PatchNR: learning from very few images by patch normalizing flow regularization

OPEN ACCESS
IOP Publishing
Inverse Problems

Inverse Problems **39** (2023) 064006 (27pp) https://doi.org/10.1088/1361-6420/acce5e

PatchNR: learning from very few images by patch normalizing flow regularization

Fabian Altekrüger[2,1,*], **Alexander Denker**[3], **Paul Hagemann**[1], **Johannes Hertrich**[1], **Peter Maass**[3] and **Gabriele Steidl**[1]

[1] Institute of Mathematics, Technische Universität Berlin, Straße des 17. Juni 136, D-10623 Berlin, Germany
[2] Department of Mathematics, Humboldt-Universität zu Berlin, Unter den Linden 6, D-10099 Berlin, Germany
[3] Center for Industrial Mathematics, University of Bremen, Bibliothekstraße 5, D-28359 Bremen, Germany

E-mail: fabian.altekrueger@hu-berlin.de

Received 22 November 2022; revised 27 February 2023
Accepted for publication 19 April 2023
Published 16 May 2023

Abstract

Learning neural networks using only few available information is an important ongoing research topic with tremendous potential for applications. In this paper, we introduce a powerful regularizer for the variational modeling of inverse problems in imaging. Our regularizer, called patch normalizing flow regularizer (patchNR), involves a normalizing flow learned on small patches of very few images. In particular, the training is independent of the considered inverse problem such that the same regularizer can be applied for different forward operators acting on the same class of images. By investigating the distribution of patches versus those of the whole image class, we prove that our model is indeed a maximum a posteriori approach. Numerical examples for low-dose and limited-angle computed tomography (CT) as well as superresolution of material images demonstrate that our method provides very high quality results. The training set consists of just six images for CT and one image for superresolution. Finally, we combine our patchNR with ideas from internal learning for performing superresolution of natural images directly from the low-resolution observation without knowledge of any high-resolution image.

* Author to whom any correspondence should be addressed.

Keywords: patch priors, normalizing flows, computed tomography, superresolution, zero-shot learning

(Some figures may appear in colour only in the online journal)

1. Introduction

Solving inverse problems with limited access to training data is an active field of research. In inverse problems, we aim to reconstruct an unknown ground truth image x from some observation

$$y = \text{noisy}(f(x)), \tag{1}$$

where f is an ill-posed forward operator and 'noisy' describes some noise model. In the recent years learning-based reconstruction methods as supervisely trained neural networks on large datasets [37, 68] and conditional generative models [6, 28, 57, 75] attained a lot of attention. However, for many applications like medical or material imaging, the acquisition of a large database of ground truth images or pairs of ground truth images and observations is very costly or even impossible [48, 89]. Model-based approaches assume that the forward operator f is known and aim to minimize a variational functional of the form

$$\mathcal{J}(x;y) = \mathcal{D}(f(x),y) + \lambda\mathcal{R}(x),\ \lambda > 0, \tag{2}$$

where $\mathcal{D}$ is a data-fidelity term which depends on the noise model and measures how well the reconstruction fits to the observation and $\mathcal{R}$ is a regularizer which copes with the ill-posedness and incorporates prior information. Over the last years, learned regularizers like the total deep variation [46, 47] or adversarial regularizers [55, 58, 63] as well as extensions of plug-and-play and unrolled methods [24, 78, 84, 88] with learned denoisers [32, 35, 67, 91] showed promising results, see [8, 59] for an overview.

Furthermore, many papers leveraged the tractability of the likelihood of normalizing flows (NFs) to learn a prior [9, 30, 85, 86, 90] or use conditional variants to learn the posterior [12, 53, 79, 87] They utilize the invertibility to optimize over the range of the flow together with the Gaussian assumption on the latent space. Also, diffusion models [39, 40, 76, 77] have shown great generative modelling capabilities and have been used as a prior for inverse problems. Moreover, other generative models, such as GANs [4, 26, 60] or VAEs [44], have been used as a regularizer, see the recent review [20] and references therein. However, even if these methods allow an unsupervised reconstruction, their training is often computationally costly and a huge amount of training images is required.

One possibility to reduce the training effort consists in using only small image patches. Denoising methods based on the comparison of similar patches provided state-of-the-art methods [13, 16, 49, 50] for a long time. Recently, the approximation of patch distributions of images was successfully exploited in certain papers [3, 18, 27, 31, 33, 34, 71]. In particular, [93] proposed the negative log likelihood of all patches of an image as a regularizer, where the underlying patch distribution was assumed to follow a Gaussian mixture model (GMM) which parameters were learned from few clean images. This method is still competitive to many approaches based on deep learning and several extensions were suggested recently [22, 61, 72]. However, even though GMMs can approximate any probability density function if the number of components is large enough, they suffer from limited flexibility in case of a fixed number of components, see [23] and the references therein. Moreover the subsequent reconstruction procedure detailed in [93] is computationally expensive.

In this paper, we propose to use a regularizer which incorporates a NF learned on small image patches, usually of size 6×6. NFs were introduced in [19, 66], see also [70] for its continuous counterpart. They build upon invertible neural networks and allow for explicit density evaluation. Our PatchNR consists of a NF which is trained to approximate the distribution of patches. As the structure of small patches is usually much simpler than those of whole images, it appears that their approximation is more accurate. Moreover, as a large database of patches can be extracted from few images, we require only a very small amount of training images. Once the patchNR is learned, we use the negative log likelihood of all patches as regularizer in (2) for its reconstruction. Indeed we will prove that our model can be obtained by a maximum a posteriori (MAP) approach. Since the regularizer is only specific to the considered image class, but not to the inverse problem, it can be applied for different forward operators as, e.g. for low-dose computed tomography (CT) and limited-angle CT without additional training. This is in contrast to data-based methods as filtered backprojection (FBP)+UNet, where the network has to be trained for each new forward operator separately.

We demonstrate by numerical examples that our patchNR admits high quality results for low-dose and limited-angle CT as well as superresolution. Moreover, we combine patchNRs with ideas from internal learning to perform superresolution on natural images without any training data. Internal learning [25, 73] is based on the observation that the patches within natural images are self-similar across the scales. In our case, this leads to the idea to train the patchNR on the low-resolution observation instead of a dataset of training images. We demonstrate on the BSD68 dataset that zero-shot superresolution with patchNRs outperforms comparable methods including the original zero shot super-resolution (ZSSR) paper by [73]. Finally, let us mention that recent works have explored meta-learning for efficient one gradient step reconstruction [74], applications to light field superresolution [15] and CT superresolution [92].

The paper is organized as follows: We start by explaining the training of our patchNR and the variational reconstruction model in section 2. Our approach is integrated into the MAP framework in section 3, i.e. the patchNR defines a probability density on the full image space. In section 4, we evaluate the performance of our model in CT and superresolution and use the patchNR also for zero-shot superresolution. Finally, conclusions are drawn in section 5.

2. Patch NFs in variational modeling

In the following, we assume that we are given a small number of high quality example images $x_1, \dots, x_M \in \mathbb{R}^{d_1 \times d_2}$—indeed $M = 1$ or $M = 6$ in our numerical examples - from a certain image class as CT, material or texture images. Our method consists of two steps, namely i) learning a NF for approximating the patch distribution of the example images, and ii) establishing a variational model which incorporates the learned patchNR in the regularizer.

2.1. Learning patchNRs

Let $\mathrm{p}_1, \dots, \mathrm{p}_N \in \mathbb{R}^{s_1 \times s_2}$, $s_i \ll d_i$, $i = 1, 2$ denote all possibly overlapping patches of the example images, where we assume that the patches $\mathrm{p}_1, \dots, \mathrm{p}_N$ are realizations of an absolute continuous probability distribution Q with density q. We aim to approximate q by a NF. For simplicity, we rearrange the images and the patches columnwise into vectors of size $d = d_1 d_2$ and $s := s_1 s_2$, respectively. Then we learn a diffeomorphism $\mathcal{T} = \mathcal{T}_\theta \colon \mathbb{R}^s \to \mathbb{R}^s$ such that $Q \approx \mathcal{T}_\# P_Z := P_Z \circ \mathcal{T}^{-1}$, where P_Z is a s-dimensional standard normal distribution. To this end, we set our NF $\mathcal{T}$ to be an invertible neural network with parameters collected in θ. There

were several approaches in the literature to construct such invertible neural networks [5, 14, 19, 43, 54]. In this paper, we adapt the architecture from [5]. In order to train our NF, we aim to minimize the (forward) Kullback-Leibler divergence

$$\mathrm{KL}(Q, \mathcal{T}_{\#}P_Z) := \int_{\mathbb{R}^s} \log\left(\frac{\mathrm{d}Q}{\mathrm{d}\mathcal{T}_{\#}P_Z}\right) \mathrm{d}Q = \mathbb{E}_{\mathrm{p}\sim Q}\left[\log\left(q(\mathrm{p})\right)\right] - \mathbb{E}_{\mathrm{p}\sim Q}\left[\log\left(p_{\mathcal{T}_{\#}P_Z}(\mathrm{p})\right)\right],$$

where we set the expression to $+\infty$ if Q is not absolutely continuous with respect to $\mathcal{T}_{\#}P_Z$. The first term on the right-hand side is a constant independent of the parameters θ. Thus, using the change-of-variables formula for probability density functions of push-forward measures $p_{\mathcal{T}_{\#}P_Z} = p_Z(\mathcal{T}^{-1})|\det(\nabla\mathcal{T}^{-1})|$, we obtain that the above formula is up to a constant equal to

$$-\mathbb{E}_{\mathrm{p}\sim Q}\left[\log\left(p_Z(\mathcal{T}^{-1}(\mathrm{p})\right) + \log\left(\left|\det(\nabla\mathcal{T}^{-1}(\mathrm{p}))\right|\right)\right],$$

where $\nabla\mathcal{T}^{-1}$ denotes the Jacobian matrix of $\mathcal{T}^{-1}$. By replacing the expectation by the empirical mean of our training set, inserting the standard normal density p_Z and neglecting some constants, we finally obtain the loss function

$$\mathcal{L}(\theta) := \frac{1}{N}\sum_{i=1}^{N} \frac{1}{2}\left\|\mathcal{T}^{-1}(\mathrm{p}_i)\right\|^2 - \log\left(\left|\det(\nabla\mathcal{T}^{-1}(\mathrm{p}_i))\right|\right). \tag{3}$$

We minimize this loss function using the Adam optimizer [42].

2.2. Reconstruction with PatchNRs

Once the patchNR $\mathcal{T}$ is learned, we aim to use it within a regularizer of the variational model (2) to solve the inverse problem (1). To this end, denote by $E_i\colon \mathbb{R}^d \to \mathbb{R}^s$, $i = 1, \ldots, N_p$ the linear operator, which extracts the ith (vectorized) patch from the unknown (vectorized) image $x \in \mathbb{R}^d$. Then, we define our regularizer by the negative log likelihood of all patches under the probability distribution learned by the patchNR. More precisely, we define the patchNR based prior

$$\frac{1}{s}\sum_{i=1}^{N_p} -\log\left(p_{\mathcal{T}_{\#}P_Z}\left(E_i(x)\right)\right),$$

where N_p is the number of patches in the image x and $s = s_1 s_2$ the number of pixels in a patch. Similar to (3), this can be reformulated by the change-of-variables formula and by ignoring some constants as

$$\mathrm{patchNR}(x) := \frac{1}{s}\sum_{i=1}^{N_p} \frac{1}{2}\left\|\mathcal{T}^{-1}\left(E_i(x)\right)\right\|^2 - \log\left(\left|\det\left(\nabla\mathcal{T}^{-1}\left(E_i(x)\right)\right)\right|\right).$$

Note that if we ignore the boundary of the image, the patchNR is translation invariant. That is, a translation of the image does not change the value of the regularizer. Now, we reconstruct our ground truth by minimizing the variational problem

$$\mathcal{J}(x; y) = \mathcal{D}(f(x), y) + \lambda\,\mathrm{patchNR}(x), \quad \lambda > 0, \tag{4}$$

with respect to x. For the minimization, we use the Adam optimizer [42]. To speed up the numerical computations, we use a stochastic gradient descent with batch size N_p to minimize (4).

Note that the resulting optimization problem is non-convex and therefore the choice of the initialization is important. In our experiments we initialize this optimization with a coarse reconstruction, i.e. a bicubic interpolation for superresolution or the FBP for CT.

Remark 1 (Relation to EPLL). Our patchNR is closely related to the expected patch log likelihood (EPLL) prior proposed by [93]. Here, the authors use the prior defined as

$$\mathrm{EPLL}(x) = \frac{1}{N_p}\sum_{i=1}^{N_p} -\log\big(p\big(E_i(x)\big)\big),$$

where p is the probability density function of a GMM approximating the patch distribution of the image class of interest. However, GMMs have a limited expressiveness and can only hardly approximate complicated probability distributions induced by patches [23]. Further, the reconstruction process proposed in [93] is computationally very costly even though a reduction of the computational effort was considered in several papers [61, 72]. Indeed, we will show in our numerical examples that the patchNR clearly outperforms the reconstructions from EPLL.

3. Analysis of patch NFs

In this section, we investigate the patch distribution which is approximated by the patchNR. More precisely, we show that any probability density on the class of images induces a probability density on the space of patches and vice versa. We will use this to interpret the minimization of the variational problem (4) as maximizing the posterior distribution.

Remark 2. Considering the inverse problem (1) as a Bayesian inverse problem

$$Y = \mathrm{noisy}(f(X)),$$

where X and Y are random variables, Bayes' theorem implies that maximizing the log-posterior distribution $\log(p_{X|Y=y}(x))$ can be written as

$$\arg\max_x\{\log(p_{X|Y=y}(x))\} = \arg\max_x\Big\{-\log p_{Y|X=x}(y) - \log p_X(x)\Big\}.$$

Consequently, the data-fidelity term and the regularizer in (2) correspond to the negative log likelihood $\mathcal{D}(f(x),y) = -\log p_{Y|X=x}(y)$ and to the negative log prior $\mathcal{R}(x) = -\log p_X(x)$, respectively.

Let $X\colon \Omega \to \mathbb{R}^d$ with $X \sim P_X$ be a d-dimensional random variable on the space of images. By $\tilde{E}_i\colon \mathbb{R}^d \to \mathbb{R}^{d-s}$ we denote the linear operator which extracts all pixels from a d-dimensional image, which do not belong to the i-th patch. Further, with E_i^T and $\tilde{E}_i^T$ we designate the transposed operators of E_i and $\tilde{E}_i$, respectively. Let $I\colon \Omega \to \{1,\dots,N_p\}$ be a random variable which follows the uniform distribution on $\{1,\dots,N_p\}$. Then, the random variable $\omega \mapsto E_{I(\omega)}(X(\omega))$ describes the selection of a random patch from a random image. We call the distribution Q of $E_I(X)$ the patch distribution corresponding to P_X. The following lemma provides an explicit formula for the density of Q.

Lemma 3. *Let P_X be a probability distribution on $\mathbb{R}^d$ with density p_X and let I and X be stochastically independent. Then, also the corresponding patch distribution Q is absolutely continuous with density*

$$q(\mathrm{p}) = \frac{1}{N_p}\sum_{i=1}^{N_p}\int_{\mathbb{R}^{d-s}} p_X\big(E_i^{\mathrm{T}}(\mathrm{p}) + \tilde{E}_i^{\mathrm{T}}(\tilde{x})\big)\mathrm{d}\tilde{x}.$$

Proof. Let $A \in \mathcal{B}(\mathbb{R}^s)$ be an arbitrary Borel set. Then, we have by Bayes' theorem that

$$Q(A) = \sum_{i=1}^{N_p} \mathbb{P}(I=i) E_{i\#} P_X(A) = \frac{1}{N_p} \sum_{i=1}^{N_p} P_X(E_i^{-1}(A)) = \frac{1}{N_p} \sum_{i=1}^{N_p} \int_{\mathbb{R}^d} 1_A\big(E_i(x)\big) p_X(x) \mathrm{d}x.$$

Now, we use the decomposition

$$\mathbb{R}^d = \mathrm{Ker}(E_i) \oplus \mathrm{Im}(E_i^{\mathrm{T}}) = \mathrm{Im}(\tilde{E}_i^{\mathrm{T}}) \oplus \mathrm{Im}(E_i^{\mathrm{T}}).$$

With Fubini's theorem, $E_i E_i^{\mathrm{T}} = I$ and $E_i \tilde{E}_i^{\mathrm{T}} = 0$, this is equal to

$$\begin{aligned} Q(A) &= \frac{1}{N_p} \sum_{i=1}^{N_p} \int_{\mathbb{R}^s} \int_{\mathbb{R}^{d-s}} 1_A(E_i(E_i^{\mathrm{T}}(\mathrm{p}) + \tilde{E}_i^{\mathrm{T}}(\tilde{x}))) p_X(E_i^{\mathrm{T}}(\mathrm{p}) + \tilde{E}_i^{\mathrm{T}}(\tilde{x})) \mathrm{d}\tilde{x}\mathrm{d}\mathrm{p} \\ &= \frac{1}{N_p} \sum_{i=1}^{N_p} \int_{\mathbb{R}^s} \int_{\mathbb{R}^{d-s}} 1_A(\mathrm{p}) p_X(E_i^{\mathrm{T}}(\mathrm{p}) + \tilde{E}_i^{\mathrm{T}}(\tilde{x})) \mathrm{d}\tilde{x}\mathrm{d}\mathrm{p} \\ &= \int_A \frac{1}{N_p} \sum_{i=1}^{N_p} \int_{\mathbb{R}^{d-s}} p_X(E_i^{\mathrm{T}}(\mathrm{p}) + \tilde{E}_i^{\mathrm{T}}(\tilde{x})) \mathrm{d}\tilde{x}\mathrm{d}\mathrm{p}. \end{aligned}$$

This proves the claim. □

For the proof of the reverse direction, namely that given a probability measure Q on the space of patches, the patchNR defines a probability density function on the space of all images, we need the following lemma. It states that the density induced by a NF with a Gaussian latent distribution is up to a constant bounded from below and above by the density of certain normal distributions. In particular, it has exponential asymptotic decay. Note that similar questions about bi-Lipschitz continuous diffeomorphisms were investigated more detailed in [29, 36].

Lemma 4. *Let $\mathcal{T}\colon \mathbb{R}^s \to \mathbb{R}^s$ be a diffeomorphism with Lipschitz constants $\mathrm{Lip}(\mathcal{T}) \leqslant K$ and $\mathrm{Lip}(\mathcal{T}^{-1}) \leqslant L$ and let $P_Z = \mathcal{N}(0, I)$. Then, it holds*

$$\frac{1}{L^s K^s} \mathcal{N}\left(\mathrm{p} | \mathcal{T}(0), \frac{1}{L^2} I\right) \leqslant p_{\mathcal{T}_\# P_Z}(\mathrm{p}) \leqslant L^s K^s \mathcal{N}(\mathrm{p} | \mathcal{T}(0), K^2 I)$$

for any $\mathrm{p} \in \mathbb{R}^s$.

Proof. Using the Lipschitz continuity of $\mathcal{T}$, we obtain

$$\frac{1}{K^2} \|\mathrm{p} - \mathcal{T}(0)\|^2 = \frac{1}{K^2} \|\mathcal{T}(\mathcal{T}^{-1}(\mathrm{p})) - \mathcal{T}(0)\|^2 \leqslant \|\mathcal{T}^{-1}(\mathrm{p}) - 0\|^2 = \|\mathcal{T}^{-1}(\mathrm{p})\|^2.$$

Now, applying the change-of-variables formula and that $|\det(\nabla \mathcal{T}^{-1}(\mathrm{p}))| \leqslant L^s$, we conclude

$$\begin{aligned} p_{\mathcal{T}_\# P_Z}(\mathrm{p}) &= p_Z(\mathcal{T}^{-1}(\mathrm{p})) |\det(\nabla \mathcal{T}^{-1}(\mathrm{p}))| \leqslant \frac{L^s}{(2\pi)^{s/2}} \exp\left(-\frac{1}{2} \|\mathcal{T}^{-1}(\mathrm{p})\|^2\right) \\ &\leqslant \frac{L^s}{(2\pi)^{s/2}} \exp\left(-\frac{1}{2K^2} \|\mathrm{p} - \mathcal{T}(0)\|^2\right) = L^s K^s \mathcal{N}(\mathrm{p} | \mathcal{T}(0), K^2 I). \end{aligned}$$

This shows the second inequality. For the first inequality, note that by the inverse function theorem

$$\nabla \mathcal{T}^{-1}(\mathrm{p}) = \nabla \mathcal{T}^{-1}(\mathcal{T}(\mathcal{T}^{-1}(\mathrm{p}))) = (\nabla \mathcal{T}(\mathcal{T}^{-1}(\mathrm{p})))^{-1}.$$

Using the Lipschitz continuity of $\mathcal{T}$, this implies

$$\left|\det\left(\nabla \mathcal{T}^{-1}(\mathrm{p})\right)\right| = \left|\det\left(\nabla \mathcal{T}\left(\mathcal{T}^{-1}(\mathrm{p})\right)\right)\right|^{-1} \geqslant 1/K^s.$$

Further, by the Lipschitz continuity of $\mathcal{T}^{-1}$ it holds that

$$\|\mathcal{T}^{-1}(\mathrm{p})\|^2 = \|\mathcal{T}^{-1}(\mathrm{p}) - \mathcal{T}^{-1}(\mathcal{T}(0))\|^2 \leqslant L^2 \|\mathrm{p} - \mathcal{T}(0)\|^2.$$

Putting the things together yields

$$\begin{aligned} p_{\mathcal{T}_\# P_Z}(\mathrm{p}) = p_Z(\mathcal{T}^{-1}(\mathrm{p}))|\det(\nabla \mathcal{T}^{-1}(\mathrm{p}))| &\geqslant \frac{1}{K^s(2\pi)^{s/2}} \exp\left(-\frac{1}{2}\|\mathcal{T}^{-1}(\mathrm{p})\|^2\right) \\ &\geqslant \frac{1}{K^s(2\pi)^{s/2}} \exp\left(-\frac{L^2}{2}\|\mathrm{p} - \mathcal{T}(0)\|^2\right) = \frac{1}{L^s K^s} \mathcal{N}\left(\mathrm{p}|\mathcal{T}(0), \frac{1}{L^2} I\right). \end{aligned}$$

This completes the proof. □

Remark 5. Lemma 4 implies coercivity of $-\log(p_{\mathcal{T}_\# P_Z})$. Since every pixel is covered by at least one patch, this yields the coercivity of the patchNR. If the data fidelity term is bounded from below and continuous, this implies the existence of a minimizer for the variational problem (4).

Now, we can show that the patchNR defines a probability distribution on the space of all images. This allows a MAP interpretation of the variational problem (4).

Proposition 6. *Let $P_Z = \mathcal{N}(0, I)$ and let $\mathcal{T}\colon \mathbb{R}^s \to \mathbb{R}^s$ be a bi-Lipschitz diffeomorphism, i.e. $\mathrm{Lip}(\mathcal{T}) < \infty$ and $\mathrm{Lip}(\mathcal{T}^{-1}) < \infty$. Then, for any $\rho > 0$, the function $\varphi(x) = \exp(-\rho\,\mathrm{patchNR}(x))$ belongs to $L^1(\mathbb{R}^d)$, where*

$$\mathrm{patchNR}(x) = \frac{1}{s}\sum_{i=1}^{N_p} -\log\left(p_{\mathcal{T}_\# P_Z}(E_i(x))\right).$$

Proof. Using lemma 4, there exists some $C > 0$ such that it holds

$$\begin{aligned} \int_{\mathbb{R}^d} \varphi(x)\mathrm{d}x &= \int_{\mathbb{R}^d} \left(\prod_{i=1}^{N_p} p_{\mathcal{T}_\# P_Z}(E_i(x))\right)^{\rho/s} \mathrm{d}x \\ &\leqslant C \int_{\mathbb{R}^d} \left(\prod_{i=1}^{N_p} \mathcal{N}(E_i(x)|\mathcal{T}(0), K^2 I)\right)^{\rho/s} \mathrm{d}x \\ &= C \int_{\mathbb{R}^d} \prod_{i=1}^{N_p}\prod_{j=1}^{s} \mathcal{N}((E_i(x))_j|(\mathcal{T}(0))_j, K^2)^{\rho/s} \mathrm{d}x, \end{aligned}$$

where $(E_i(x))_j$ is the jth element from $E_i(x)$. Since $(E_i(x))_j = x_{\sigma(i,j)}$ for some mapping $\sigma\colon \{1,\dots,N_p\} \times \{1,\dots,s\} \to \{1,\dots,d\}$ and using Fubini's theorem, this simplifies to

$$
\begin{aligned}
&C \int_{\mathbb{R}^d} \prod_{i=1}^{N_p} \prod_{j=1}^{s} \mathcal{N}(x_{\sigma(i,j)} | (\mathcal{T}(0))_j, K^2)^{\rho/s} \mathrm{d}x \\
&= C \int_{\mathbb{R}^d} \prod_{k=1}^{d} \prod_{(i,j) \in \sigma^{-1}(\{k\})} \mathcal{N}(x_k | (\mathcal{T}(0))_j, K^2)^{\rho/s} \mathrm{d}x \\
&= C \prod_{k=1}^{d} \int_{\mathbb{R}} \prod_{(i,j) \in \sigma^{-1}(\{k\})} \mathcal{N}(x | (\mathcal{T}(0))_j, K^2)^{\rho/s} \mathrm{d}x.
\end{aligned}
$$

Here $\sigma^{-1}(\{k\})$ denotes the preimage of σ which denotes the set of all index pairs (i,j) such that $(E_i(x))_j = x_k$ for $x \in \mathbb{R}^d$. As each pixel in the images is covered by at least one patch, this set is non-empty. Using the fact that products and powers of normal densities are integrable, we obtain that this expression is finite and the proof is complete. □

4. Numerical examples

In this section, we demonstrate the performance of our method. We focus on linear inverse problems, but the approach can also be extended to non-linear forward operators. First, in section 4.3, we apply the patchNR to low-dose CT in a full angle and a limited angle setting and present an empirical convergence study for the optimization of the objective functional. Afterwards, in section 4.2, we consider superresolution on material data. This is a typical setting, where only little data is available and superresolution is needed to obtain sufficient detail for material research [17, 38, 65]. Lastly, we extend our findings to zero-shot superresolution in section 4.3. To get a better impression on the very good performance of our method, we added more numerical examples in appendix B[4].

Comparison Methods We compare our method with established methods from the literature, in particular with

- Wasserstein Patch Prior (WPP) [3, 31],
- EPLL [93],
- Local Adversarial Regularizer (localAR) [63],

since these methods also work on patches and are model-based. Note that we optimized the EPLL GMM prior using a gradient descent optimizer ourselves, as the half quadratic splitting proposed originally by the authors of [93] is much more expensive for the superresolution and CT forward operator. Moreover, we include comparisons with

- Plug-and-play forward backward splitting with DRUNet (DPIR) [91],
- Deep image prior in combination with a TV prior (DIP+TV) [10, 82],
- data-based methods as the post-processing UNet (FBP+UNet) [37, 68] for CT and an asymmetric CNN (ACNN) [81] for superresolution.

[4] The code for all experiments is written in PyTorch [62] and is available athttps://github.com/FabianAltekrueger/patchNR.

Details on the comparison methods are given in appendix A. Note that post-processing approaches are no longer the state-of-the-art for CT reconstruction, but are still widely used and serve as comparison methods. Currently some learned iterative methods provide better results [2].

Architecture of PatchNR For the architecture we use that there exists a universal approximation theorem [80] in weak topology. Therefore, a sufficiently large NF can approximate arbitrary probability distributions. We use five GlowCoupling blocks and permutations in an alternating manner, where the coupling blocks are from the freely available FrEIA package[5]. The three-layer subnetworks are fully connected with ReLU activation functions and 512 nodes resulting in 2908520 learnable parameters. The patchNR is trained on 6×6 patches, i.e. $s = 36$. For each image class, we trained the patchNR using Adam optimizer [42] with a learning rate of 0.0001, a batch size of 32 and for 750 000 optimizer steps. Training took about 2.5 h on a single NVIDIA GeForce RTX 2080 super GPU with 8 GB GPU memory.

4.1. Computed tomography

For CT we use the LoDoPaB dataset [51][6] for low-dose CT imaging. It is based on scans of the Lung Image Database Consortium and Image Database Resource Initiative [7] which serve as ground truth images, while the measurements are simulated. The dataset contains 35 820 training images, 3522 validation images and 3553 test images. Here the ground truth images have a resolution of 362×362px. The LoDoPab dataset uses a two-dimensional parallel beam geometry with equidistant detector bins. The forward operator is the discretized linear Radon transformation and the noise can be modelled using a Poisson distribution. Concretely, we consider the inverse problem

$$y = f(x) + \eta, \text{ where } \eta = -f(x) - \frac{1}{\mu}\log\left(\frac{\tilde{N}_1}{N_0}\right), \ \tilde{N}_1 \sim \mathrm{Pois}\big(N_0 \exp(-f(x)\mu)\big).$$

Here $N_0 = 4096$ is the mean photon count per detector bin without attenuation and $\mu = 81.358\,58$ is a normalization constant. In order to compute the corresponding data-fidelity term, we see that the observation y can be rewritten by

$$y = -\frac{1}{\mu}\log\left(\frac{\tilde{N}_1}{N_0}\right),$$

and thus $\exp(-y\mu)N_0 = \tilde{N}_1 \sim \mathrm{Pois}(\big(N_0 \exp(-f(x)\mu)\big)$. Since we assume that the pixels are corrupted independently and the negative log-likelihood of a Poisson distributed random variable is given by the Kullback–Leibler divergence, we have

$$\begin{aligned} -\log p(\exp(-y\mu)N_0 | \exp(-f(x)\mu)N_0) &= \sum_{i=1}^{d} -\log p(\exp(-y_i\mu)N_0 | \exp(-f(x)_i\mu)N_0) \\ &= \sum_{i=1}^{d} e^{-f(x)_i\mu}N_0 + e^{-y_i\mu}N_0\big(f(x)_i\mu - \log(N_0)\big). \end{aligned}$$

[5] Available at https://github.com/VLL-HD/FrEIA.

[6] Available at https://zenodo.org/record/3384092#.Ylglz3VBwgM.

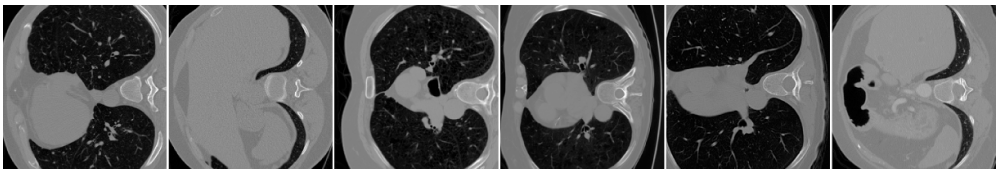

Figure 1. Ground truth images used for training the patchNR (CT).

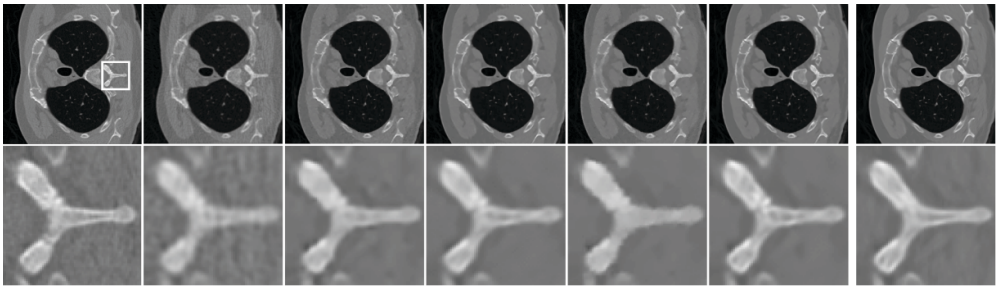

Figure 2. Full angle CT using different methods. The zoomed-in part is marked with a white box in the ground truth image. Our approach gives significantly better results than the model-based comparison methods. Top: full image. Bottom: zoomed-in part.

Table 1. Full angle CT. Averaged quality measures and standard deviations of the reconstructions. The best values are marked in bold.

	FBP	DIP + TV	EPLL	localAR	patchNR	FBP+UNet (data-based)
PSNR	30.37 ± 2.95	34.45 ± 4.20	34.89 ± 4.41	33.64 ± 3.74	**35.19** ± 4.52	35.48 ± 4.52
SSIM	0.739 ± 0.141	0.821 ± 0.147	0.821 ± 0.154	0.807 ± 0.145	**0.829** ± 0.152	0.837 ± 0.143
Runtime	0.03 s	1514.33 s	36.65 s	30.03 s	48.39 s	0.46 s

Thus, the concrete form of (4) we aim to minimize, is given by

$$\mathcal{J}(x;y) = \sum_{i=1}^{d} e^{-f(x)_i} N_0 + e^{-y_i} N_0 \big(f(x)_i - \log(N_0)\big) + \lambda \mathcal{R}(x).$$

We trained the patchNR using patches of a small set of $M = 6$ handpicked CT ground truth images of size 362×362 illustrated in figure 1. Once trained, the patchNR can be used both for the full angle CT and the limited angle CT setting, where we use a regularization parameter $\lambda = 700 \frac{s}{N_p}$, a random subset of $N_p = 40\,000$ overlapping patches in each iteration and the Adam optimizer with a learning rate of 0.005. To avoid boundary artifacts, we extend the image by reflection padding of 4 pixels when evaluating the patchNR. While for full angle CT we optimized over 300 iterations, for limited angle CT 3000 iterations are used.

Full angle CT For full angle CT we consider 1000 equidistant angles between 0 and π. In figure 2 we compare different methods for full angle low-dose CT imaging. Here the patchNR yields better results than DIP+TV and localAR, in particular the edges are sharper and more

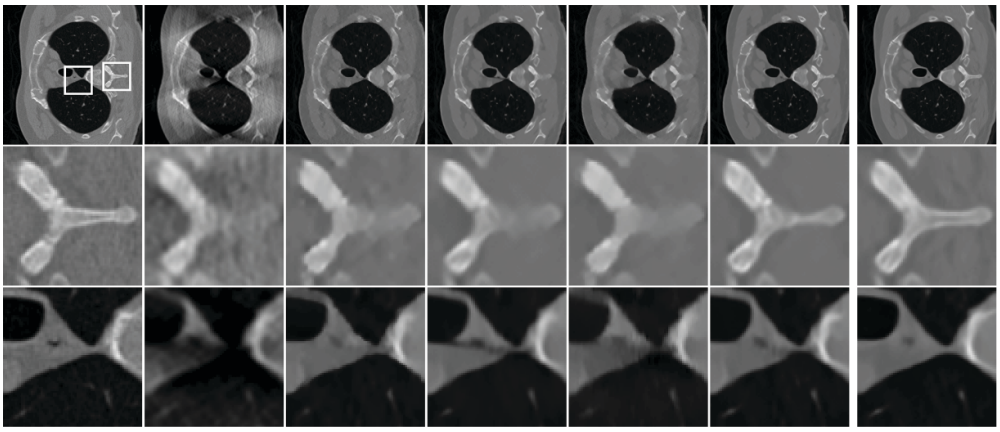

Figure 3. Limited angle reconstruction of the ground truth CT image using different methods. The zoomed-in part is marked with a white box in the ground truth image. The improvement of the image quality by our method is even better visible than in figure 2. Top: full image. Middle and bottom: zoomed-in parts.

Table 2. Limited angle CT. Averaged quality measures and standard deviations of the reconstructions. The best values are marked in bold.

	FBP	DIP + TV	EPLL	localAR	patchNR	FBP+UNet (data-based)
PSNR	21.96 ± 2.25	32.57 ± 3.25	32.78 ± 3.46	31.06 ± 2.95	**33.20** ± 3.55	33.75 ± 3.58
SSIM	0.531 ± 0.097	0.803 ± 0.146	0.801 ± 0.151	0.779 ± 0.142	**0.811** ± 0.151	0.820 ± 0.140
Runtime	0.02 s	1770.89 s	127.21 s	53.47 s	485.93 s	0.53 s

realistic in the reconstruction of patchNR. Visually, there are only small differences between patchNR and FBP+UNet observable, although FBP+UNet is a data-based method trained on 35 820 image pairs, while we only used 6 ground truth images for training the patchNR. The quality measures averaged over the first 100 test images of the LoDoPaB dataset in table 1 confirm these observations.Peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM) were evaluated on an adaptive data range. Note that the diversity of the test set causes relatively high standard deviations.

Limited angle CT Now we consider the limited angle CT reconstruction problem, i.e. instead of considering equidistant angles between 0 and π we only have a subset of angles. In our experiment we cut off the first and last 100 angles, i.e. we cut off 36° out of 180°. This leads to a much worse FBP reconstruction. In figure 3, we compare the different reconstruction methods for the limited angle problem. Although the FBP shows a very bad reconstruction in the 36° part, where the angles are cut off, the patchNR can reconstruct these details well and in a realistic manner. In particular, the edges of patchNR reconstruction are preserved, while for the other methods these have a pronounced blur, see table 2 for an average of the quality measures over the first 100 test images.

Empirical convergence analysis for full angle CT To reconstruct the ground truth image from given measurements y, we minimize the functional $\mathcal{J}(x;y)$ in (4) w.r.t. x using the Adam optimizer. The resulting optimization problem is non-convex and the final minimizer could

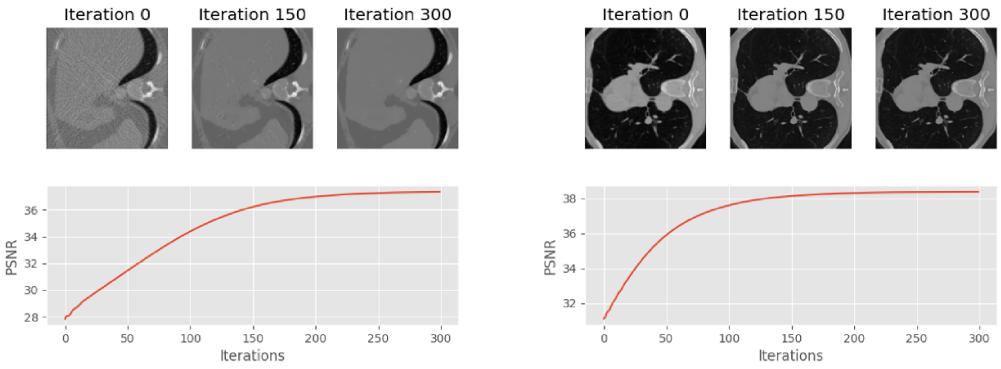

Figure 4. The PSNR for the first two test images per iteration of the optimization process.

Table 3. Comparison of full angle CT results for different patch sizes. Averaged quality measures and standard deviations of the reconstructions.

Patch size	$s = 4 \times 4$	$s = 6 \times 6$	$s = 8 \times 8$	$s = 10 \times 10$
PSNR	35.00 ± 4.45	35.19 ± 4.52	35.20 ± 4.58	35.17 ± 4.56
SSIM	0.825 ± 0.153	0.829 ± 0.152	0.827 ± 0.154	0.828 ± 0.154

depend on the initialization. In our experiments, it has proven useful to start the optimization with a rough reconstruction. For both full angle CT and limited angle CT we choose a FBP reconstruction. In order to empirically test the convergence of $\mathcal{J}(x;y)$ we evaluated the PSNR during the optimization process. In figure 4 we visualize the PSNR per iteration for the first two images of the test dataset and show reconstructions at iteration 0, 150 and 300. It can be seen that the PSNR is steadily rising during the optimization. Arguably for the left image in figure 4 we could have chosen even more iterations.

Ablation studies for full angle CT First, we trained the patchNR for patch sizes $4 \times 4, 6 \times 6, 8 \times 8$ and 10×10. In table 3 we tested the sensitivity w.r.t. the patch size. For all patch sizes we extracted 40 000 patches per iteration and used the optimal regularization parameter λ (which is set to $1600\frac{s}{N_p}$, $700\frac{s}{N_p}$, $400\frac{s}{N_p}$ and $250\ \frac{s}{N_p}$ for the patch sizes $4 \times 4, 6 \times 6, 8 \times 8$ and 10×10, respectively). We can observe that a larger patch size leads to slightly more blurry images. However the PSNR seems to change very little within different patch sizes, therefore we observe that our method is quite robust against the choice of the patch size.

Next, in table 4 we show reconstruction results for different numbers of patches used per iteration. Here we consider the patch size 6×6 and use the regularization parameter $\lambda = 700$. We see that the PSNR slightly increases with number of patches, while the SSIM seems to get worse at some point.

Finally, we examine the choice of the training set. In table 5 we evaluate the patchNR with patch size 6×6 and $\lambda = 700$, when trained on 1, 6 or 50 images. Obviously, for the CT dataset one training image is not enough to learn the patch distribution. This can be explained by the diversity of the CT dataset, see e.g. figure 1.

In table 6 we varied the training set of 6 images and evaluated the model on 3 different choices. In total we trained the patchNR 15 times on 6 randomly chosen training images of the LoDoPaB dataset and then evaluated on the test set. Again we consider the patch size 6×6

Table 4. Comparison of full angle CT results for varying number of patches per iteration. Averaged quality measures and standard deviations of the reconstructions.

Extracted patches per iteration	20 000	30 000	40 000	50 000	60 000
PSNR	35.04 ± 4.39	35.16 ± 4.49	35.19 ± 4.52	35.21 ± 4.55	35.21 ± 4.56
SSIM	0.829 ± 0.148	0.829 ± 0.151	0.829 ± 0.152	0.828 ± 0.153	0.828 ± 0.154

Table 5. Comparison of full angle CT results for different sets of 6 ground truth images. Averaged quality measures and standard deviations of the reconstructions.

Number of training images	1	6	50
PSNR	33.68 ± 3.57	35.19 ± 4.52	35.24 ± 4.60
SSIM	0.802 ± 0.127	0.829 ± 0.152	0.827 ± 0.156

Table 6. 40 000 extracted patches per iteration. Patch size $s = 6 \times 6$. Regularization parameter $\lambda = 700$. 6 random training images. Averaged quality measures and standard deviations of the reconstructions.

	Worst run	Our run	Best run	Mean $\pm$ standard deviation
PSNR	34.90 ± 4.39	35.19 ± 4.52	35.26 ± 4.60	35.13 ± 0.09
SSIM	0.825 ± 0.153	0.829 ± 0.152	0.828 ± 0.154	0.827 ± 0.001

and a regularization parameter $\lambda = 700$ and used 40 000 extracted patches per iteration. Note that the bad case in table 6 comes from a very noisy ground truth training set of the patchNR.

Overall, we see that the method is quite robust towards certain hyperparameter changes, and it can even be a matter of taste which ones to prefer as the image metrics do not always agree.

4.2. Superresolution

We choose the forward operator f as a convolution with a 16×16 Gaussian blur kernel with standard deviation 2 and subsampling with stride 4. To keep the dimensions consistent, we use zero-padding. For the experiments, we extract a dataset of 2D slices of size 600×600 from a 3D material image of size $2560 \times 2560 \times 2120$, which has been acquired by synchrotron micro-computed tomography. We consider a composite ('SiC Diamonds') obtained by microwave sintering of silicon and diamonds, see [83]. We generate observations which we call low resolution images by using the predefined forward operator and adding additive Gaussian noise with standard deviation $\sigma = 0.01$, i.e. we have

$$y = f(x) + \eta, \text{ where } \eta \sim \mathcal{N}(0, \sigma^2 I).$$

Consequently, from a Bayesian viewpoint the negative log likelihood $-\log(p_{Y|X=x}(y))$ can be, up to a constant, rewritten as

$$-\log(p_{Y|X=x}(y)) \propto -\log\left(\exp(-\|f(x)-y\|^2/(2\sigma^2)\right) = \frac{1}{2\sigma^2}\|f(x)-y\|^2.$$

Thus the concrete form of (4) is given by

$$\mathcal{J}(x) = \frac{1}{2\sigma^2}\|f(x)-y\|^2 + \rho\mathcal{R}(x) = \|f(x)-y\|^2 + \lambda\mathcal{R}(x),$$

with $\lambda = 2\rho\sigma^2$.

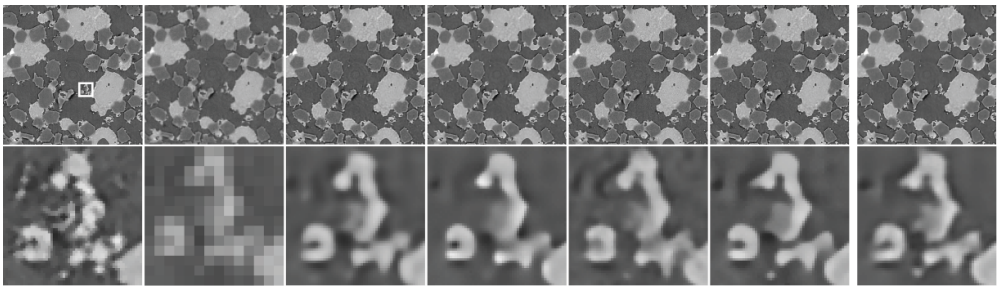

Figure 5. Comparison of different methods for superresolution. The zoomed-in part is marked with a white box in the ground truth image. Having in particular a look to disconnected parts in the lower right corner the improvement of the reconstruction by our method becomes evident. Top: full image. Bottom: zoomed-in part.

Table 7. Superresolution. Averaged quality measures and standard deviations of the high resolution reconstructions. The best values are marked in bold.

	bicubic (not shown)	DPIR (not shown)	DIP+TV	EPLL	WPP	patchNR	ACNN (data-based)
PSNR	25.63 ± 0.56	27.78 ± 0.53	27.99 ± 0.54	28.11 ± 0.55	27.80 ± 0.37	**28.53** ± 0.49	28.89 ± 0.53
SSIM	0.699 ± 0.012	0.770 ± 0.011	0.764 ± 0.007	0.779 ± 0.010	0.749 ± 0.011	**0.780** ± 0.008	0.804 ± 0.010
Runtime	0.0002 s	56.62 s	234.00 s	60.28 s	387.28 s	150.79 s	0.03 s

The patchNR is trained on patches of only $M = 1$ example image of size 600×600. For reconstruction, we use the regularization parameter $\lambda = 0.15\frac{s}{N_p}$, the random subset of overlapping patches is of size $N_p = 130000$ in each iteration and we optimize over 300 iterations using the Adam optimizer with a learning rate of 0.03. Since we do not want to consider boundary effects, we cut off a boundary of 40 pixels before evaluating the quality measures.

In figure 5 we compare different methods for reconstructing the high resolution image x from the given low-resolution observation y. As initialization we choose the bicubic interpolation. The patchNR yields very clear and better images than the other model-based methods, visually and in terms of the quality metrics; see table 7 for an average over 100 test images. In particular, the reconstruction of patchNR is less blurry than the DIP+TV and WPP reconstruction, specifically in regions between edges.

4.3. Zero-shot superresolution with PatchNRs

In the case of superresolution, we can train the patchNR even without any training image. To this end, we combine some concepts of zero-shot superresolution by internal learning [25, 73] with patchNRs. In these approaches, the main assumption is that the patch distribution of natural images is self-similar across the scales. Consequently, the patch distributions of the same image at different resolutions should be similar. Motivated by this observation, we train the patchNR on the low-resolution observations such that we do not longer require any sample from the high-resolution ground truth.

In the following, we consider the case, where we have given one single low-resolution observation at training time and additionally the forward operator at test time. In particular,

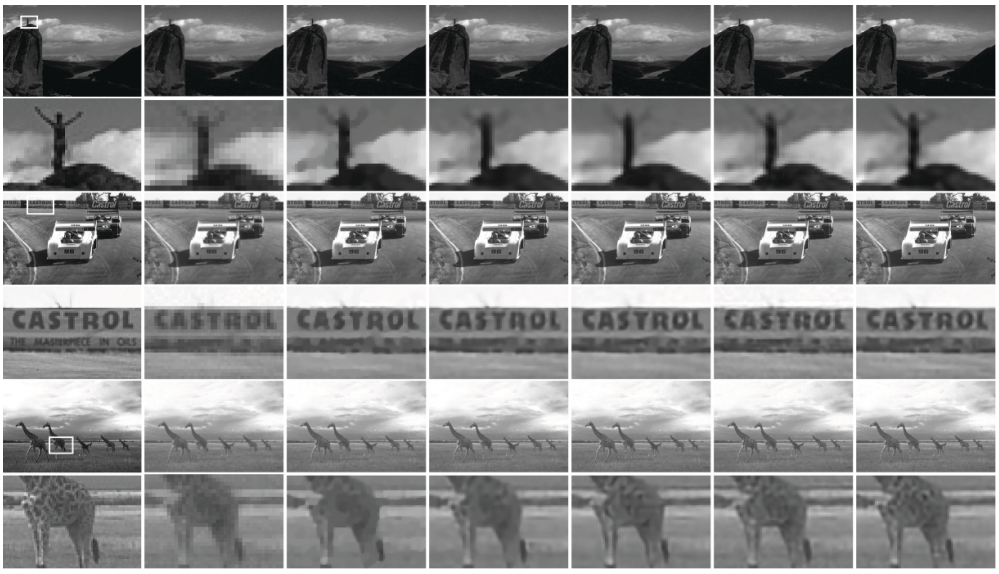

Figure 6. Zero-shot superresolution for three images from the BSD68 dataset. The zoomed-in part is marked with a white box in the ground truth image. Top: full image. Bottom: zoomed-in part. Reproduced from https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/

Table 8. Zero-shot superresolution. Averaged quality measures and standard deviations of the reconstructions of BSD68 dataset. The best values are marked in bold.

	L^2-TV	DIP+TV	ZSSR	DualSR	patchNR
PSNR	28.35 ± 3.55	28.44 ± 3.69	28.83 ± 3.57	28.64 ± 3.47	**29.08** ± 3.58
SSIM	0.820 ± 0.072	0.821 ± 0.087	0.834 ± 0.066	0.829 ± 0.061	**0.846** ± 0.061
Runtime	13.12 s	171.51 s	56.64 s	53.47 s	132.36 s

we do not require access to any high-resolution ground truth image and therefore the method is fully unsupervised. We train the patchNR on the patches from the low-resolution observation , where the training data is enriched by rotating and mirror reflecting of the patches such that we get 8 times more training patches. Note that in this setting the patchNR needs to be retrained for every new observation.

First we use a convolution with a 16×16 Gaussian blur kernel with standard deviation 1 and stride 2 as forward operator and add Gaussian noise with standard deviation 0.01 on the low-resolution observation. The patchNR is trained for 10 000 optimizer steps with a learning rate of 0.0001, a batch size of 128 and a patch size of 6×6. Then, for reconstruction, we use a random subset of $N_p = 80\,000$ patches per iteration, a regularization parameter $\lambda = 0.25\frac{s}{N_p}$ and optimize over 60 iterations using the Adam optimizer with a learning rate of 0.01. As comparison baselines we use L^2-TV [69], DIP+TV, ZSSR [73] and DualSR [21]. We test the methods on the BSD68 dataset [56]and the resulting quality measures are given in table 8. In figure 6 we present three reconstruction examples of the test set. Reconstructions of the patchNR lead to less blurry images and in particular structures and edges

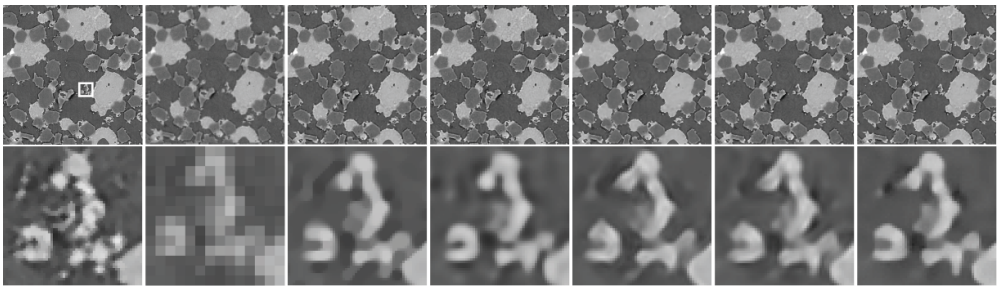

Figure 7. Zero-shot superresolution. The zoomed-in part is marked with a white box in the ground truth image. Top: full image. Bottom: zoomed-in part. Top: full image. Bottom: zoomed-in part.

Table 9. Zero-shot superresolution. Averaged quality measures and standard deviations of the high-resolution reconstructions. The best values are marked in bold.

	L^2-TV	DIP+TV	ZSSR	DualSR	patchNR
PSNR	27.85 ± 0.55	**27.99** ± 0.54	27.44 ± 0.55	27.64 ± 0.57	27.94 ± 0.55
SSIM	0.768 ± 0.008	0.764 ± 0.007	0.758 ± 0.008	0.764 ± 0.008	**0.776** ± 0.009
Runtime	38.11 s	234.00 s	42.43 s	46.55 s	120.47 s

are preserved. In contrast, in L^2-TV and DIP+TV some parts of the images are smoothed out.

In a second experiment we consider the same forward operator as in section 4.2, that is a convolution with a 16×16 Gaussian blur kernel with standard deviation 2 and stride 4, and add Gaussian noise with standard deviation 0.01 on the low-resolution observation. The patchNR is trained for 10 000 optimizer steps with a learning rate of 0.0001, a batch size of 128 and a patch size of 6×6. Then, for reconstruction, we use a random subset of $N_p = 50\,000$ patches per iteration, a regularization parameter $\lambda = 0.25 \frac{s}{N_p}$ and optimize over 60 iterations using the Adam optimizer with a learning rate of 0.01

We test the methods on the same test images as in section 4.2. In figure 7 we compare the reconstructions of the different methods. We can observe a visually similar quality of the reconstructions for the DIP+TV and patchNR, while the other methods lead to a significant blur in the reconstructions. This can be also seen in the resulting quality measures given in table 9. Note that here we do not consider natural images but material data and the assumption of self-similarity between different scales is not fulfilled. Consequently, we cannot expect a good reconstruction of the both methods ZSSR and DualSR. The methods L^2-TV and DIP+TV do not need these assumptions and thus perform well on the data, while there is a slight worsening in the quality of the patchNR in contrast to section 4.2. This nicely demonstrates the gain of using a small amount of ground truth data for training the patchNR.

5. Discussion and conclusions

In this paper, we introduced patchNRs, which are patch-based NFs used for regularizing the variational modeling of inverse problems. Learning patchNRs requires only few ground truth

images. In particular no paired data is necessary. We demonstrated the performance of our method by numerical examples and showed that it leads to better results than comparable, established methods, visually and in terms of the quality measures. Note that it is not clear how patch based learning influences biases in datasets. Using a small number of images can be very dangerous as these datasets are easily imbalanced. Further research needs to be invested into understanding how many images are sufficient for an adequate patch representation of a dataset and when the patch representation can be used as a prior for inverse problems. Moreover, quality measures for images are not sufficient for judging the quality of an image, in particular in medical applications. Further evaluation with medical expertise needs to be done before drawing any conclusions. Moreover, all patch-based methods are essentially limited in the sense that they can not capture global image correlations. Furthermore, NFs are often not able to capture out of distribution data [45], so if a patch is far away from the patch 'manifold', the likelihoods might be meaningless. However, it is an open question whether patch-based learning mitigates this effect.

The patchNR can be extended into several directions. First, we want to use the regularizer for training NNs in an model-based way for a fast reconstruction of several observations. Then the patchNR can be applied for uncertainty quantification by using, e.g. invertible architectures [5, 19, 28] or Langevin sampling methods [77].

Data availability statement

All data that support the findings of this study are included within the article (and any supplementary files).

Acknowledgments

F A acknowledge support from the German Research Foundation (DFG) under Germany's Excellence Strategy—The Berlin Mathematics Research Center MATH+ within the Project EF3-7, A D from (DFG; GRK 2224/1) and the Klaus Tschira Stiftung via the project MALDISTAR (Project Number 00.010.2019), P H from the DFG within the Project SPP 2298 'Theoretical Foundations of Deep Learning' and J H by the DFG within the Project STE 571/16-1. The data from section 4.2 has been acquired in the frame of the EU Horizon 2020 Marie Sklodowska-Curie Actions Innovative Training Network MUMMERING (MUltiscale, Multimodal and Multidimensional imaging for EngineeRING, Grant Number 765604) at the beamline TOMCAT of the SLS by A Saadaldin, D Bernard, and F Marone Welford. We acknowledge the Paul Scherrer Institut, Villigen, Switzerland for provision of synchrotron radiation beamtime at the TOMCAT beamline X02DA of the SLS. P H thanks Cosmas Heiß and Shayan Hundrieser for fruitful discussions.

Appendix A. Comparison methods

- **Bicubic interpolation.** For superresolution, the simplest comparison is the bicubic interpolation [41], which is based on the local approximation of the image by polynomials of degree 3.

- **FBP and UNet.** For CT a classical method is the FBP, described by the adjoint Radon transform [64]. We used the ODL implementation [1] for our experiments. There we choose the filter type Hann and a frequency scaling of 0.641. In order to improve the image quality of the FBP, a post-processing network can be learned. Here we consider the popular choice of a UNet (FBP+UNet) [68], which was used in [37] for CT imaging. We use the implementation from [52][7], which is a network with 610 673 parameters trained on the 35 820 training images of LoDoPaB dataset. Training took around 1 week.
- **Wasserstein Patch Prior.** The idea of the WPP [3, 31][8] is to use the Wasserstein-2 distance between the patch distribution of the reconstruction and the patch distribution of a given reference image. Here a high resolution reference image $\tilde{x}$ (or a high-resolution cutout) with a similar patch distribution as the unknown high resolution image x is needed. For a representation of structures of different sizes, x and $\tilde{x}$ are considered at different scales $x_1 = x, \tilde{x}_1 = \tilde{x}, x_l = Ax_{l-1}, \tilde{x}_l = A\tilde{x}_{l-1}$ for a downsampling operator A.
  Then the aim is to minimize the functional

$$\mathcal{J}(x) = \mathcal{D}(f(x), y) + \lambda \sum_{l=1}^{L} W_2^2(\mu_{x_l}, \mu_{\tilde{x}_l}),$$

  where the patch distributions of x and $\tilde{x}$ are defined by

$$\mu_{x_l} = \frac{1}{N_l} \sum_{i=1}^{N_l} \delta_{P_i x_l}, \mu_{\tilde{x}_l} = \frac{1}{\tilde{N}_l} \sum_{i=1}^{\tilde{N}_l} \delta_{P_i \tilde{x}_l}.$$

- **Deep Image Prior with TV regularization.** The idea of the DIP [82] is to solve the optimization problem

$$\hat{\theta} \in \arg\min_\theta \mathcal{D}(f(G_\theta(z)), y),$$

  where G_θ is a convolutional neural network with parameters θ and z is a randomly chosen input. Then, the reconstruction $\hat{x}$ is given by $\hat{x} = G_\theta(z)$. For CT we used a network with 2973 880 trainable parameters and for superresolution the network has 2162 561 trainable parameters. It was shown in [82] that DIP admits competitive results for many inverse problems. A combination of DIP with the TV (DIP+TV) regualizer was successfully used in [10][9] for CT reconstruction. Here the optimization problem is extended to

$$\hat{\theta} \in \arg\min_\theta \mathcal{D}(f(G_\theta(z)), y) + \mathrm{TV}((G_\theta(z)).$$

  Note that each reconstruction with the DIP+TV requires the training of a neural network. In contrast to WPP and patchNR, the DIP+TV is a data-free method, i.e. it does not require any clean image for training. We tested a pre-trained variant of the DIP+TV on the same training images. However, this did not improve the results significantly. Note that warm-start

[7] Available at https://jleuschn.github.io/docs.dival/dival.reconstructors.fbpunet_reconstructor.html.

[8] We use the original implementation from [31] available at https://github.com/johertrich/Wasserstein_Patch_Prior.

[9] For superresolution, we use the original implementation from [82] available at https://github.com/DmitryUlyanov/deep-image-prior in combination with a TV regulariser; for CT, we use the original implementation from [10] available at https://github.com/jleuschn/dival/blob/master/dival/reconstructors.

intialization techniques were proposed in [11]. Here, the authors observed faster reconstruction times for a pre-trained DIP but not significantly better results. Therefore we stick to the random initialization.

- **Plug-and-Play Forward Backward Splitting with DRUNet.** In Plug-and-Play methods, first introduced by [84], the main idea is to consider an optimization algorithm from convex analysis for solving (2) and to replace the proximal operator with respect to the regularizer by a more general denoiser. Here, modify the forward backward splitting algorithm

$$x_{n+1} = \mathrm{prox}_{\eta R}(x_n - \eta \nabla_x \mathcal{D}(f(x_n), y))$$

for minimizing the functional (2) by the iteration

$$x_{n+1} = \mathcal{G}(x_n - \eta \nabla_x \mathcal{D}(f(x_n), y)), \tag{A.1}$$

where $\mathcal{G}$ is a neural network trained for denoising natural images. We use the DRUNet (DPIR) from [91] as denoiser and run (A.1) for 100 iterations. Note that the denoiser is trained on natural images and not on images from the specific image domain. However, as we have given only very few clean images from the considered image domain it is impossible to train a denoiser with comparable quality on them.

- **Local Adversarial Regularizer.** The adversarial regularizer was introduced in [55] and this framework was recently applied for learning patch-based regularizers (localAR) [63]. The idea is to train a network r_θ as a critic between patch distributions in order to distinguish between clean and degraded patches. The network is trained by minimizing

$$D(\theta) = \mathbb{E}_{z\sim\mathbb{P}_c}\left[r_\theta(z)\right] - \mathbb{E}_{z\sim\mathbb{P}_n}\left[r_\theta(z)\right] + \mu\mathbb{E}_{z\sim\mathbb{P}_i}\left[(\|\nabla_z r_\theta(z)\|_2 - 1)^2\right],$$

where $\mathbb{P}_c$ and $\mathbb{P}_n$ are the distributions of clean and noisy patches, respectively, and $\mathbb{P}_i$ is the distribution of all lines connecting samples in $\mathbb{P}_c$ and $\mathbb{P}_n$. Then the aim is to minimize the functional

$$\mathcal{J}(x) = \mathcal{D}(f(x), y) + \lambda \frac{1}{|I|} \sum_{i\in\mathcal{I}} r_\theta(P_i(x)).$$

For our experiments we used the code of [63], but instead of patch size 15 we used the patch size 6 and replaced the fully convolutional discriminator by a discriminator with 2 convolutional layers, followed by 4 fully connected layers. This results in a network with 198 529 trainable parameters and a training time of 1 h.

- **Expected Patch Log Likelihood.** The EPLL prior [93] assumes that the patch distribution of the ground truth can be approximated by a GMM p fitted to the patch distribution of the reference images. Reconstruction is done by minimizing the functional

$$\mathcal{J}(x) = \mathcal{D}(f(x), y) - \lambda \sum_{i=1}^{N} p(P_i(x)).$$

The GMM we used has 200 components resulting in 140 400 trainable parameters and training takes 3 h. In [93] the authors used half quadratic splitting to optimize this objective function. For our experiments we implemented the GMM in PyTorch and used the Adam [42] optimizer. This is because we are not aware how to efficiently implement the half quadratic splitting for the CT forward operator.

- **Asymmetric CNN.** The ACNN [81] is a 23-layer CNN with 1071 104 parameters trained in a data-based way on 28 paired images pairs of the composite 'SiC Diamonds' using the L^2 loss function. Training took 10 h.
- **Zero Shot Super-Resolution.** For ZSSR [73] the main assumption is that the patch distribution of natural images is self-similar across the scales. Exploiting this fact, a lightweight CNN with 887 040 parameters is trained in a supervised fashion on a paired dataset generated by the low-resolution image itself. This dataset is created by downsampling the low-resolution image to obtain a lower-resolution image and is enlarged by data augmentation like random rotations, random crops or mirror reflections. A high-resolution prediction is then created by applying the trained model to the low-resolution observation. For our experiments we reimplemented the ZSSR.
- **DualSR.** The idea of DualSR [21][10] is a dual-path pipeline, where an upsampling GAN learns the upsampling process and a downsampling GAN learns the degradation model, trained on cropped parts of the low-resolution image. This method can be used for blind superresolution, but since we know the forward operator in our case, we replace the downsampling GAN by the given degradation process. The model has 247 490 trainable parameters.

[10] We use the original implementation available at https://github.com/memad73/DualSR.

Appendix B. Further examples

Here we give some more examples of our experiments from section 4; see figures B1–B3.

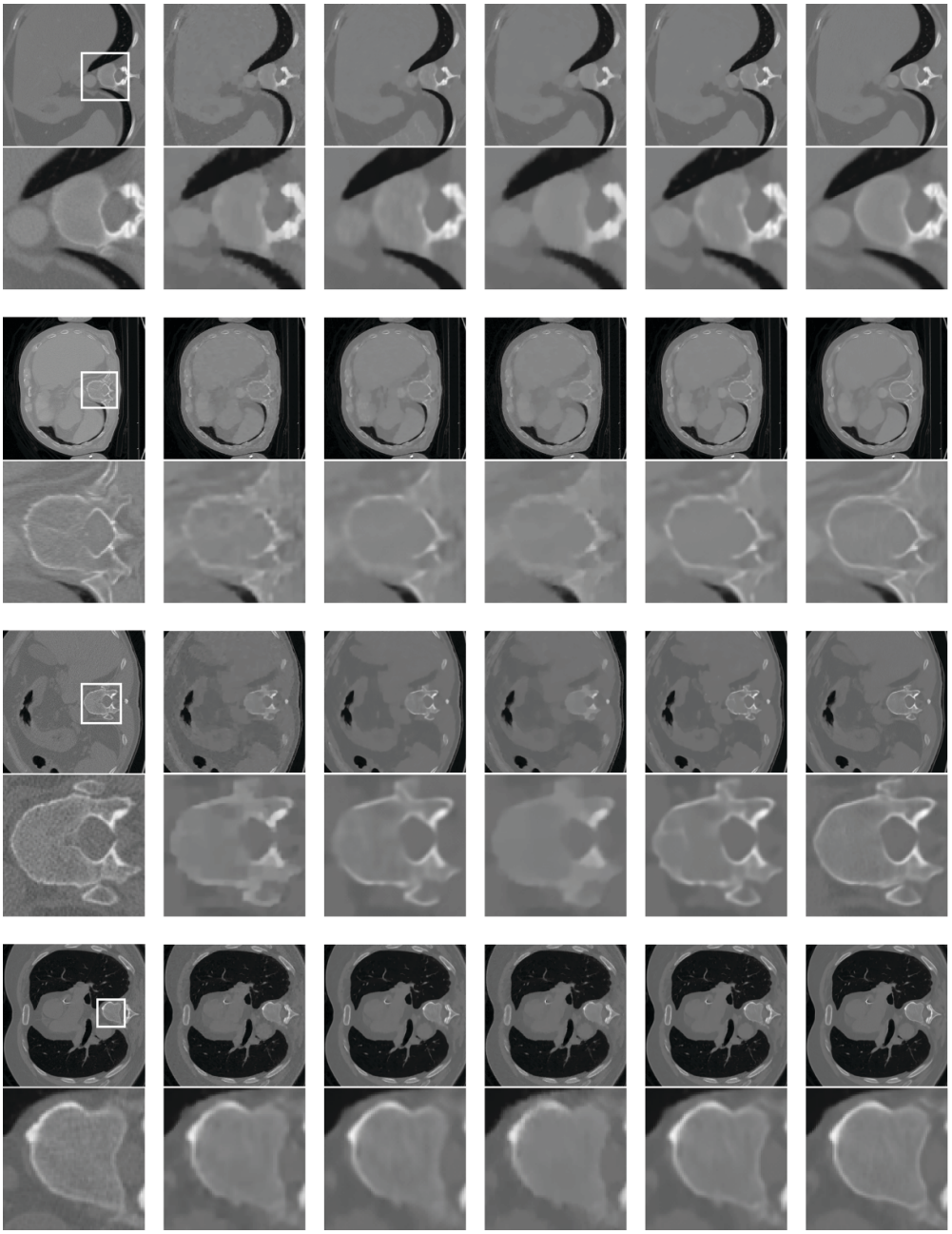

Figure B1. Full angle reconstruction of the ground truth CT image using different methods. The zoomed-in part is marked with a white box in the ground truth image. Top: full image. Bottom: zoomed-in part.

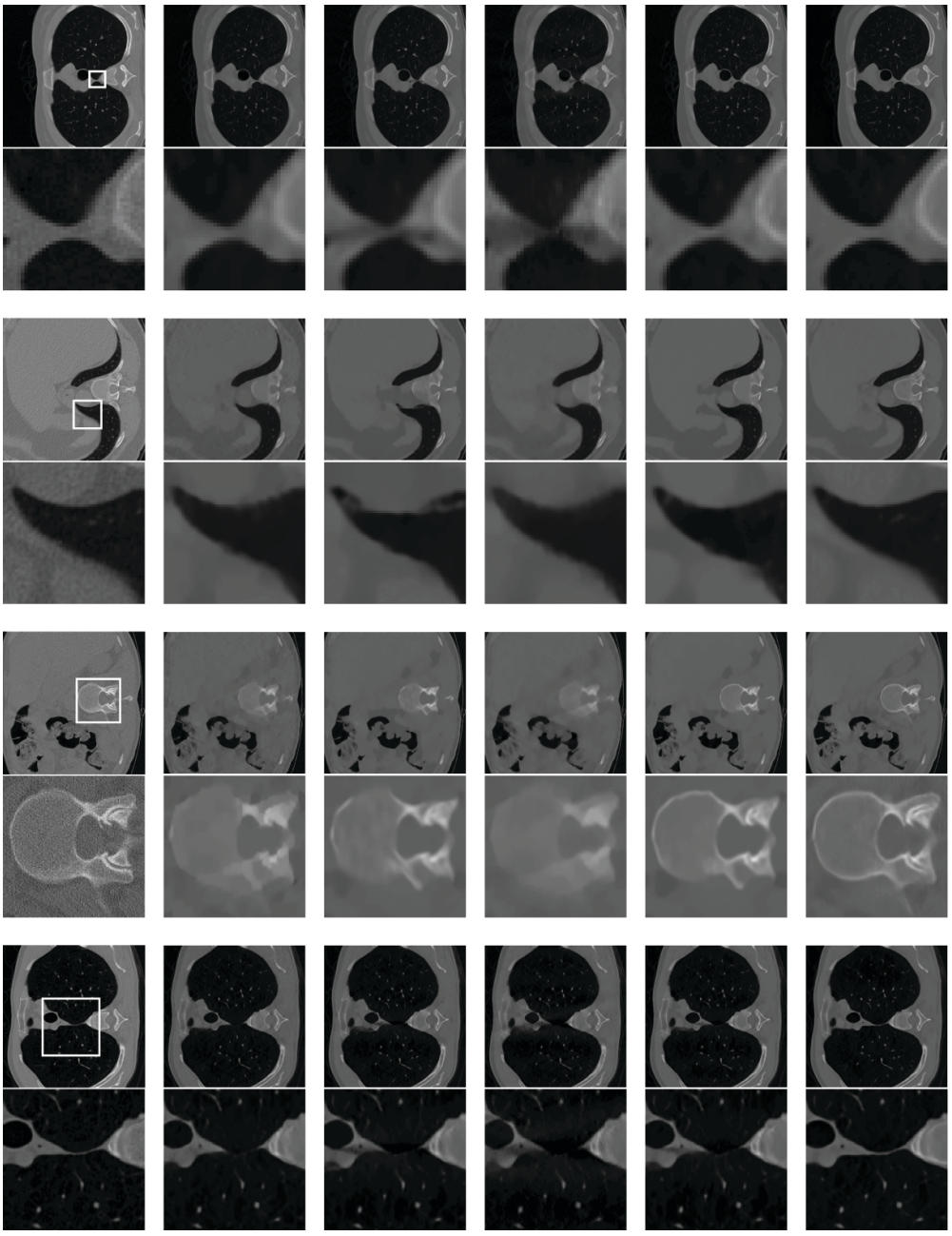

Figure B2. Limited angle reconstruction of the ground truth CT image using different methods. The zoomed-in part is marked with a white box in the ground truth image. Top: full image. Bottom: zoomed-in part.

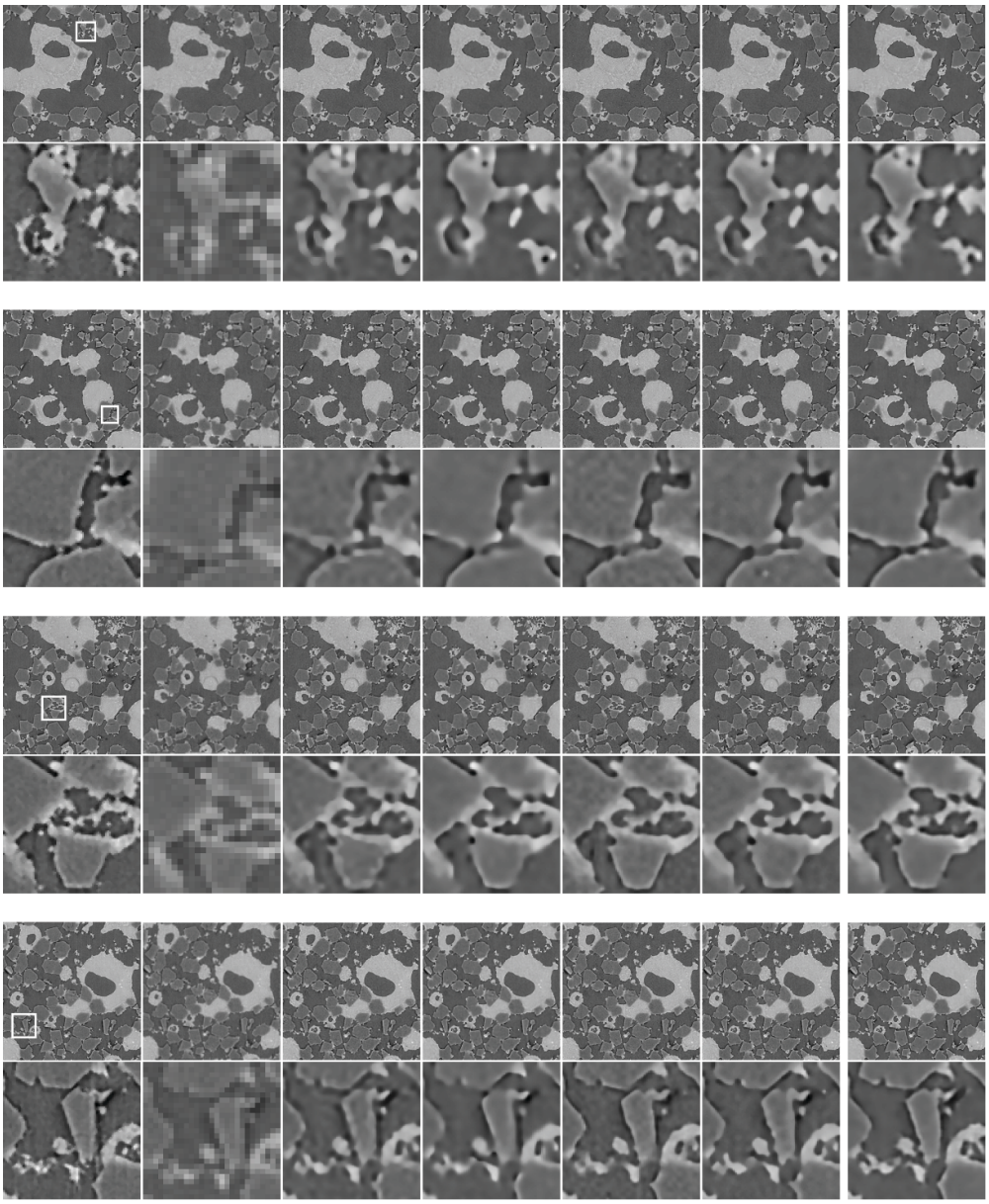

Figure B3. Comparison of different methods for superresolution. The zoomed-in part is marked with a white box in the ground truth image. *Top*: full image. *Bottom*: zoomed-in part.

ORCID iDs

Fabian Altekrüger https://orcid.org/0000-0002-9628-4735
Alexander Denker https://orcid.org/0000-0002-7265-261X
Paul Hagemann https://orcid.org/0009-0006-9679-5654
Johannes Hertrich https://orcid.org/0000-0003-4433-8604
Peter Maass https://orcid.org/0000-0003-1448-8345

References

[1] Adler J *et al* 2018 Operator discretization library (ODL) (https://doi.org/10.5281/zenodo.249479)
[2] Adler J and Öktem O 2018 Learned primal-dual reconstruction *IEEE Trans. Med. Imaging* **37** 1322–32
[3] Altekrüger F and Hertrich J 2022 WPPNets and WPPFlows: the power of Wasserstein patch priors for superresolution (arXiv:2201.08157)
[4] Anirudh R, Thiagarajan J J, Kailkhura B and Bremer T 2018 An unsupervised approach to solving inverse problems using generative adversarial networks (arXiv:1805.07281)
[5] Ardizzone L, Kruse J, Rother C and Köthe U 2019 Analyzing inverse problems with invertible neural networks *7th Int. Conf. on Learning Representations, ICLR 2019 (New Orleans, LA)*
[6] Ardizzone L, Lüth C, Kruse J, Rother C and Köthe U 2019 Guided image generation with conditional invertible neural networks (arXiv:1907.02392)
[7] Armato S G *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
[8] Arridge S, Maass P, Öktem O and Schönlieb C-B 2019 Solving inverse problems using data-driven models *Acta Numer.* **28** 1–174
[9] Asim M, Daniels M, Leong O, Ahmed A and Hand P 2020 Invertible generative models for inverse problems: mitigating representation error and dataset bias *Proc. 37th Int. Conf. on Machine Learning (Proc. Machine Learning Research)* vol 119, ed H D III and A Singh (PMLR) pp 399–409
[10] Baguer D O, Leuschner J and Schmidt M 2020 Computed tomography reconstruction using deep image prior and learned reconstruction methods *Inverse Problems* **36** 094004
[11] Barbano R, Leuschner J, Schmidt M, Denker A, Hauptmann A, Maaß P and Jin B 2021 Is deep image prior in need of a good education? (arXiv:2111.11926)
[12] Batzolis G, Carioni M, Etmann C, Afyouni S, Kourtzi Z and Schönlieb C B 2021 CAFLOW: conditional autoregressive flows (arXiv:2106.02531)
[13] Buades A, Coll B and Morel J-M 2005 A non-local algorithm for image denoising *2005 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition* vol 2 (IEEE) pp 60–65
[14] Chen R, Behrmann J, Duvenaud D K and Jacobsen J-H 2019 Residual flows for invertible generative modeling *Advances in Neural Information Processing Systems* vol 32 (Curran Associates, Inc.)
[15] Cheng Z, Xiong Z, Chen C, Liu D and Zha Z-J 2021 Light field super-resolution with zero-shot learning *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 10010–9
[16] Dabov K, Foi A, Katkovnik V and Egiazarian K 2009 BM3D image denoising with shape-adaptive principal component analysis *SPARS'09-Signal Processing With Adaptive Sparse Structured Representations*
[17] Dahari A, Kench S, Squires I and Cooper S J 2021 Super-resolution of multiphase materials by combining complementary 2d and 3d image data using generative adversarial networks (arXiv:2110.11281)
[18] Delon J and Houdard A 2018 Gaussian priors for image denoising *Denoising of Photographic Images and Video* (Berlin: Springer) pp 125–49
[19] Dinh L, Sohl-Dickstein J and Bengio S 2017 Density estimation using real NVP *5th Int. Conf. on Learning Representations, ICLR 2017 (Conf. Track Proc.) (Toulon, France)* (available at: OpenReview.net)
[20] Duff M, Campbell N D F and Ehrhardt M J 2021 Regularising inverse problems with generative machine learning models (arXiv:2107.11191)

[21] Emad M, Peemen M and Corporaal H 2021 DualSR: zero-shot dual learning for real-world super-resolution *Proc. IEEE/CVF Winter Conf. on Applications of Computer Vision* pp 1629–38
[22] Friedman R and Weiss Y 2021 Posterior sampling for image restoration using explicit patch priors (arXiv:2104.09895)
[23] Genovese C R and Wasserman L 2000 Rates of convergence for the Gaussian mixture sieve *Ann. Stat.* **28** 1105–27
[24] Gilton D, Ongie G and Willett R 2019 Learned patch-based regularization for inverse problems in imaging *2019 IEEE 8th Int. Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)* (IEEE) pp 211–5
[25] Glasner D, Bagon S and Irani M 2009 Super-resolution from a single image *2009 IEEE 12th International Conference on Computer Vision* (IEEE) pp 349–56
[26] Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A and Bengio Y 2014 Generative adversarial nets *Advances in Neural Information Processing Systems* vol 27, ed Z Ghahramani, M Welling, C Cortes, N Lawrence and K Weinberger (Curran Associates, Inc.)
[27] Granot N, Feinstein B, Shocher A, Bagon S and Irani M 2022 Drop the GAN: in defense of patches nearest neighbors as single image generative models *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)* pp 13460–9
[28] Hagemann P, Hertrich J and Steidl G 2021 Stochastic normalizing flows for inverse problems: a Markov Chains viewpoint (arXiV:2109.11375)
[29] Hagemann P and Neumayer S 2021 Stabilizing invertible neural networks using mixture models *Inverse Problems* **37** 085002
[30] Helminger L, Bernasconi M, Djelouah A, Gross M H and Schroers C 2021 Generic image restoration with flow based priors *2021 IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW)* pp 334–43
[31] Hertrich J, Houdard A and Redenbach C 2021 Wasserstein patch prior for image superresolution (arXiv:2109.12880)
[32] Hertrich J, Neumayer S and Steidl G 2021 Convolutional proximal neural networks and plug-and-play algorithms *Linear Algebr. Appl.* **631** 203–34
[33] Hertrich J, Nguyen D P L, Aujol J-F, Bernard D, Berthoumieu Y, Saadaldin A and Steidl G 2021 PCA reduced Gaussian mixture models with applications in superresolution *Inverse Problems Imaging* **15** 1135–69
[34] Houdard A, Bouveyron C and Delon J 2018 High-dimensional mixture models for unsupervised image denoising (HDMI) *SIAM J. Imaging Sci.* **11** 2815–46
[35] Hurault S, Leclaire A and Papadakis N 2022 Gradient step denoiser for convergent plug-and-play *Int. Conf. on Learning Representations*
[36] Jaini P, Kobyzev I, Yu Y and Brubaker M 2020 Tails of lipschitz triangular flows *Int. Conf. on Machine Learning* (PMLR) pp 4673–81
[37] 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
[38] Jung J, Na J, Park H K, Park J M, Kim G, Lee S and Kim H S 2021 Super-resolving material microstructure image via deep learning for microstructure characterization and mechanical behavior analysis *npj Comput. Mater.* **7** 96
[39] Kawar B, Elad M, Ermon S and Song J 2022 Denoising diffusion restoration models *ICLR Workshop on Deep Generative Models for Highly Structured Data*
[40] Kawar B, Vaksman G and Elad M 2021 SNIPS: solving noisy inverse problems stochastically *Advances in Neural Information Processing Systems* ed A Beygelzimer, Y Dauphin, P Liang and J W Vaughan
[41] Keys R 1981 Cubic convolution interpolation for digital image processing *IEEE Trans. Acoust. Speech Signal Process.* **29** 1153–60
[42] Kingma D P and Ba J 2015 Adam: a method for stochastic optimization *Int. Conf. on Learning Representations*
[43] Kingma D P and Dhariwal P 2018 Glow: generative flow with invertible 1x1 convolutions *Advances in Neural Information Processing Systems* vol 31
[44] Kingma D P and Welling M 2014 Auto-encoding variational bayes *2nd Int. Conf. on Learning Representations, ICLR 2014 (Conf. Track Proc.) (Banff, AB, Canada)* ed Y Bengio and Y LeCun

[45] Kirichenko P, Izmailov P and Wilson A G 2020 Why normalizing flows fail to detect out-of-distribution data *Advances in Neural Information Processing Systems* vol 33, ed H Larochelle, M Ranzato, R Hadsell, M Balcan and H Lin (Curran Associates, Inc.) pp 20578–89
[46] Kobler E, Effland A, Kunisch K and Pock T 2020 Total deep variation for linear inverse problems *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition* pp 7549–58
[47] Kobler E, Effland A, Kunisch K and Pock T 2021 Total deep variation: a stable regularization method for inverse problems *IEEE Trans. Pattern Anal. Mach. Intell.* **44** 9163–80
[48] Kohli M D, Summers R M and Geis J R 2017 Medical image data and datasets in the era of machine learning-whitepaper from the 2016 C-MIMI meeting dataset session *J. Digit. Imaging* **30** 392–9
[49] Laus F, Nikolova M, Persch J and Steidl G 2017 A nonlocal denoising algorithm for manifold-valued images using second order statistics *SIAM J. Imaging Sci.* **10** 416–48
[50] Lebrun M, Buades A and Morel J-M 2013 A nonlocal Bayesian image denoising algorithm *SIAM J. Imaging Sci.* **6** 1665–88
[51] Leuschner J, Schmidt M, Baguer D O and Maass P 2021 LoDoPaB-CT, a benchmark dataset for low-dose computed tomography reconstruction *Sci. Data* **8** 109
[52] Leuschner J *et al* 2021 Quantitative comparison of deep learning-based image reconstruction methods for low-dose and sparse-angle CT applications *J. Imaging* **7** 44
[53] Liang J, Lugmayr A, Zhang K, Danelljan M, Van Gool L and Timofte R 2021 Hierarchical conditional flow: a unified framework for image super-resolution and image rescaling *IEEE Int. Conf. on Computer Vision* pp 4076–85
[54] Lugmayr A, Danelljan M, Van Gool L and Timofte R 2020 SRFlow: Learning the super-resolution space with normalizing flow *European Conf. on Computer Vision*
[55] Lunz S, Öktem O and Schönlieb C-B 2018 Adversarial regularizers in inverse problems *Advances in Neural Information Processing Systems* vol 31
[56] Martin D, Fowlkes C, Tal D and Malik J 2001 A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics *Proc. 8th IEEE Int. Conf. on Computer Vision. ICCV 2001* vol 2 (IEEE) pp 416–23
[57] Mirza M and Osindero S 2014 Conditional generative adversarial nets (arXiv:1411.1784)
[58] Mukherjee S, Carioni M, Öktem O and Schönlieb C-B 2021 End-to-end reconstruction meets data-driven regularization for inverse problems *Advances in Neural Information Processing Systems* vol 34 pp 21413–25
[59] Ongie G, Jalal A, Metzler C A, Baraniuk R G, Dimakis A G and Willet R 2020 Deep learning techniques for inverse problems in imaging (arXiv:2005.06001)
[60] Pan X, Zhan X, Dai B, Lin D, Loy C C and Luo P 2020 Exploiting deep generative prior for versatile image restoration and manipulation *European Conf. on Computer Vision (ECCV)*
[61] Parameswaran S, Deledalle C, Denis L and Nguyen T Q 2019 Accelerating GMM-based patch priors for image restoration: three ingredients for a 100x speed-up *IEEE Trans. Image Process.* **28** 687–98
[62] Paszke A *et al* 2019 PyTorch: An Imperative Style, High-Performance Deep Learning Library *Advances in Neural Information Processing Systems* vol 32, ed H Wallach, H Larochelle, A Beygelzimer, F d'Alché Buc, E Fox and R Garnett (Curran Associates, Inc.) pp 8024–35
[63] Prost J, Houdard A, Almansa A and Papadakis N 2021 Learning local regularization for variational image restoration *Int. Conf. on Scale Space and Variational Methods in Computer Vision* (Springer) pp 358–70
[64] Radon J 1986 On the determination of functions from their integral values along certain manifolds *IEEE Trans. Med. Imaging* **5** 170–6
[65] Reid E J, Drummy L F, Bouman C A and Buzzard G T 2022 Multi-resolution data fusion for super resolution imaging *IEEE Trans. Comput. Imaging* **8** 81–95
[66] Rezende D and Mohamed S 2015 Variational inference with normalizing flows *Int. Conf. on Machine Learning* (PMLR) pp 1530–8
[67] Romano Y, Elad M and Milanfar P 2017 The little engine that could: regularization by denoising (RED) *SIAM J. Imaging Sci.* **10** 1804–44
[68] Ronneberger O, Fischer P and Brox T 2015 U-Net: convolutional networks for biomedical image segmentation *Medical Image Computing and Computer-Assisted Intervention (MICCAI)* (*LNCS* vol 9351) (Berlin: Springer) pp 234–41
[69] Rudin L I, Osher S and Fatemi E 1992 Nonlinear total variation based noise removal algorithms *Physica* D **60** 259–68

[70] Ruthotto L and Haber E 2021 An introduction to deep generative modeling *DMV Mitteilungen* **44** 1–24
[71] Sandeep P and Jacob T 2016 Single image super-resolution using a joint GMM method *IEEE Trans. Image Process.* **25** 4233–44
[72] Shi H, Traonmilin Y and Aujol J-F 2021 Compressive learning for patch-based image denoising *HAL Preprint hal-03429102*
[73] Shocher A, Cohen N and Irani M 2018 Zero-shot' super-resolution using deep internal learning *Proc. IEEE Conference on Computer Vision and Pattern Recognition* pp 3118–26
[74] Soh J W, Cho S and Cho N I 2020 Meta-transfer learning for zero-shot super-resolution *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*
[75] Sohn K, Lee H and Yan X 2015 Learning structured output representation using deep conditional generative models *Advances in Neural Information Processing Systems* vol 28
[76] Song Y and Ermon S 2019 Generative modeling by estimating gradients data distribution *Advances in Neural Information Processing Systems* vol 32, ed H Wallach, H Larochelle, A Beygelzimer, F d'Alché-Buc, E Fox and R Garnett (Curran Associates, Inc.)
[77] Song Y, Shen L, Xing L and Ermon S 2022 Solving inverse problems in medical imaging with score-based generative models *The 10th Int. Conf. on Learning Representations*
[78] Sreehari S, Venkatakrishnan S V, Wohlberg B, Buzzard G T, Drummy L F, Simmons J P and Bouman C A 2016 Plug-and-play priors for bright field electron tomography and sparse interpolation *IEEE Trans. Comput. Imaging* **2** 408–23
[79] Sun H, Mehta R, Zhou H, Huang Z, Johnson S, Prabhakaran V and Singh V 2019 DUAL-GLOW: Conditional flow-based generative model for modality transfer *IEEE Int. Conf. on Computer Vision* pp 10610–9
[80] Teshima T, Ishikawa I, Tojo K, Oono K, Ikeda M and Sugiyama M 2020 Coupling-based invertible neural networks are universal diffeomorphism approximators *Proc. 34th Int. Conf. on Neural Information Processing Systems (NIPS'20) (Vancouver, BC)*
[81] Tian C, Xu Y, Zuo W, Lin C-W and Zhang D 2021 Asymmetric CNN for image superresolution *IEEE Trans. Syst. Man Cybern.* **52** 3718–30
[82] Ulyanov D, Vedaldi A and Lempitsky V 2018 Deep image prior *Proc. IEEE Conf. on Computer Vision and Pattern Recognition* pp 9446–54
[83] Vaucher S, Unifantowicz P, Ricard C, Dubois L, Kuball M, Catala-Civera J-M, Bernard D, Stampanoni M and Nicula R 2007 On-line tools for microscopic and macroscopic monitoring of microwave processing *Physica* B **398** 191–5
[84] Venkatakrishnan S V, Bouman C A and Wohlberg B 2013 Plug-and-play priors for model based reconstruction *2013 IEEE Global Conf. on Signal and Information Processing* (IEEE) pp 945–8
[85] Wei X, Gorp H V, Carabarin L G, Freedman D, Eldar Y C and van Sloun R 2022 Deep unfolding with normalizing flow priors for inverse problems *IEEE Trans. Signal Process.* **70** 2962–71
[86] Whang J, Lei Q and Dimakis A 2021 Solving inverse problems with a flow-based noise model *Proc. 38th Int. Conf. on Machine Learning (Proc. Machine Learning Research)* vol 139, ed M Meila and T Zhang (PMLR) pp 11146–57
[87] Winkler C, Worrall D, Hoogeboom E and Welling M 2019 Learning likelihoods with conditional normalizing flows (arXiv:1912.00042)
[88] Xia W, Lu Z, Huang Y, Shi Z, Liu Y, Chen H, Chen Y, Zhou J and Zhang Y 2021 MAGIC: manifold and graph integrative convolutional network for low-dose ct reconstruction *IEEE Trans. Med. Imaging* **40** 3459–72
[89] Xu C S, Hayworth K J, Lu Z, Grob P, Hassan A M, García-Cerdán J G, Niyogi K K, Nogales E, Weinberg R J and Hess H F 2017 Enhanced FIB-SEM systems for large-volume 3D imaging *eLife* **6** e25916
[90] Yu J J, Derpanis K G and Brubaker M A 2020 Wavelet flow: fast training of high resolution normalizing flows *Advances in Neural Information Processing Systems* vol 33, ed H Larochelle, M Ranzato, R Hadsell, M Balcan and H Lin (Curran Associates, Inc.) pp 6184–96
[91] Zhang K, Li Y, Zuo W, Zhang L, Van Gool L and Timofte R 2021 Plug-and-play image restoration with deep denoiser prior *IEEE Trans. on Pattern Analysis and Machine Intelligence*
[92] Zhang Z, Yu S, Qin W, Liang X, Xie Y and Cao G 2020 Ct super resolution via zero shot learning (arXiv:2012.08943)
[93] Zoran D and Weiss Y 2011 From learning models of natural image patches to whole image restoration *IEEE Int. Conf. on Computer Vision* (IEEE) pp 479–86