fine as the unrolled iterates progress until the final resolution is achieved. Hence, the full high-resolution forward operator is only needed for the final unrolled iterate. Clearly, the approach is not limited to medical CT and readily applies to other tomographic modalities that involve the ray transform. Furthermore, it can be extended to any modality that arises as discretisation from a continuum model, such as MRI or even seismic imaging, in contrast to purely discrete problems.

This paper is structured as follows. In section II we review common approaches for learned reconstructions and discuss possible limitations for large-scale applications. In section III we introduce the proposed multi-scale schemes. In section IV we discuss implementation of the mutli-scale schemes, evaluate scalability, and test performance on human abdominal CT scans in comparison to established learned reconstruction approaches. In section V we discuss the results and mention possibilities for extension of the proposed methods. Some final conclusions are presented in section VI.

II. Learned reconstructions for tomographic imaging

In computed tomography we aim to reconstruct an image of the inside of a patient from X-ray measurements. Mathematically, this reconstruction task is an inverse problem where we seek to recover the unknown absorption coefficient $f^* \in X$ (image) from measured photons $g \in Y$ at the sensor (projection data or sinogram) where

$$g = \mathcal{A}(f^*) + \delta g. \tag{1}$$

Here, $\mathcal{A}\colon X \to Y$ is the forward operator, that is assumed to be known, and models how data is generated in absence of noise; $\delta g \in Y$ denotes noise in the observation.

In the following we will assume that $\mathcal{A}$ is a linear operator whose sampling is given by the data acquisition geometry, such as the fan-beam transform in 2D and cone-beam in 3D.

Reconstruction is typically an ill-posed task, so one needs to use noise-robust inversion procedures. Either by direct reconstruction algorithms, such as filtered backprojection (FBP), or by iterative algorithms that solve a variational problem

$$\hat{f} := \operatorname*{arg\,min}_{f \geq 0} \big\{ \mathcal{D}(f; g) + \alpha \mathcal{R}(f) \big\}. \tag{2}$$

Here, $f \mapsto \mathcal{D}(f; g)$ measures the goodness of fit against data g and a regularisation term $f \mapsto \mathcal{R}(f)$ with a weighting parameter $\alpha > 0$ ensures stability. These methods tend to perform well, but are ultimately limited by the expressiveness of the hand-crafted regularisation term $\mathcal{R}\colon X \to \mathbb{R}$. Recently, several groups have proposed to either combine direct reconstructions with a learning based post-processing or to learn an iterative algorithm. In the following we give a short overview of possible approaches and their advantages and disadvantages to motivate the formulation of our proposed algorithm.

A. Reconstruction and post-processing

A straightforward approach to use data driven methods in reconstruction is by post-processing an initial reconstruction. More precisely, let $\mathcal{A}^\dagger\colon Y \to X$ be an analytically known reconstruction operator that is proven to be robust. One can then train a convolutional neural network to remove reconstruction artefacts that arise from using $\mathcal{A}^\dagger$ [14], [15], [26]. These artefacts can be quite notable when data is highly noisy or under-sampled. The *inverse mapping* is now given as

$$\mathcal{A}^\dagger_\theta = \Lambda_\theta \circ \mathcal{A}^\dagger.$$

The advantage in this approach lies in the analytical knowledge of the reconstruction operator, and hence networks can be designed to exploit structure in reconstruction artefacts. For instance in spatio-temporal problems, if under-sampling artefacts are known to be incoherent in time, the network only needs to learn to combine the spatial information by a temporal interpolation [27]. On the other hand, for lower dimensional

problems, the capacity of the network is essentially limited by the richness of the training data [28], [29]. Clearly such an approach is computationally fast since it only requires a single operator evaluation. On the downside, large capacity networks tend to overfit to the training data and especially so when the training data is scarce. Furthermore, as shown in [12], [17], [18], [30] the results are clearly outperformed by learned iterative reconstruction algorithms that we next describe.

B. Learned iterative reconstructions

In learned iterative reconstruction schemes, neural networks are interlaced with evaluations of the forward operator $\mathcal{A}$, its adjoint $\mathcal{A}^*$, and possibly other hand-crafted operators. For example, a simple learned gradient-like scheme [12], [31] would be given by

$$f_{i+1} = \Lambda_{\theta_i}\big(f_i, \mathcal{A}^*(\mathcal{A}(f_i) - g)\big),\ i = 0, \dots, N-1. \tag{3}$$

This defines a reconstruction operator when stopped after N iterates:

$$\mathcal{A}_\theta^\dagger(g) := f_N \quad \text{where } \theta = (\theta_0, \dots, \theta_{N-1})$$

and initialisation $f_0 = \mathcal{A}^\dagger(g)$. Note that Λ_{θ_i} is a *learned updating operator* for the i:th iterate. The terminology 'gradient-like' comes from the following observation: if we consider minimising $\mathcal{D}(f; g) = \frac{1}{2}\big\|\mathcal{A}(f) - g\big\|_2^2$, then $\Lambda_\theta(f, h) := f - \theta h$ corresponds to a learned update in a gradient descent scheme.

The parameters θ in the reconstruction operator $\mathcal{A}_\theta^\dagger$ are learned by end-to-end supervised training. More precisely, assume one has access to supervised training data $(f^{(j)}, g^{(j)}) \in X \times Y$ where $g^{(j)} \approx \mathcal{A}(f^{(j)})$. Then an optimal parameter is found by

$$\min_\theta \frac{1}{m} \sum_{j=1}^{m} \mathrm{L}_\theta(f^{(j)}, g^{(j)})$$

where the loss function is given as

$$\mathrm{L}_\theta(f, g) := \big\|\mathcal{A}_\theta^\dagger(g) - f\big\|_X^2 \quad \text{for } (f, g) \in X \times Y.$$

Note here that computing the gradient of the loss function w.r.t. θ requires performing back-propagation through all of the unrolled iterates $i = 0, \dots, N-1$.

In gradient boosting, that follow the greedy training [18], the loss function is changed. Instead of looking for a reconstruction operator that is optimal end-to-end, we only require iterate-wise optimality. For the learned gradient scheme above, this amounts to the following loss function for the i:th unrolled iterate:

$$\mathrm{L}_{\theta_i}(f_i, g) = \Big\|\Lambda_{\theta_i}\big(f_i, \mathcal{A}^*(\mathcal{A}(f_i) - g)\big) - f\Big\|_X^2$$

where $f_i := \Lambda_{\theta_{i-1}}\big(f_{i-1}, \mathcal{A}^*(\mathcal{A}(f_{i-1}) - g)\big)$ and initialisation $f_0 = \mathcal{A}^\dagger(g)$. These schemes can be viewed as a greedy approach and consequently constitute an upper bound to end-to-end networks. Thus, in the following we seek for a possibility to utilise end-to-end networks for large-scale problems.

III. Multi-scale learned iterative reconstructions

The major limitations when employing learned iterative reconstruction methods for large problems are their prohibitive training times and memory requirements. This is mainly due to the fact that all iterations are performed at full resolution and hence require to evaluate the full scale forward operator for each iterate. To overcome this limitation we propose a multi-scale scheme.

A. Discretisation sequence

In the inverse problem in eq. (1), both the unknown image f^* and data g are considered as continuum objects, which in imaging are typically represented by real-valued functions defined on some domains. In reality discrete data is recorded through a measurement device and we can only compute a digitised version of the unknown f^*. By discretisation we refer loosely to the procedure for defining a finite dimensional version of eq. (1) that is given by the finite sampling of the data and the digitisation of f^*. Likewise, a *discretisation sequence* is a finite sequence of discretisations that start from a coarse discretisation and is successively refined towards the desired finest resolution. The refinement and coarsening of the discretisation is through specific up- and down-sampling schemes that will be defined later. Consequently, motivated by the discretisation invariance of the ray transform, we aim to iteratively increase the resolution of our reconstructions. For that purpose, let $S_0, \dots, S_N$ denote a fixed sequence of discretisations of X and Y that increase in resolution through subsequent up-sampling. In the following we will associate each iterate f_i with such a discretisation space S_i.

Stated more formally, a discretisation sequence is given by

$$S_i := (X_i \times Y_i) \quad \text{for } i = 0, \dots, N.$$

Here, $X_i \subset X$ is a finite dimensional subspace with $|X_i| < |X_{i+1}|$. Likewise, $Y_i \subset Y$ with $|Y_i| < |Y_{i+1}|$. Furthermore, let $\{f_i, g_i\} \in S_i$ denote the reconstructed image and data in each discretisation space. In the following we will need a projection operator in the data space $\pi_i \colon Y \to Y_i$, for $i = 0, \dots, N$, and an up-sampling operator in the image space $\tau \colon X_i \to X_{i+1}$, for $i = 0, \dots, N-1$. Whereas the projection operator maps the data into the respective discretisation space, the up-sampling operator maps the reconstruction in the i:th discretisation space to the subsequent finer one in the discretisation sequence.

The discretisation sequence $S_0, \dots, S_N$ defines as well a sequence of discretised versions of the inverse problem in eq. (1). More precisely, for each discretisation S_i we obtain the corresponding inverse problem of recovering $f_i^* \in X_i$ from finitely sampled data $g_i \in Y_i$ where

$$g_i = \mathcal{A}_i(f_i^*) + \delta g_i$$

with δg_i denoting the noise in data and $\mathcal{A}_i \colon X_i \to Y_i$ denoting the corresponding forward operator. Similarly, we have $\mathcal{A}_i^* \colon Y_i \to X_i$ for the adjoint and $\mathcal{A}_i^\dagger \colon Y_i \to X_i$ for the filtered backprojection on the discretisation space S_i. With these concepts we can now formulate the multi-scale iterative reconstructions schemes.

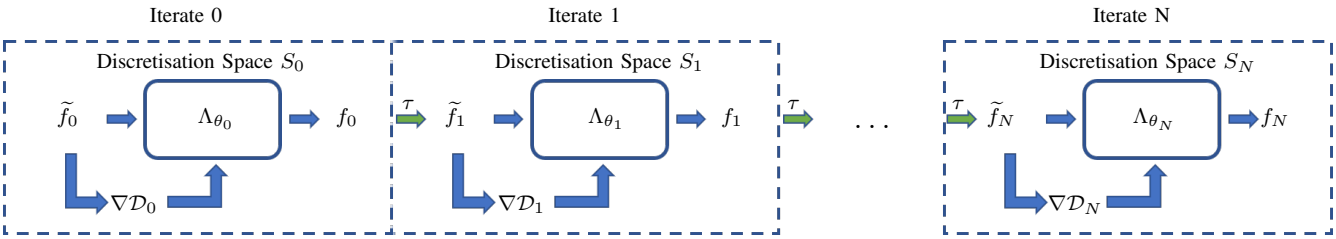

Fig. 1: Visualisation of the multi-scale learned gradient scheme (MS-LGS) as outlined in Algorithm algorithm 1. Each iteration is performed on their respective discretisation space, where the gradient $\nabla\mathcal{D}_i := \nabla\mathcal{D}_i(f_i; g)$ is computed and the update is performed by the network Λ_{θ_i}. After each update an upsampling by τ to the next finer space is performed until the final resolution S_N is achieved.

B. Multi-scale learned gradient descent

The underlying principle of the proposed multi-scale scheme is to start at the coarsest discretisation space S_0 and after each iterate we up-sample until we obtain the reconstruction in the final discretisation space in the desired full-resolution. This way each iterate has its own discretisation space and hence the number of iterations we perform is $N + 1$, equal to the number of discretisation spaces. Since we aim to train the algorithm end-to-end, this maximum number of iterations has to be fixed. For each iterate we then compute the gradient in the corresponding discretisation space $\nabla\mathcal{D}_i(f_i; g) \in S_i$ given by

$$\nabla\mathcal{D}_i(f_i; g) := \mathcal{A}_i^* \left(\mathcal{A}_i(f_i) - \pi_i(g)\right). \tag{4}$$

Following the structure of the classic learned gradient schemes eq. (3), we perform a learned update with the current reconstruction f_i and the corresponding gradient $\mathcal{D}_i(f_i; g)$, followed by an upsampling to the next finer resolution,

$$\begin{cases} f_i = \mathcal{G}_{\theta_i}\left(\widetilde{f}_i, \nabla\mathcal{D}_i(\widetilde{f}_i; g)\right) \\ \widetilde{f}_{i+1} = \tau(f_i). \end{cases}$$

The full multi-scale learned gradient scheme (MS-LGS) is summarised in Algorithm algorithm 1 and a schematic is illustrated in Figure fig. 1.

Algorithm 1 Multi-scale learned gradient scheme (MS-LGS)

for $i = 0, \dots, N$ **do**
  if $i = 0$ **then**
    $\widetilde{f}_0 \leftarrow \mathcal{A}_0^\dagger \pi_0(g)$
  else
    $\widetilde{f}_i \leftarrow \tau(f_{i-1})$
  end if
  $f_i \leftarrow \Lambda_{\theta_i}\left(\widetilde{f}_i, \nabla\mathcal{D}_i(\widetilde{f}_i; g)\right)$
end for
$f^* \leftarrow f_N$

1) Including a filtered gradient: Let us first note, that the up-sampling operator in each iteration restricts the high frequency components that can be present after up-sampling. Additionally, the normal operator $\mathcal{A}^*\mathcal{A}$ is known to be smoothing of order 1 [23], that means effectively any high frequency components in the final reconstruction can only be introduced by the network, similarly to the role of the regulariser in classical variational techniques. Thus, to complement the information for the network, we consider a version of MS-LGS with an additional filtered gradient that retains higher frequencies. That means we do not only compute the classic gradient $\nabla\mathcal{D}_i(f_i; g)$ in each iteration, but additionally a filtered version by substituting the adjoint with the filtered backprojection

$$\nabla^\dagger\mathcal{D}_i(f_i; g) := \mathcal{A}_i^\dagger \left(\mathcal{A}_i(f_i) - \pi_i(g)\right). \tag{5}$$

A similar approach has been studied earlier for classic iterative methods in [32]. In our case the filtered gradient will be computed additionally to the classic gradient eq. (4) and hence this will increase the computational cost by the application of one filtered backprojection in each step, but as can be seen later improves reconstruction quality. The resulting Multi-scale learned filtered gradient scheme (MS-LFGS) with the additional computation of the filtered gradient is described in Algorithm algorithm 2.

Algorithm 2 Multi-scale learned filtered gradient scheme (MS-LFGS)

for $i = 0, \dots, N$ **do**
  if $i = 0$ **then**
    $\widetilde{f}_0 \leftarrow \mathcal{A}_0^\dagger \pi_0(g)$
  else
    $\widetilde{f}_i \leftarrow \tau(f_{i-1})$
  end if
  $g_{\text{res}} \leftarrow \mathcal{A}_i(\widetilde{f}_i) - \pi_i(g)$
  $\nabla\mathcal{D}_i(\widetilde{f}_i; g) \leftarrow \mathcal{A}_i^*(g_{\text{res}})$
  $\nabla^\dagger\mathcal{D}_i(\widetilde{f}_i; g) \leftarrow \mathcal{A}_i^\dagger(g_{\text{res}})$
  $f_i \leftarrow \Lambda_{\theta_i}\left(\widetilde{f}_i, \nabla\mathcal{D}_i(\widetilde{f}_i; g), \nabla^\dagger\mathcal{D}_i(\widetilde{f}_i; g)\right)$
end for
$f^* \leftarrow f_N$

C. Computational cost

Concerning the total computational cost. Due to sub-sampling on the coarser discretisation spaces the computation of projections is essentially governed by the computations on the final resolution. If we assume that the computational cost

of evaluating the network Λ_{θ_i} is negligible in comparison to the forward and adjoint operator, then the total computational complexity is governed by the cost of the normal operator at the finest scale.

Formally, the total computational cost can be estimated as follows. Let us assume that the computational cost of the projection operators at each coarser scale reduces by 2^{-d} and we neglect any other computations, such as network evaluation and initialisation. Then, the total computational cost of computing the normal operator on all scales can be bounded by a geometric series

$$C_d := \sum_{k=0}^{\infty} \left(\frac{1}{2^d}\right)^k = \frac{1}{1-1/2^d}.$$

For $d = 2$ we have $C_2 = 4/3$ and $C_3 = 8/7$ for $d = 3$. Thus, the total computational cost of MS-LGS can be roughly estimated by C_d applications of the normal operator on the fine scale. For MS-LFGS this becomes C_d applications of the normal operator plus an application of filtered backprojection.

IV. Computational experiments

In the following we will examine the scalability of the proposed algorithms on simulated data. The reconstruction performance will then be tested with realistically generated data from human phantoms supplied for the 2019 AAPM Low Dose CT Grand Challenge. Before that, let us first address some details on the implementation.

A. Implementation

The proposed algorithms in section III provide a general framework for iterative reconstructions and hence for implementation some choices on down-sampling level, network architectures, and up-sampling operator need to be made. Let us first fix the number of iterations to $N + 1 = 5$. We will limit the following computational study to phantoms in $d = 2$ dimensions, but we want to emphasise that the presented framework does apply also to higher dimensions.

To create the discretisation spaces, we fix the resolution of the finest desired reconstruction space as $X_N = \mathbb{R}^{n\times n}$. The coarser resolutions are then obtained by reducing the resolution for each downsampling by a factor of 2 in each dimension. Thus, for 5 iterates the coarsest scale is obtained by 4 times downsampling and hence factor of 16 per dimension and hence the amount of data is reduced by a factor of 256 in 2D and 4096 in 3D. Similarly, for the data space we reduce the amount of angles by a factor of 2, the projection resolution is determined for each scale separately to fully cover the domain. The mapping π_i to the coarser scale is implemented by an area mean, the upsampling with τ is performed by bilinear interpolation.

The network architecture for Λ is based on a downscaled U-Net [33], that we will call here mini U-Net. This mini U-Net consists of only 2 scales (one max-pool layer), instead of the classic 4, and an initial channel depth of 32 on the first scale, similarly to what has been used in [22]. For increasing stability in the training we perform an explicit gradient descent as initialisation, then the updating Network in case of MS-LGS becomes

$$\begin{aligned}\Lambda_{\theta_i}\left(\widetilde{f}_i, \nabla\mathcal{D}_i(\widetilde{f}_i; g)\right) = f_i - \lambda_i \nabla\mathcal{D}_i(f_i, g) \\ + s_i \mathcal{U}_{\widetilde{\theta}_i}\left(f_i, \nabla\mathcal{D}_i(\widetilde{f}_i; g)\right).\end{aligned} \tag{6}$$

Here, λ_i is calculated from the operator norm to ensure a descent direction, $s_i \in \mathbb{R}$ is a learnable weighting that is initialised with 0 and $\mathcal{U}_{\widetilde{\theta}_i}$ denotes the mini U-Net. The learnable parameters are then $\theta_i = \{s_i, \widetilde{\theta}_i\}$. The formulation in eq. (6) can be understood as a residual mini U-Net in both input variables.

All algorithms, including reference methods, are implemented in Python using PyTorch [34] for the networks. The image and projection spaces are implemented with ODL (Operator Discretization Library) [35] and ASTRA [36] as back end for the ray transforms. Training details and parameter choices will be stated in the following sections.

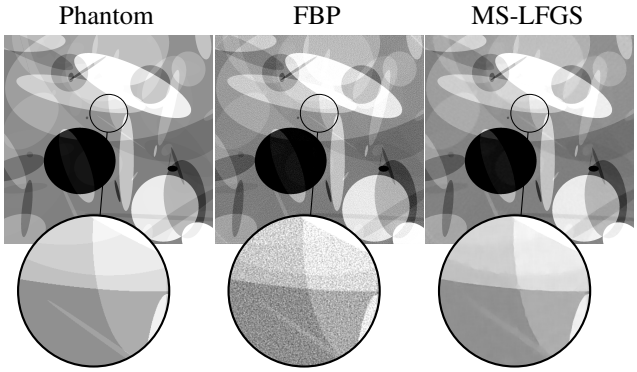

Fig. 2: Reconstruction of an ellipse phantom of size 1536^2 from 512 angles with 5% Gaussian random noise. (Left) Phantom used to create the data, (Middle) reconstruction by filtered backprojection, (Right) obtained reconstruction with MS-LFGS.

B. Scalability of reconstruction algorithms

We aim to examine the scalability of the proposed multi-scale algorithms in comparison to reference learned reconstruction methods. For comparison we choose post-processing with U-Net, following [15], and a learned gradient scheme (LGS) [12]. The learned gradient scheme is implemented consistent with the proposed MS-LGS algorithm, that means we use 5 iterations and mini U-Net. This can be seen as a subclass of MS-LGS, where all discretisation spaces are of the same resolution.

For the test data we chose phantoms consisting of randomly generated ellipses, see fig. 2. The data is produced by the ray transform

$$\mathcal{A}(f;\ell) = \int_\ell f(x)\,\mathrm{d}x,\ \ell \in \mathcal{M}, \tag{7}$$

where $\mathcal{M}$ is the set of lines defined by the measurement geometry. For this experiment we chose a fan beam geometry with 512 angles. The simulated measurement is corrupted by additional 5% of Gaussian noise.

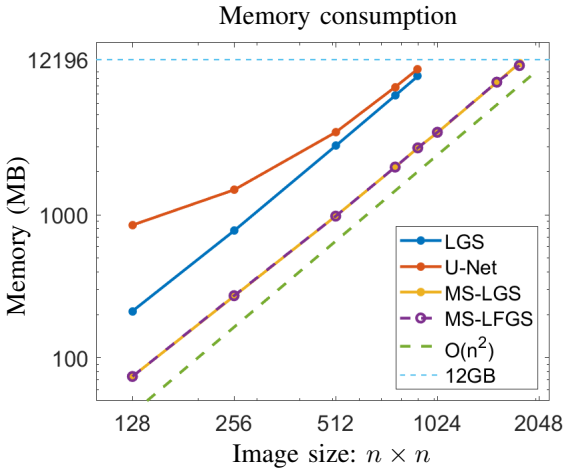

Fig. 3: Memory consumption of proposed algorithms and reference learned methods for simulated data of increasing size. Maximal available memory on the GPU was 12196MB.

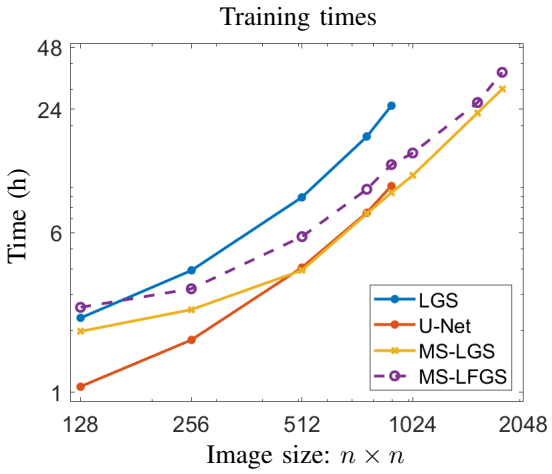

Fig. 4: Estimated training times for 50,000 iterations of proposed algorithms and reference learned methods for simulated data of increasing size.

Since the aim of this experiment is only to examine scalability, we have trained each network for 1000 iterations with one sample in each iteration and recorded the maximum memory consumption. The smallest phantom size was chosen as 128^2 and was increased until memory consumption exceeded the available memory on a single NVIDIA Titan XP GPU with 12GB memory, or more specifically 12196 MB. The resulting plot is shown in fig. 3.

Additionally, we have recorded the training time for 1000 iterations and extrapolated this to an estimated training time for 50,000 iterations for a full scale training. The resulting plot is shown in fig. 4.

A reconstruction obtained with the MS-LFGS for a resolution of 1536^2 is shown in fig. 2 in comparison to a reconstruction by filtered backprojection.

C. Application to human CT scans

In order to evaluate the reconstruction quality on a clinically relevant case, we simulate realistic measurement data from human abdomen CT scans provided by the Mayo Clinic for the

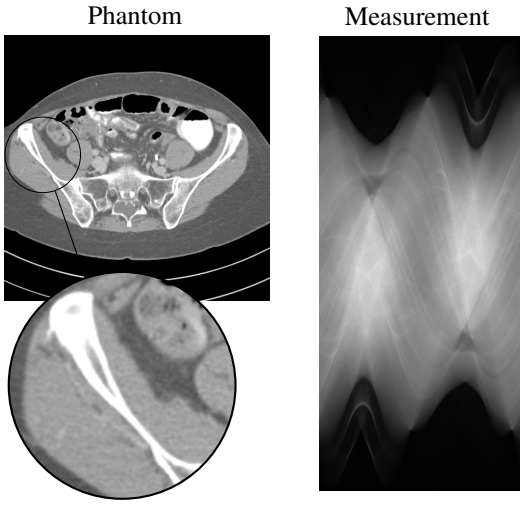

Fig. 5: Sample slice from the test patient windowed to $[-300, 300]$HU and the corresponding measurement data from 600 angles with 8000 photon counts.

2016 AAPM Low Dose CT Grand Challenge [37]. The data set consists of high-dose scans from 10 patients. We used the provided reconstructions with 3 mm slice thickness and image size 512×512. We divided the data into 9 patients for training, resulting in 2168 slices, and 1 patient for testing purposes with 210 slices.

For the data simulation, we used a fan-beam geometry with source to axis distance 500 mm and axis to detector distance 500 mm. In order to create realistic measurement data, we use the non-linear forward model given by the Beer-Lamberts law

$$\mathcal{A}(f;\ell) = e^{-\mu \int_\ell f(x)\,\mathrm{d}x},$$

where we select the mass attenuation coefficient $\mu = 0.2$ cm^2/g, which corresponds approximately to the value of water. We simulate low dose scans by adding Poisson noise to the measurement data. For computations we linearise the obtained data by applying $-\log(\cdot)/\mu$ to the measurements, by which the forward model simplifies to the ray-transform as in eq. (7). A slice from the test patient with the corresponding measurement data is shown in fig. 5.

We remind that we chose the number of iterations as $N + 1 = 5$ and hence the image resolution in the coarsest discretisation space S_0 is just 32×32. For the experiments we consider three scenarios of increasing difficulty, which means decreasing angles and photon counts. The first case corresponds roughly to a clinical low-dose CT scan with 600 angles and a photon count of 8000 that could be used for diagnostic purposes. The other two cases are sparse view scenarios, with which we aim to illustrate the limitations of the proposed algorithms. The selected parameters for each case are summarised in table I. We note, that the downsampling of angles can lead to very sparse angular sampling in the first iterates (coarse scales), which can render the initial reconstructions meaningless and hence we limited the number of angles in the sub-sampling to a minimum of 30. That means there is no downsampling in the projections for case 2 and 3 in the coarse scales.

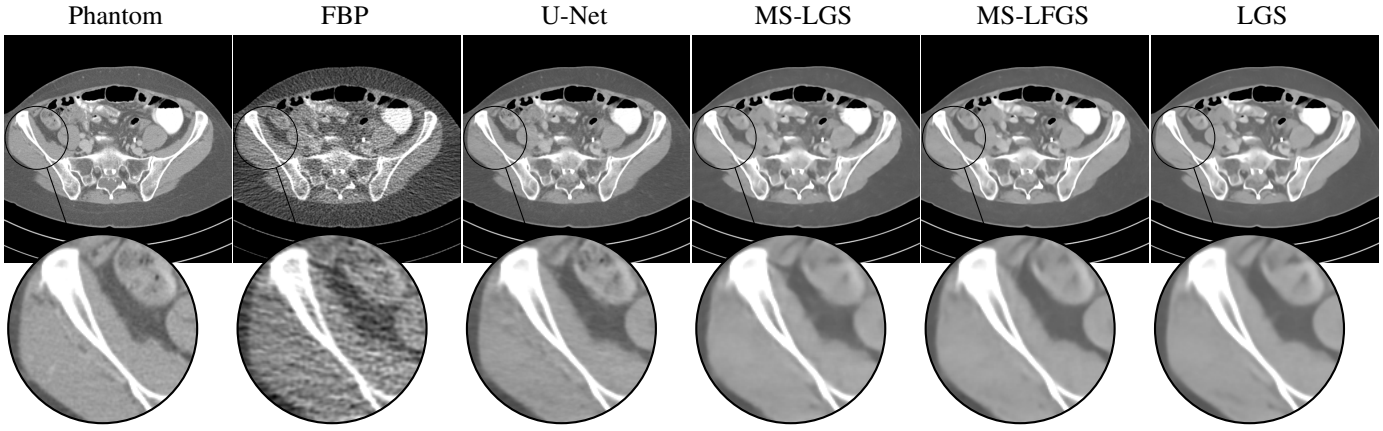

Fig. 6: Reconstructions of the test patient for measurement case 1 with 600 angles. All images are windowed and displayed on $[-300, 300]$HU. The filtered backprojection here is computed with $h = 0.4$.

TABLE I: Summary of parameter selection for the study on human phantoms.

	Angles	Photon count	Scales	Angles per scale
Case 1	600	8000	5	600, 300, 150, 75, 37
Case 2	240	6000	5	240, 120, 60, 30, 30
Case 3	120	5000	5	120, 60, 30, 30, 30

1) Training procedure for low dose scans: We compute reconstructions for the same 4 algorithms as in the ellipsoidal case, where the data is produced for all algorithms as outlined in the previous section and summarised in table I. We compute an initial reconstruction by filtered backprojection with the Hann filter and frequency scaling of $h = 0.6$, this reconstruction is then chosen as the input to the post-processing with U-Net The same parameters are selected to compute the filtered gradient eq. (5) for the MS-LFGS.

To make the comparison uniform for all test cases we optimised all algorithms in the same manner. In particular we chose Adam as the optimiser with an ℓ^2-loss, each network is trained for 50,000 iterations with one training sample per minimisation step. The initial learning rate is set to 10^{-3} with a cosine decay These choices have shown to perform well for all presented algorithms.

The resulting reconstructions for case 1 with 600 angles are shown in fig. 6. We will discuss the reconstruction results for all cases and present a quantitative evaluation in the next section.

V. Discussion

The presented framework for multi-scale learned iterative reconstructions in section III presents a general framework for a scalable iterative learned image reconstruction. For this study we have chosen a simple and straight-forward implementation of the scaling operations, but other choices are possible to extend the framework. In the following we will first evaluate the performance of the presented algorithms and continue to discuss possible extensions.

A. Evaluation of reconstruction quality

Obtained reconstructions for all algorithms under consideration are shown in fig. 6 for case 1, fig. 7 for case 2, and fig. 8 for case 3. Let us first note that U-Net does generally produce sharper images than the learned approaches, but also tends to include features in the reconstruction that are not present in the high-dose scan. The learned approaches tend to produce smoother reconstructions, in particular we observe that in areas of uncertainty the learned approaches are more conservative in recreating features and tend to uniform areas instead of reproducing features from the training data. For case 2 and 3 with sparse angles, we can observe that the simple MS-LGS does perform as well as in the high angle case 1. The additional filtered gradient in the MS-LFGS does generally improve reconstruction quality considerably and improves uniformity of reconstructed regions. At this point we would like to note, that case 2 and 3 with sparse angular sampling are typically not used for diagnostic purposes, but rather for dose calculations and monitoring purposes.

We have computed quantitative measures for all cases as shown in table II. Quantitatively, LGS does outperform all other methods, but we note that this version operating only on the full resolution is not scalable, as well as the U-Net approach. We also note that LGS is expected to perform best in this comparison, since it does operate on the full resolution in each iteration. Of the two scalable algorithms, MS-LFGS does clearly perform better and consistently outperforms post-processing with U-Net and is close to LGS in its performance. On case 1 with dense angular sampling, MS-LGS does outperform U-Net as well, but on the harder cases loses accuracy and does perform slightly worse than U-Net.

Regarding memory consumption, the multi-scale approaches are expected to be cheaper, both using only 980MB for training. Compared to 3784MB for U-Net and 3042MB for LGS. With respect to computational cost, as it has been discussed in section III-C, we expect both multi-scale approaches to be cheaper than LGS. In fact, as can be seen in table III, MS-LGS is consistently faster in execution times than filtered backprojection and U-Net. Consequently, MS-LFGS is slightly slower than the U-Net post-processing, but still considerably

TABLE II: Quantitative measures for low dose scans. Averaged over 210 slices of test patient. Mean values are shown with their standard deviation.

	600 angles		240 angles		120 angles	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
FBP	32.80 ±1.51	0.781 ±0.0515	31.27 ±1.46	0.732 ±0.0466	29.61 ±1.43	0.636 ±0.0334
LGS	**43.57 ±1.22**	**0.964 ±0.0033**	**41.36 ±1.22**	**0.953 ±0.0039**	**39.45 ±1.23**	**0.938 ±0.0054**
U-Net	42.24 ±1.50	0.957 ±0.0044	40.41 ±1.38	0.944 ±0.0031	38.03 ±1.32	0.901 ±0.0053
MS-LGS	42.55 ±1.26	0.958 ±0.0033	39.73 ±1.33	0.938 ±0.0044	37.16 ±1.39	0.913 ±0.0065
MS-LFGS	**43.34 ±1.22**	**0.963 ±0.0034**	**40.80 ±1.24**	**0.947 ±0.0040**	**38.30 ±1.26**	**0.926 ±0.0061**

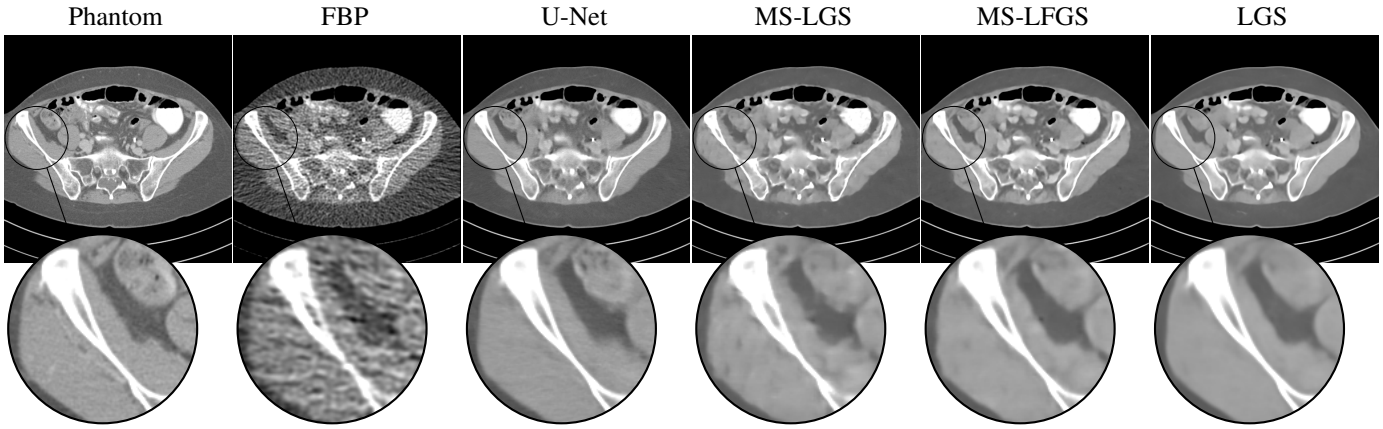

Fig. 7: Reconstructions of the test patient for measurement case 2 with 240 angles. All images are windowed and displayed on $[-300, 300]$HU. The filtered backprojection here is computed with $h = 0.3$.

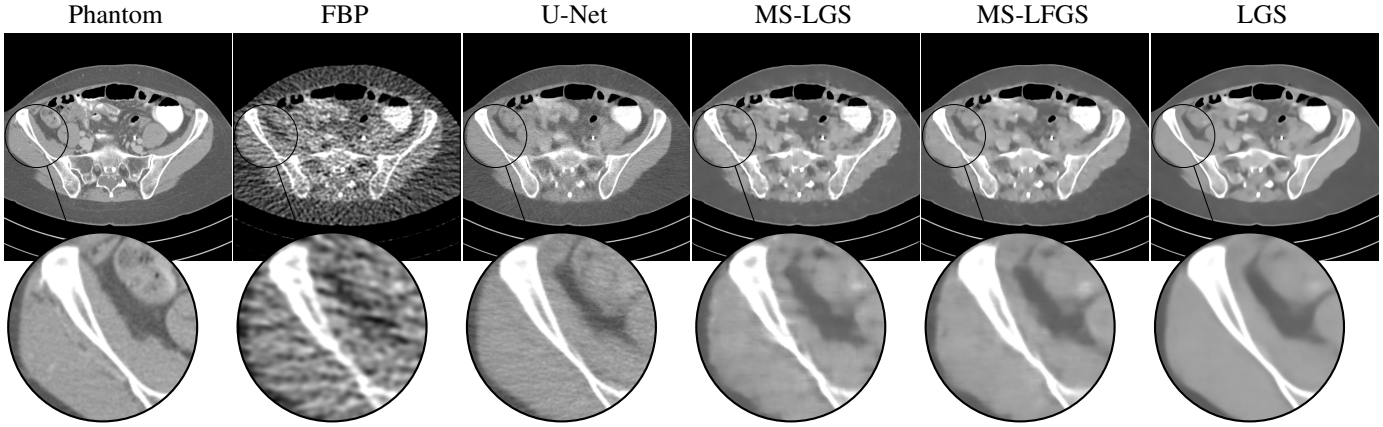

Fig. 8: Reconstructions of the test patient for measurement case 3 with 120 angles. All images are windowed and displayed on $[-300, 300]$HU. The filtered backprojection here is computed with $h = 0.25$.

TABLE III: Training and execution times of the learned algorithms under consideration.

	600 angles		240 angles		120 angles	
	TRAIN	EXEC.	TRAIN	EXEC.	TRAIN	EXEC.
LGS	9h50m	315ms	7h00m	206ms	5h50m	158ms
U-NET	5h30m	68ms	4h15m	57ms	3h50m	52ms
MS-LGS	4h25m	65ms	3h15m	55ms	2h30m	49ms
MS-LFGS	6h55m	129ms	4h40m	104ms	4h00m	96ms

faster than LGS. The same tendencies are seen in the training times for each algorithm.

B. Influence of scales

The computational advantage of the multiscale approach is primarily due to the low-cost computations on the coarse resolution, but these come with some subtleties. As we have experienced that the early iterates on low resolutions are prone to overfitting and can negatively influence the reconstruction quality on the following iterates. We have experienced that the lowest resolution should not be lower than 32^2, the same

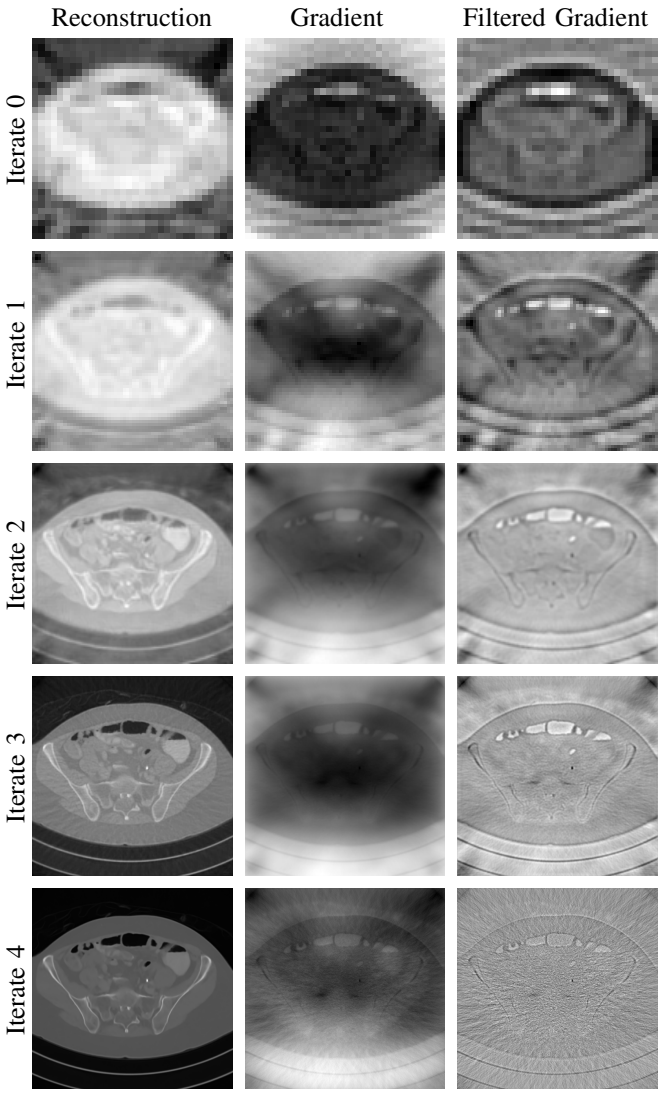

Fig. 9: Representation of the multi-scale schemes with obtained reconstruction, gradient, and filtered gradient on each discretisation space.

goes for a minimum amount of angles, set to 30, as we have discussed earlier. This restriction enforces the suitability of the proposed algorithm for large-scale problems, where this will not be a major issue. The obtained reconstructions on the increasing scales and their corresponding (filtered) gradients are shown in fig. 9.

C. Extensions of the multi-scale approach

In this study we have chosen the structure of the multi-scale algorithms as simplistic as possible. Nevertheless, the proposed framework does offer larger flexibility in choices that might be more suitable for other applications. In particular with respect to network design and choice of discretisation spaces. In the following we would like to mention some possibilities how the multi-scale schemes can be extended:

- In our study the mini U-Net has shown to be effective to restore high-frequency components more effectively than a basic feed forward CNN as utilised in [12]. We note that more memory efficient networks might be used, such as the MS-D Net [38]. Futhermore, since the early iterates on the coarse spaces do not need such an expressive network, it could be advised to chose a progressively growing network, such that only the last iterate utilises the most expressive network.
- We have chosen to identify each discretisation space with one iteration. This limitation can be easily relaxed, for instance by computing two iterations in the same discretisation space. In case all iterates are computed on the same space, this simplifies to the basic LGS.
- We have chosen to reduce the resolution in all dimensions equally. It would be also possible to only reduce the resolution along one dimension in each step and alternate in dimensions, similarly the amount of angles can be adjusted as we have done for case 2 and 3. Along the same lines, the upsampling operator can be chosen differently, including the possibility of a learned upsampling.
- Lastly, the multi-scale framework is not limited to learned gradient schemes and can be extended to other learned approaches such as variational networks [17] and learned primal-dual [30].

VI. Conclusions

We have presented a general framework for scalable learned iterative reconstruction algorithms for large-scale problems and higher dimensions, by restricting the expensive computation of the forward operator to only one application in the final reconstruction space. We presented two methods, a multi-scale learned gradient scheme based on the previous work [12] and a multi-scale learned filtered gradient scheme, that additionally computes the gradient by filtered backprojection.

The presented algorithms are evaluated on human abdominal CT scans for three different imaging scenarios of dense and sparse angular sampling. Both algorithms perform competetive to previously introduced methods, while being scalable and faster in execution time. It is especially notable, that MS-LGS does present a learned iterative reconstruction method that is faster than filtered backprojection with U-Net. Whereas this work is primarily a feasibility study, we believe that it will be of high relevance to applications where high dimensionality of the imaging problem is inherent, such as in cone-beam CT.

Acknowledgment

We acknowledge the support of NVIDIA Corporation with one Titan Xp GPU. This work was partially supported by the Academy of Finland (Project 312123, Finnish Centre of Excellence in Inverse Modelling and Imaging, 2018–2025) and by British Heart Foundation grant NH/18/1/33511, EPSRC grant EP/M020533/1, and EPSRC-Wellcome grant WT101957. The authors would also like to acknowledge the Swedish Foundation for Strategic Research grants *Low complexity reconstruction for medicine* (AM13-0049) and *3D reconstruction with simulated image formation models* (ID14-0055). Funding has also been provided by Elekta (Stockholm, Sweden).

REFERENCES

[1] W. Stiller. Basics of iterative reconstruction methods in computed tomography: A vendor-independent overview. *European Journal of Radiology*, 109:147–154, 2018.

[2] Lucas L. Geyer, U. Joseph Schoepf, Felix G. Meinel, John W. Nance, Gorka Bastarrika, Jonathon A. Leipsic, Narinder S. Paul, Marco Rengo, Andrea Laghi, and Carlo N. De Cecco. State of the art: Iterative CT reconstruction techniques. *Radiology*, 276(2):339–357, 2015.

[3] Emil Y Sidky, Jakob H Jørgensen, and Xiaochuan Pan. Convex optimization problem prototyping for image reconstruction in computed tomography with the Chambolle–Pock algorithm. *Physics in Medicine & Biology*, 57(10):3065, 2012.

[4] J. Kaipio and E. Somersalo. *Statistical and Computational Inverse Problems*, volume 160 of *Applied Mathematical Sciences*. Springer Verlag, 2004.

[5] Samuli Siltanen, Ville Kolehmainen, Seppo Järvenpää, JP Kaipio, P Koistinen, M Lassas, J Pirttilä, and E Somersalo. Statistical inversion for medical X-ray tomography with few radiographs: I. General theory. *Physics in Medicine & Biology*, 48(10):1437, 2003.

[6] Ville Kolehmainen, Samuli Siltanen, Seppo Järvenpää, Jari P Kaipio, P Koistinen, M Lassas, J Pirttilä, and E Somersalo. Statistical inversion for medical X-ray tomography with few radiographs: II. Application to dental radiology. *Physics in Medicine & Biology*, 48(10):1465, 2003.

[7] L.I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, 1992.

[8] Kristian Bredies, Karl Kunisch, and Thomas Pock. Total generalized variation. *SIAM Journal on Imaging Sciences*, 3(3):492–526, 2010.

[9] Xiaoqun Zhang, Martin Burger, Xavier Bresson, and Stanley Osher. Bregmanized nonlocal regularization for deconvolution and sparse reconstruction. *SIAM Journal on Imaging Sciences*, 3(3):253–276, 2010.

[10] Maaria Rantala, Simopekka Vanska, Seppo Jarvenpaa, Martti Kalke, Matti Lassas, Jan Moberg, and Samuli Siltanen. Wavelet-based reconstruction for limited-angle x-ray tomography. *IEEE transactions on medical imaging*, 25(2):210–217, 2006.

[11] Tatiana A Bubba, Federica Porta, Gaetano Zanghirati, and Silvia Bonettini. A nonsmooth regularization approach based on shearlets for poisson noise removal in ROI tomography. *Applied Mathematics and Computation*, 318:131–152, 2018.

[12] Jonas Adler and Ozan Öktem. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 33(12):124007, 2017.

[13] Bo Zhu, Jeremiah Z Liu, Stephen F Cauley, Bruce R Rosen, and Matthew S Rosen. Image reconstruction by domain-transform manifold learning. *Nature*, 555(7697):487, 2018.

[14] Eunhee Kang, Junhong Min, and Jong Chul Ye. A deep convolutional neural network using directional wavelets for low-dose X-ray CT reconstruction. *Medical Physics*, 44(10), 2017.

[15] Kyong Hwan Jin, Michael T McCann, Emmanuel Froustey, and Michael Unser. Deep convolutional neural network for inverse problems in imaging. *IEEE Transactions on Image Processing*, 26(9):4509–4522, 2017.

[16] Jong Chul Ye, Yoseob Han, and Eunju Cha. Deep convolutional framelets: A general deep learning framework for inverse problems. *SIAM Journal on Imaging Sciences*, 11(2):991–1048, 2018.

[17] Kerstin Hammernik, Teresa Klatzer, Erich Kobler, Michael P Recht, Daniel K Sodickson, Thomas Pock, and Florian Knoll. Learning a variational network for reconstruction of accelerated MRI data. *Magnetic resonance in medicine*, 79(6):3055–3071, 2018.

[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 3D photoacoustic tomography. *IEEE Transactions on Medical Imaging*, 2018.

[19] Jo Schlemper, Jose Caballero, Joseph V Hajnal, Anthony Price, and Daniel Rueckert. A deep cascade of convolutional neural networks for MR image reconstruction. In *International Conference on Information Processing in Medical Imaging*, pages 647–658. Springer, 2017.

[20] Housen Li, Johannes Schwab, Stephan Antholzer, and Markus Haltmeier. Nett: Solving inverse problems with deep neural networks. *arXiv preprint arXiv:1803.00092*, 2018.

[21] Yoeri E Boink, Christoph Brune, and Srirang Manohar. Robustness of a partially learned photoacoustic reconstruction algorithm. In *Photons Plus Ultrasound: Imaging and Sensing 2019*, volume 10878, page 108781D. International Society for Optics and Photonics, 2019.

[22] Andreas Hauptmann, Ben Cox, Felix Lucka, Nam Huynh, Marta Betcke, Paul Beard, and Simon Arridge. Approximate k-space models and deep learning for fast photoacoustic reconstruction. In *International Workshop on Machine Learning for Medical Image Reconstruction*, pages 103–111. Springer, 2018.

[23] Frank Natterer. *The mathematics of computerized tomography*. SIAM, 2001.

[24] M. Lassas, E. Saksman, and S. Siltanen. Discretization-invariant Bayesian inversion and Besov space priors. *Inverse Problems and Imaging*, 3:87–122, 2009.

[25] Zenith Purisha, Juho Rimpeläinen, Tatiana Bubba, and Samuli Siltanen. Controlled wavelet domain sparsity for x-ray tomography. *Measurement Science and Technology*, 29(1):014002, 2017.

[26] Johannes Schwab, Stephan Antholzer, and Markus Haltmeier. Deep null space learning for inverse problems: convergence analysis and rates. *Inverse Problems*, 2018.

[27] Andreas Hauptmann, Simon Arridge, Felix Lucka, Vivek Muthurangu, and Jennifer A Steeden. Real-time cardiovascular MR with spatio-temporal artifact suppression using deep learning–proof of concept in congenital heart disease. *Magnetic resonance in medicine*, 81(2):1143–1156, 2019.

[28] Sarah Jane Hamilton and Andreas Hauptmann. Deep D-bar: Real-time electrical impedance tomography imaging with deep neural networks. *IEEE transactions on medical imaging*, 37(10):2367–2377, 2018.

[29] Sarah J Hamilton, Asko Hänninen, Andreas Hauptmann, and Ville Kolehmainen. Beltrami-net: domain independent deep D-bar learning for absolute imaging with electrical impedance tomography (a-EIT). *Physiological measurement*, 2019.

[30] Jonas Adler and Ozan Öktem. Learned primal-dual reconstruction. *IEEE transactions on medical imaging*, 37(6):1322–1332, 2018.

[31] Patrick Putzky and Max Welling. Recurrent inference machines for solving inverse problems. *arXiv preprint arXiv:1706.04008*, 2017.

[32] Hao Gao. Fused analytical and iterative reconstruction (AIR) via modified proximal forward–backward splitting: a FDK-based iterative image reconstruction example for CBCT. *Physics in Medicine & Biology*, 61(19):7187, 2016.

[33] 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.

[34] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[35] J Adler, H Kohr, and O Öktem. Operator discretization library (ODL). *Software available from https://github. com/odlgroup/odl*, 2017.

[36] Wim van Aarle, Willem Jan Palenstijn, Jeroen Cant, Eline Janssens, Folkert Bleichrodt, Andrei Dabravolski, Jan De Beenhouwer, K Joost Batenburg, and Jan Sijbers. Fast and flexible x-ray tomography using the ASTRA toolbox. *Optics Express*, 24(22):25129–25147, 2016.

[37] C McCollough. Tu-fg-207a-04: Overview of the low dose CT grand challenge. *Medical physics*, 43(6Part35):3759–3760, 2016.

[38] Daniël M Pelt and James A Sethian. A mixed-scale dense convolutional neural network for image analysis. *Proceedings of the National Academy of Sciences*, 115(2):254–259, 2018.