Computed tomography reconstruction using deep image prior and learned reconstruction methods

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:

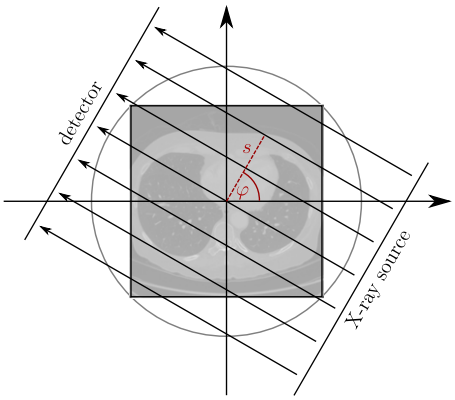

Figure 1. Parallel beam geometry.

$$Ax(s,\varphi) = \int_{\mathbb{R}} x\left(L_{s,\varphi}(t)\right)\ \mathrm{d}t. \tag{2}$$

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,

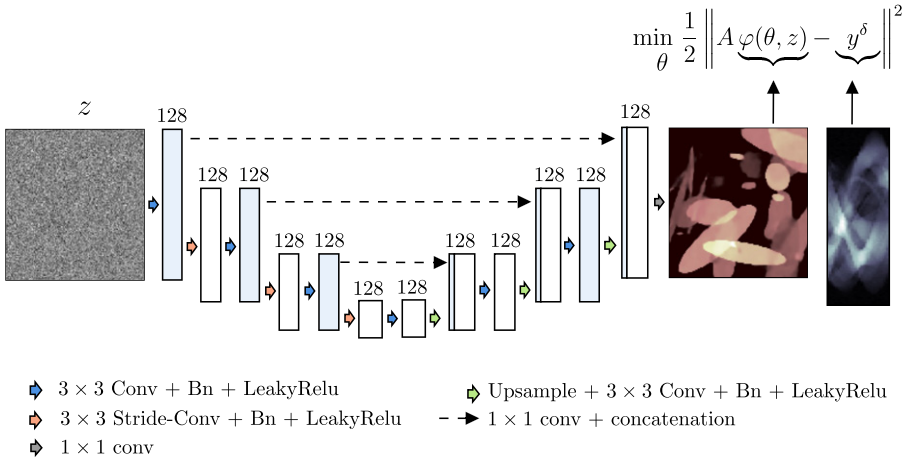

Figure 2. The figure illustrates the DIP approach. We use a U-Net architecture with 128 channels at every layer. Some layers have additionally the skip channels (coming from the dashed arrows). We always use either 4 or 0 skip channels.

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.

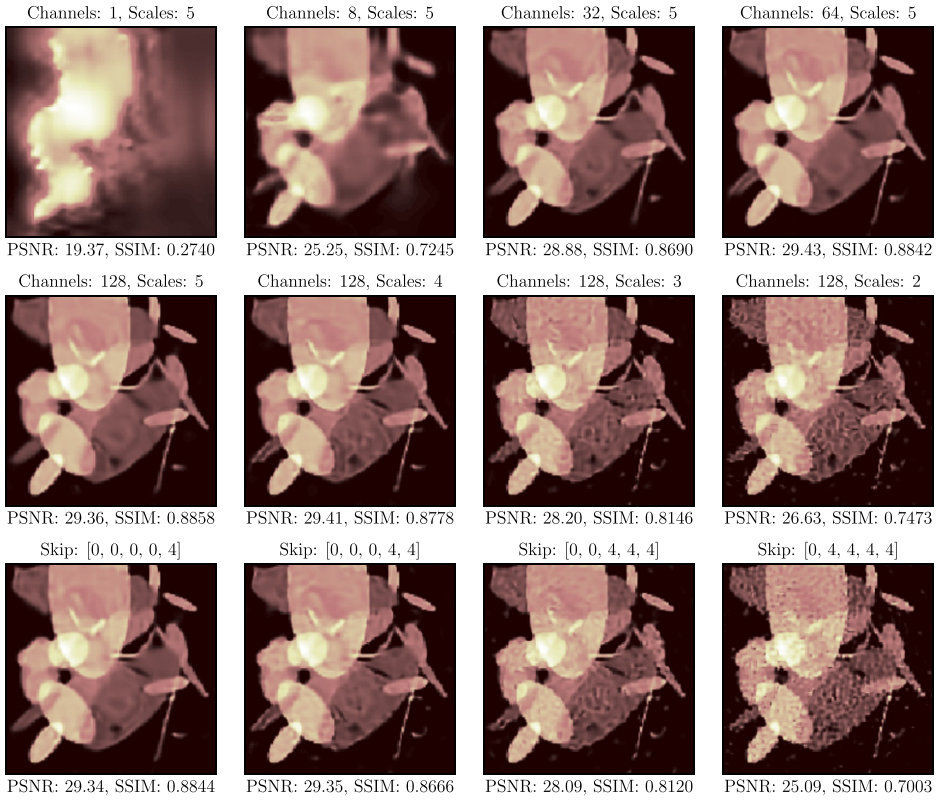

Figure 4. CT reconstructions after 5000 iterations using the DIP with a U-Net architecture and different scales (depths), channels per layer (the network has the same number of channels at every layer) and number of skip connections (the first two rows do not use skip connections, i.e., skip: $[0, 0, 0, 0, 0]$). In the last row all reconstructions use 5 scales and 128 channels.

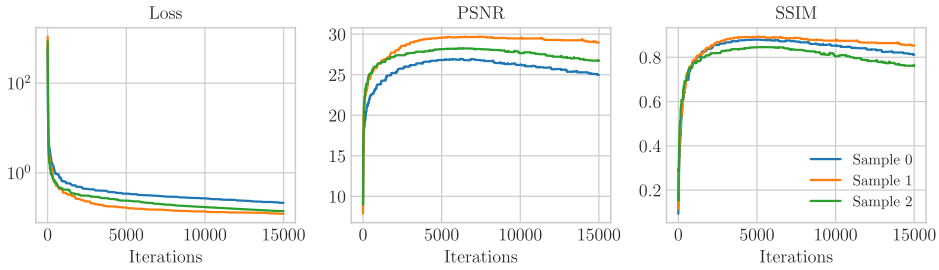

Figure 5. Training loss and true similarity (PSNR and SSIM) of CT reconstructions using the DIP approach. The training was done over 15 000 iterations and the architecture is an encoder–decoder (no skip channels) with 5 scales and 128 channels per layer.

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

$$\theta_x \in \arg\min_{\theta\in\Theta} \|\varphi(\theta,\, z) - x\|^2. \tag{15}$$

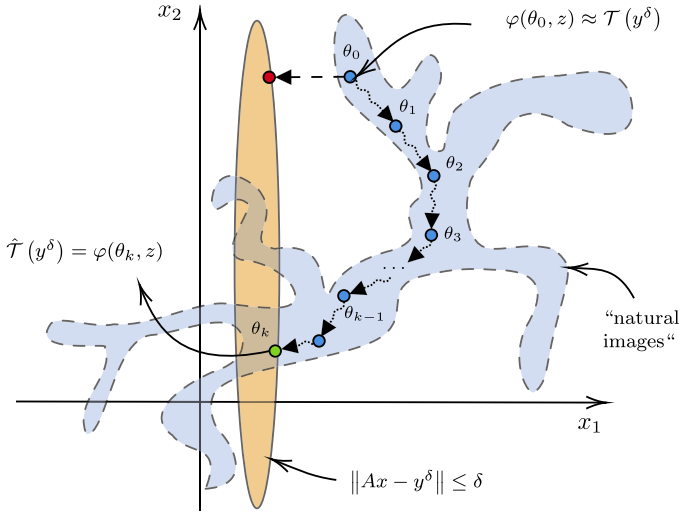

Figure 6. Graphical illustration of the DIP approach with initial reconstruction. The blue area refers to an approximation of some part of the space of *natural* images.

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

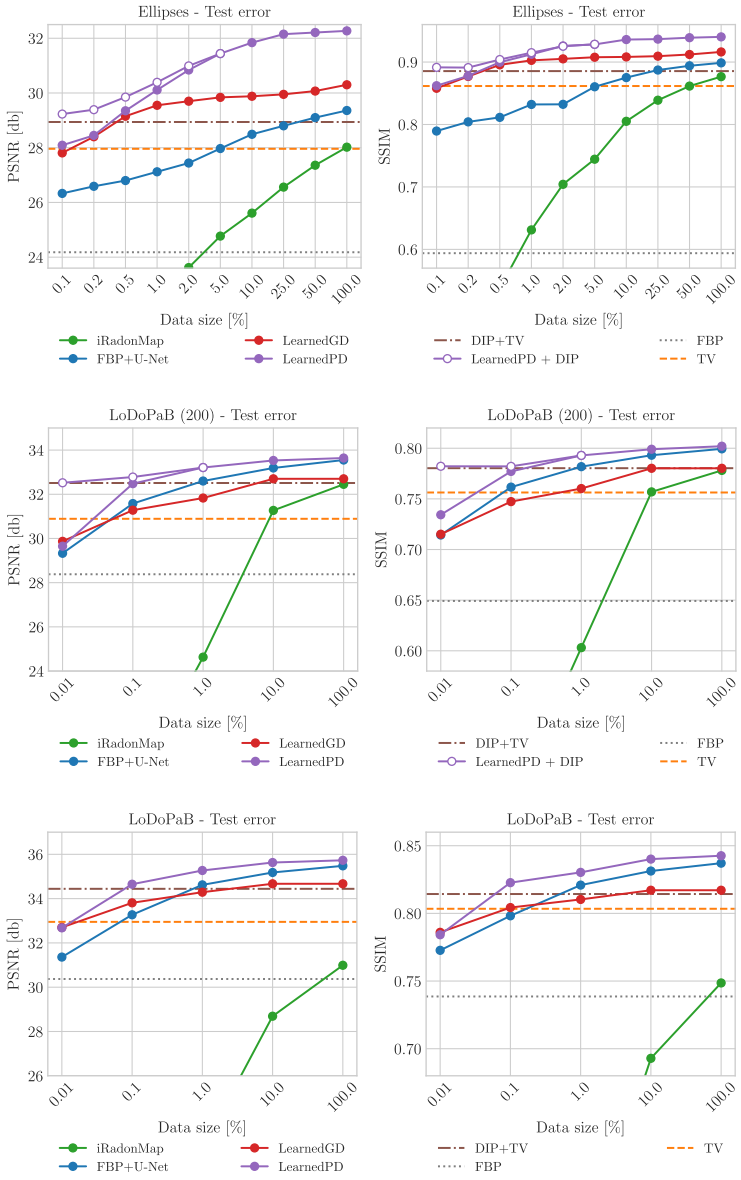

Figure 7. Benchmark results of several existing methods and the proposed approaches (DIP + TV, learned primal-dual + DIP) on the Ellipses, LoDoPaB (200) and LoDoPaB datasets. The horizontal lines indicate the performance of data-free methods.

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.

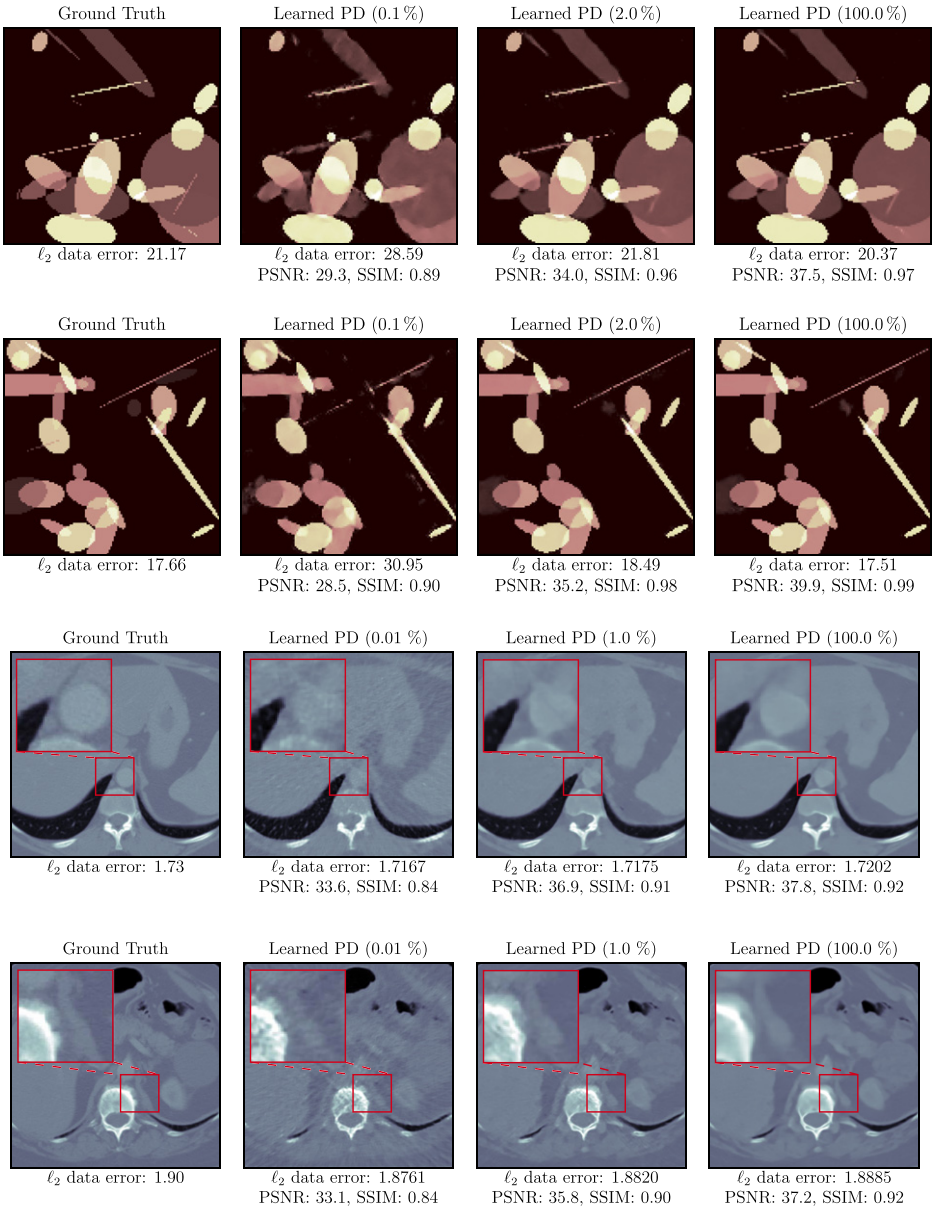

Figure 8. Reconstructions of test samples using the learned primal-dual method trained with different amounts of data from the ellipses and LoDoPaB datasets. The ℓ_2 data error measures the discrepancy between the noisy observation and the noise-free projection of the (reconstructed) image.

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

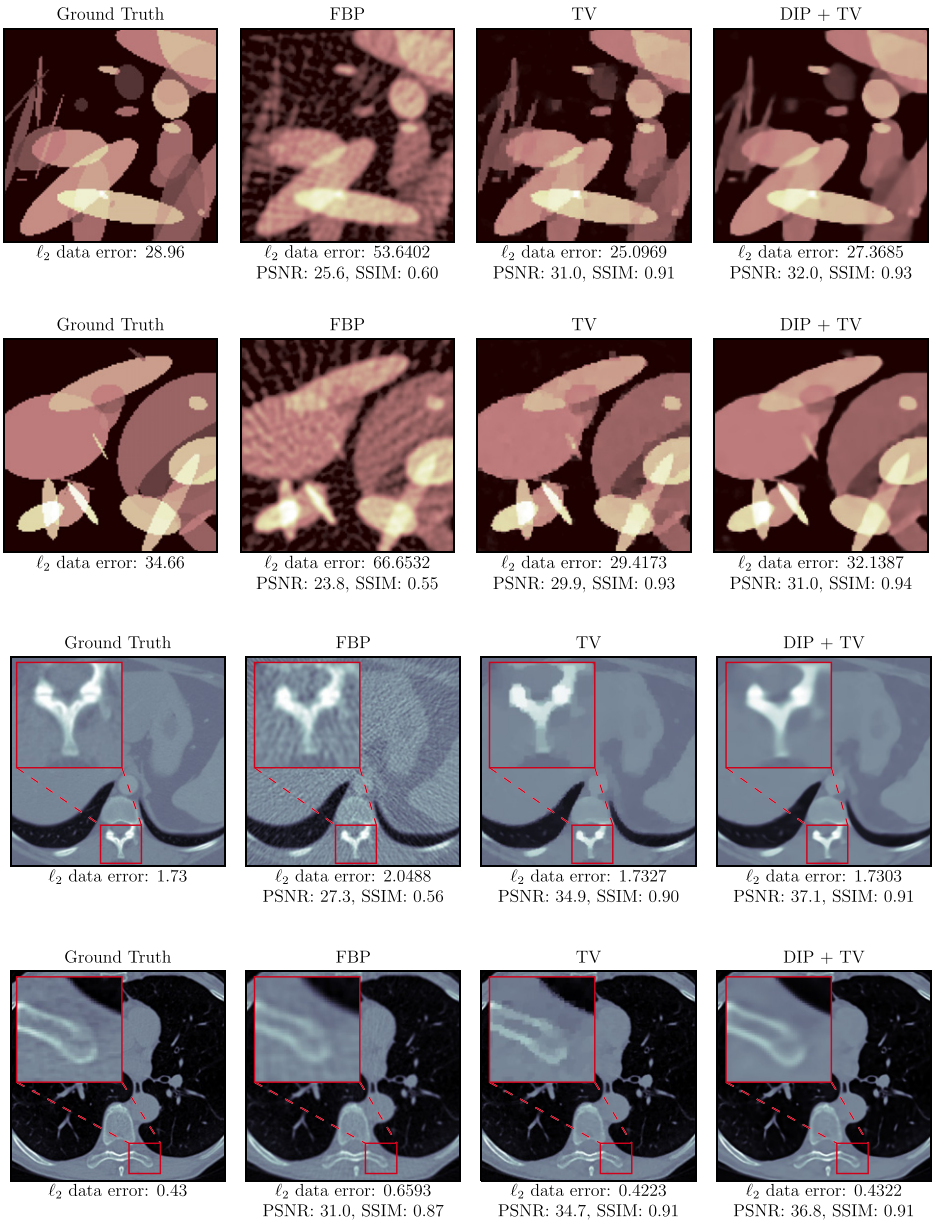

Figure 9. Reconstruction obtained with the FBP method, isotropic TV regularization and the DIP approach combined with TV, for test samples from the ellipses and LoDoPaB datasets. The ℓ_2 data error measures the discrepancy between the noisy observation and the noise-free projection of the (reconstructed) 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.

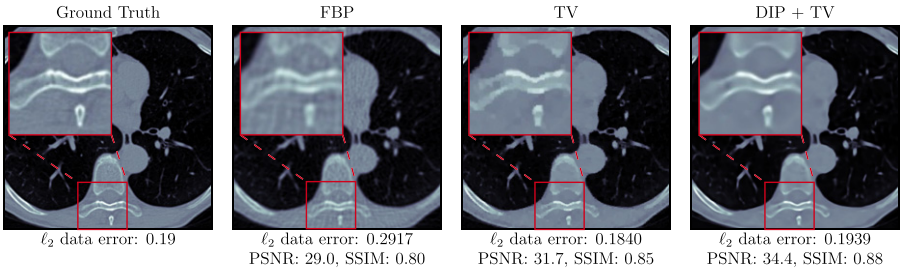

Figure 10. Reconstruction obtained with the FBP method, isotropic TV regularization and the DIP approach combined with TV, for test samples from the LoDoPaB (200) dataset. The ℓ_2 data error measures the discrepancy between the noisy observation and the noise-free projection of the (reconstructed) image.

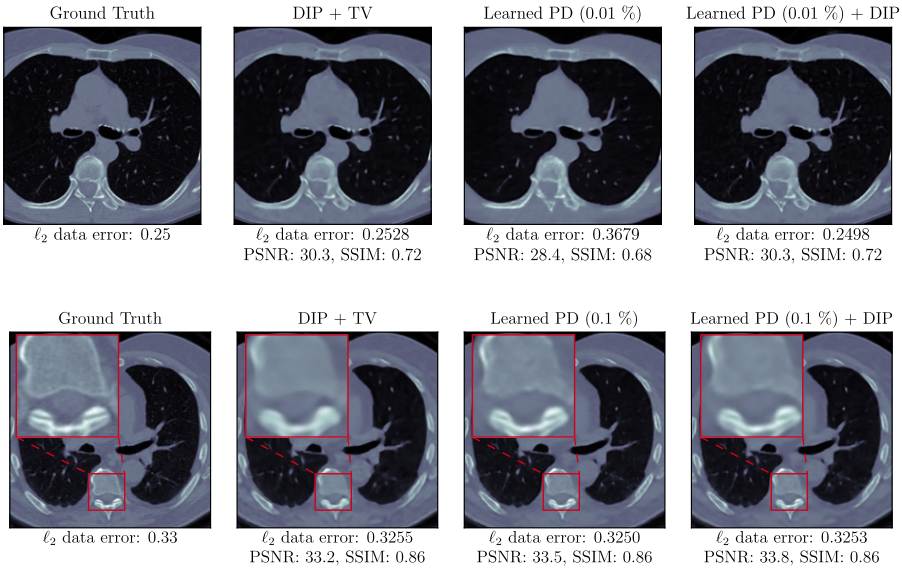

Figure 11. Examples of reconstructions obtained with the DIP + TV approach, the learned primal-dual method trained with 0.01% and 0.1% of the LoDoPaB (200) dataset and the DIP + TV approach with initial reconstruction. The ℓ_2 data error measures the discrepancy between the noisy observation and the 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