Received: 13 January 2024 | Revised: 22 May 2024 | Accepted: 25 June 2024

DOI: 10.1002/gamm.202470002

REVIEW

Learning from small data sets: Patch-based regularizers in inverse problems for image reconstruction

Moritz Piening[1] | **Fabian Altekrüger**[2] | **Johannes Hertrich**[3] | **Paul Hagemann**[1] | **Andrea Walther**[2] | **Gabriele Steidl**[1]

[1] Institute of Mathematics, Technische Universität Berlin, Berlin, Germany

[2] Department of Mathematics, Humboldt-Universität zu Berlin, Berlin, Germany

[3] Department of Computer Science, University College London, London, UK

Correspondence
Moritz Piening, Institute of Mathematics, Technische Universität Berlin, Straße des 17. Juni 136, D-10623 Berlin, Germany.
Email: piening@math.tu-berlin.de

Funding information
EPSRC, Grant/Award Number: EP/V026259/1; EU Horizon 2020 Marie Sklodowska-Curie Actions Innovative Training Network MUMMERING, Grant/Award Number: 765604; Deutsche Forschungsgemeinschaft, Grant/Award Numbers: AA5-6, GRK2260, STE 571/16-1, STE 571/17-1

Abstract

The solution of inverse problems is of fundamental interest in medical and astronomical imaging, geophysics as well as engineering and life sciences. Recent advances were made by using methods from machine learning, in particular deep neural networks. Most of these methods require a huge amount of data and computer capacity to train the networks, which often may not be available. Our paper addresses the issue of learning from small data sets by taking patches of very few images into account. We focus on the combination of model-based and data-driven methods by approximating just the image prior, also known as regularizer in the variational model. We review two methodically different approaches, namely optimizing the maximum log-likelihood of the patch distribution, and penalizing Wasserstein-like discrepancies of whole empirical patch distributions. From the point of view of Bayesian inverse problems, we show how we can achieve uncertainty quantification by approximating the posterior using Langevin Monte Carlo methods. We demonstrate the power of the methods in computed tomography, image super-resolution, and inpainting. Indeed, the approach provides also high-quality results in zero-shot super-resolution, where only a low-resolution image is available. The article is accompanied by a GitHub repository containing implementations of all methods as well as data examples so that the reader can get their own insight into the performance.

KEYWORDS

computed tomography, generative neural networks, inpainting, inverse problems, Langevin Monte Carlo Sampling, small data sets, super-resolution, uncertainty quantification, Wasserstein distances, zero-shot learning

1 | INTRODUCTION

In medical and astronomical imaging, engineering, and life sciences, transformed image data of the form

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

is acquired based on a forward process underlying a physical model. In general, direct inversion of the forward operator F is not possible due to multiple solutions and/or amplification of "noise." This ill-posedness is critical in applications like image-guided medical diagnostics. Treating such problems by including prior knowledge of the desired images leads to a variational formulation of the problem of the form

$$\underset{x\in\mathbb{R}^d}{\arg\min}\ \mathcal{D}(F(x),y)+\beta\mathcal{R}(x).$$

Here, the first term is a "distance" term between the received data y and the acquisition model, where the chosen distance D reflects the noise model. The second one is an image prior, also called a regularizer, since it should force the variational problem to become well-posed, see [15, 16, 28, 108]. The choice of the image prior is more difficult. A prominent example is the total variation regularizer [91] and its vast amount of adaptations.

The past decades have witnessed a paradigm shift in data processing due to the emergence of the artificial intelligence revolution. Sophisticated optimization strategies based on the reverse mode of automatic differentiation methods [41], also known as backpropagation, were developed. The great success of deep learning methods has entered the field of inverse problems in imaging in quite different ways. For an overview of certain techniques, we refer to [11].

However, for many applications, there is only limited data available such that most deep learning based methods cannot be applied. In particular, for very high dimensional problems in image processing, the necessary amount of training data pairs is often out of reach and the computational costs for model training are high. On the other hand, the most powerful denoising methods before deep learning entered the field were patched-based as BM3D [20] or MMSE-based techniques [62, 63].

This review paper aims to advertise a combination of model-based and data-driven methods for learning from small data sets. The idea consists of retaining the distance term in the variational model and establishing a new regularizer that takes the internal image statistics, in particular the patch distribution of very few images into account. The main contribution of this paper is to summarize different approaches for constructing such patch-based regularizers and to compare their performance in terms of image reconstruction quality and uncertainty quantification. Our specific focus on small data sets complements the existing review literature on the regularization of inverse problems in imaging, for example, [11, 74, 78, 97]. To this end, we follow the path outlined below.

1.1 | Outline of the paper

We start by recalling Bayesian inverse problems in Section 2. In particular, we highlight the difference between

- the maximum a posteriori approach which leads to a variational model whose minimization provides one solution to the inverse problem, and
- the approximation of the whole posterior distribution from which we intend to sample in order to get, for example, uncertainty estimations.

We demonstrate by example the notations of well-posedness due to Hadamard and Stuart's Bayesian viewpoint.

Section 3 shows the relevance of internal image statistics. Although we will exclusively deal with image patches, we briefly sketch feature extraction by neural networks. Then we explain two different strategies to incorporate feature information into an image prior (regularizer). The first one is based on maximum likelihood estimations of the patch distribution which can also be formulated in terms of minimizing the forward Kullback–Leibler divergence. The second one penalizes Wasserstein-like divergences between the empirical measure obtained from the patches of the target image and an empirical reference measure obtained from the patches of a small set of reference images. Section 4 addresses three methods for parameterizing the function in the maximum likelihood approach, namely via Gaussian mixture models, the push-forward of a Gaussian by a normalizing flow, and a local adversarial approach. Section 5 shows three methods for choosing, based on the Wasserstein-2 distance, appropriate divergences for comparing the empirical patch measures. Having determined various patch-based regularizers, we use them to approximate the posterior measure and describe how to sample from this measure using a Langevin Monte Carlo approach in Section 6.

Section 7 illustrates the performance of the different approaches by numerical examples in computed tomography (CT), image super-resolution, and inpainting. Moreover, we consider zero-shot reconstructions in super-resolution. Further, we give an example for sampling from the posterior in inpainting and for uncertainty quantification in computed tomography. Since quality measures in image processing reflecting the human visual impressions are still a topic of research, we decided to give an impression of the different quality measures used in this section at the beginning.

The code base for the experiments is made publicly available on GitHub[1] to allow for benchmarking for future research. It includes ready-to-use regularizers within a common framework and multiple examples. Implementation on top of the popular programming language Python and the library PyTorch [83] that provides algorithmic differentiation enhances its accessibility.

Finally, note that alternative patch-based regularization strategies exist in addition to the presented ones, for example, based on patch-based denoisers [36] or an estimation of the latent dimension of the patch manifold [79].

2 | INVERSE PROBLEMS: A BAYESIAN VIEWPOINT

Throughout this paper, we consider digital gray-valued images of size $d_1 \times d_2$ as arrays $x \in \mathbb{R}^{d_1,d_2}$ or alternatively, by reordering their columns, as vectors $x \in \mathbb{R}^d$, $d = d_1 d_2$. For simplicity, we ignore that in practice gray values are encoded as finite discrete sets. The methods can directly be transferred to RGB color images by considering three arrays of the above form for the red, green, and blue color channels. For this setting, we first give a detailed Bayesian characterization of the inverse problems presented in the introduction.

In inverse problems in image processing, we are interested in the reconstruction of an image $x \in \mathbb{R}^d$ from its noisy measurement

$$y = \text{noisy}(F(x)), \tag{2}$$

where $F : \mathbb{R}^d \to \mathbb{R}^{\tilde{d}}$ is a forward operator and "noisy" describes the underlying noise model. In all applications of this paper, F is a known linear operator which is either not invertible as in image super-resolution and inpainting or ill-conditioned as in computed tomography, so that the direct inversion of F would amplify the noise. A typical noise model is additive Gaussian noise, resulting in

$$y = F(x) + \xi, \tag{3}$$

where ξ is a realization of a Gaussian random variable $\Xi \sim \mathcal{N}(0, \sigma^2 I_{\tilde{d}})$. Recall that the density function of the normal distribution $\mathcal{N}(m, \Sigma)$ with mean $m \in \mathbb{R}^{\tilde{d}}$ and covariance matrix $\Sigma \in \mathbb{R}^{\tilde{d},\tilde{d}}$ is given by

$$\varphi(x|\ m, \Sigma) := (2\pi)^{-\frac{\tilde{d}}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(x-m)^\intercal \Sigma^{-1}(x-m)\right). \tag{4}$$

More generally, we may assume that x itself is a realization of a continuous random variable $X \in \mathbb{R}^d$ with law P_X determined by the density function $p_X : \mathbb{R}^d \to [0, \infty)$ with $\int_{\mathbb{R}^d} p_X(x)\,\mathrm{d}x = 1$, that is, x is a sample from P_X. Then we can consider the random variable

$$Y = F(X) + \Xi, \quad \Xi \sim \mathcal{N}(0, \sigma^2 I_{\tilde{d}}) \tag{5}$$

and the posterior distribution $P_{X|Y=y}$ for given $y \in \mathbb{R}^{\tilde{d}}$. The crucial law to handle this is Bayes' rule

$$\underbrace{p_{X|Y=y}(x)}_{\text{posterior}} = \frac{\overbrace{p_{Y|X=x}(y)}^{\text{likelihood}}\ \overbrace{p_X(x)}^{\text{prior}}}{\underbrace{p_Y(y)}_{\text{evidence}}}. \tag{6}$$

Now we can ask at least for three different quantities.

1. MAP estimator: The maximum a posteriori (MAP) estimator provides the value with the highest probability of the posterior

$$x_{\mathrm{MAP}}(y) \in \underset{x\in\mathbb{R}^d}{\arg\max}\left\{p_{X|Y=y}(x)\right\} = \underset{x\in\mathbb{R}^d}{\arg\max}\left\{\log p_{X|Y=y}(x)\right\}. \tag{7}$$

By Bayes' rule (6) and since the evidence is constant with respect to x, this can be rewritten as

$$x_{\mathrm{MAP}}(y) \in \underset{x\in\mathbb{R}^d}{\arg\max}\left\{\log p_{Y|X=x}(y) + \log p_X(x)\right\}. \tag{8}$$

[1] https://github.com/MoePien/PatchbasedRegularizer.

The first term depends on the noise model, while the second one on the distribution within the image class. Assuming that $p_{Y|X=x}(y) = C\exp(-\mathcal{D}(Fx,y))$ for an appropriate choice of discrepancy or metric D and a *Gibbs prior* distribution

$$p_X(x) = C_\beta \exp(-\beta\mathcal{R}(x)), \tag{9}$$

we arrive at the variational model for solving inverse problems

$$x_{\mathrm{MAP}}(y) \in \underset{x\in\mathbb{R}^d}{\arg\min}\left\{\underbrace{\mathcal{D}(F(x),y)}_{\text{data term}} + \beta\underbrace{\mathcal{R}(x)}_{\text{prior}}\right\}, \quad \beta > 0. \tag{10}$$

Instead of a "prior" term, $\mathcal{R}$ is also known as a "regularizer" in inverse problems since it often transfers the original ill-posed or ill-conditioned problem into a well-posed one. By Hadamard's definition, this means that for any y there exists a unique solution that continuously depends on the input data. For example, for Gaussian noise as in (5) we have that $F(x) + \Xi \sim \mathcal{N}(F(x), \sigma^2 I_{\tilde{d}})$ so that by (4) we get

$$\log p_{Y|X=x}(y) = \log\left(2\pi\sigma^2\right)^{-\frac{\tilde{d}}{2}} - \frac{1}{2\sigma^2}\|F(x)-y\|^2, \tag{11}$$

which results with $\alpha := \sigma^2\beta$ in

$$x_{\mathrm{MAP}}(y) \in \underset{x\in\mathbb{R}^d}{\arg\min}\left\{\frac{1}{2}\|F(x)-y\|^2 + \alpha\mathcal{R}(x)\right\}. \tag{12}$$

2. Posterior distribution: Here we are searching for a measure $P_{X|Y=y} \in \mathcal{P}(\mathbb{R}^d)$ and not as in MAP for a single sample that is most likely for a given y, where $\mathcal{P}(\mathbb{R}^d)$ is the space of probability measures on $\mathbb{R}^d$. We will see that approximating the posterior, which mainly means to find a way to sample from it, provides a tool for uncertainty quantification. It was shown in [60, 106] that the posterior $P_{X|Y=y}$ is often locally Lipschitz continuous with respect to y, that is,

$$\mathrm{d}(P_{X|Y=y_1}, P_{X|Y=y_2}) \le L\|y_1 - y_2\|$$

with some $L > 0$ and a discrepancy d between measures as the Kullback–Leibler divergence or Wasserstein distances explained in Section 3. Indeed, this Lipschitz continuity is the key feature of Stuart's formulation of a well-posed *Bayesian* inverse problem [107] as a counterpart of Hadamard's definition.

There are only few settings in (5) where the posterior can be computed analytically, see [39, 47], namely if X is distributed by a *Gaussian mixture model* (GMM) $X \sim \sum_{k=1}^K \alpha_k \mathcal{N}(m_k, \Sigma_k) \in \mathbb{R}^d$, that is,

$$p_X = \sum_{k=1}^K \alpha_k \varphi(\cdot|m_k,\Sigma_k), \quad \sum_{k=1}^K \alpha_k = 1,\ \alpha_k > 0, \tag{13}$$

the forward operator $F \in \mathbb{R}^{\tilde{d},d}$ is linear and $\Xi \sim N(0, \sigma^2 I_{\tilde{d}})$. Then it holds

$$p_{X|Y=y} = \sum_{k=1}^K \tilde{\alpha}_k \varphi(\cdot|\tilde{m}_k, \tilde{\Sigma}_k) \tag{14}$$

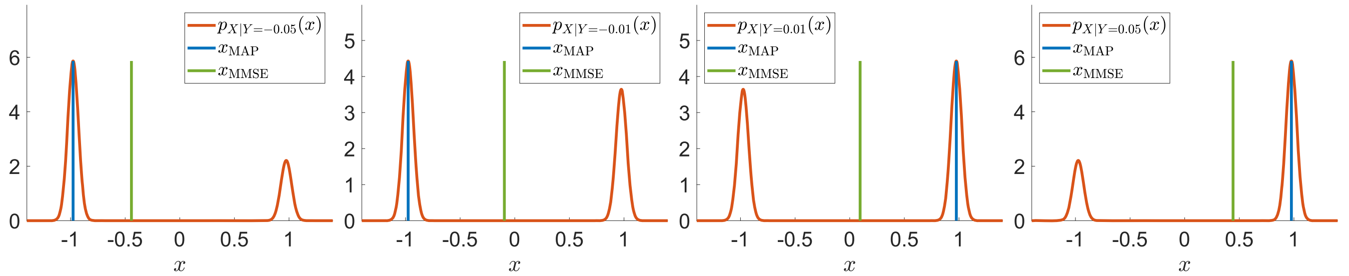
with

$$\tilde{\Sigma}_k := \left(\frac{1}{\sigma^2}F^{\mathrm{T}}F + \Sigma_k^{-1}\right)^{-1}, \quad \tilde{m}_k := \tilde{\Sigma}_k\left(\frac{1}{\sigma^2}F^{\mathrm{T}}y + \Sigma_k^{-1}\mu_k\right), \tag{15}$$

$$\tilde{\alpha}_k := \frac{\alpha_k}{|\Sigma_k|^{1/2}}\exp\left(\frac{1}{2}(\tilde{m}_k^{\mathrm{T}}\tilde{\Sigma}_k^{-1}\tilde{m}_k - m_k^{\mathrm{T}}\Sigma_k^{-1}m_k)\right). \tag{16}$$

3. MMSE estimator: The maximum mean square error (MMSE) estimator is just the expected value of the posterior, that is,

$$x_{\mathrm{MMSE}}(y) = \mathbb{E}[X|Y=y] = \int_{\mathbb{R}^d} x p_{X|Y=y}(x)\,\mathrm{d}x. \tag{17}$$



$y = -0.05$ $y = -0.01$ $y = 0.01$ $y = 0.05$

FIGURE 1 Posterior density (red), MAP estimator (blue), and MMSE (green) for different observations $y = -0.05, -0.01, 0.01, 0.05$ (from left to right) given $X \sim \frac{1}{2}\mathcal{N}(-1, \varepsilon^2) + \frac{1}{2}\mathcal{N}(1, \varepsilon^2)$ with $\varepsilon^2 = 0.05^2$, $\Xi \sim \mathcal{N}(0, \sigma^2)$ with $\sigma^2 = 0.1$ and $F = I$. While the MAP estimator is discontinuous with respect to the observation y at zero, the posterior is continuous with respect to y. Its expectation value, the MMSE, is far away from the value with the highest probability. Image is taken from [4].

If $X \sim \mathcal{N}(m, \Sigma)$, F is linear and $\Xi \sim N(0, \sigma^2 I_{\tilde{d}})$, then the MMSE can be computed analytically by

$$x_{\mathrm{MMSE}}(y) = m + \Sigma F^{\mathrm{T}}(F\Sigma F^{\mathrm{T}} + \sigma^2 I_{\tilde{d}})^{-1}(y - Fm). \tag{18}$$

We would like to note that for more general distributions the estimator (18) is known as the *best linear unbiased estimator* (BLUE). MMSE techniques in conjunction with patch-based techniques were among the most powerful techniques for image denoising before ML-based methods entered the field, see [62, 64].

A simple example of the different estimators in the case of a Gaussian mixture with two components is visualized in Figure 1 [4]. There, the posterior is again a Gaussian mixture, where the MAP estimator is given by the mode of its biggest Gaussian component, and the MMSE estimator is a weighted average of the two components. In the numerical experiments we will consider the MAP estimator in Sections 7.2, 7.3, 7.4, and the posterior sampling in Section 7.5.

3 | INTERNAL IMAGE STATISTICS

In this paper, when dealing with the MAP estimator, that is, with problems of the form (10), we follow a physics-informed approach, where both the forward operator and the noise model are known. Then the data term $\mathcal{D}(F(x), y)$ is completely determined. The challenging part is the modeling of the prior distribution P_X, where we only know samples from. In contrast to deep learning methods which rely on a huge amount of ground truth data, we are in a situation, where only one or a few images are available. Then, instead of working with the distribution P_X of whole images in the prior, we consider typical features of the images and ask for the feature distribution. These features live in a much lower dimensional space than the images. Indeed, one key finding in image processing was the expressiveness of this internal image statistics [100, 109]. Clearly, there are many ways to extract meaningful features and we refer only to the "field of experts" framework [90] here. In the following, we explain two typical choices of meaningful features, namely image patches and features obtained from a nonlinear filtering process of a neural network.

3.1 | Image patches

Image patches are square-shaped (or rectangular) regions of size $p \times p$ within an image $x \in \mathbb{R}^{d_1, d_2}$ which can be extracted by operators $\mathcal{P}_i : \mathbb{R}^{d_1, d_2} \to \mathbb{R}^{p,p}$, $i = (i_1, i_2) \in \{1, \dots, d_1\} \times \{1, \dots, d_2\}$ via $\mathcal{P}_i(x) = (x_{l_1, l_2})_{l_1 = i_1, l_2 = i_2}^{i_1 + p, i_2 + p}$. The patch extraction is visualized in Figure 2 (left). The use of such patches for image reconstruction has a long history [25, 84] and statistical analyses of empirical patch distributions reveal their importance to image characterization [120]. Furthermore, the patch distributions are similar at different scales for many image classes. Therefore, the approach is not sensitive to scale shifts. Figure 3 illustrates this behavior, see also Figure 16. Indeed, replication of patch distributions by means of patch sampling [67] or statistical distance minimization [55] enables the synthesis of high-quality texture images. Neural network image generators are able to generate diverse outputs on the basis of patch discriminators [96, 98] or patch distribution matching [27, 40]. Furthermore, patch-matching methods have successfully been employed for style transfer [19].

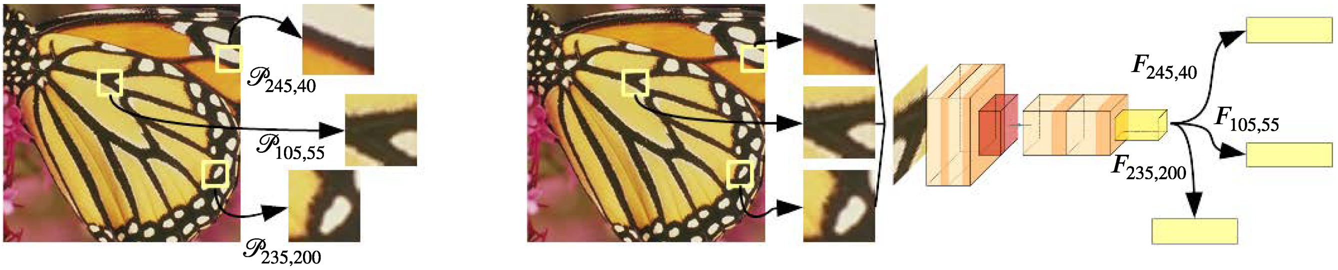


FIGURE 2 Visualization of the process of patch extraction (left) and hidden feature extraction (right).

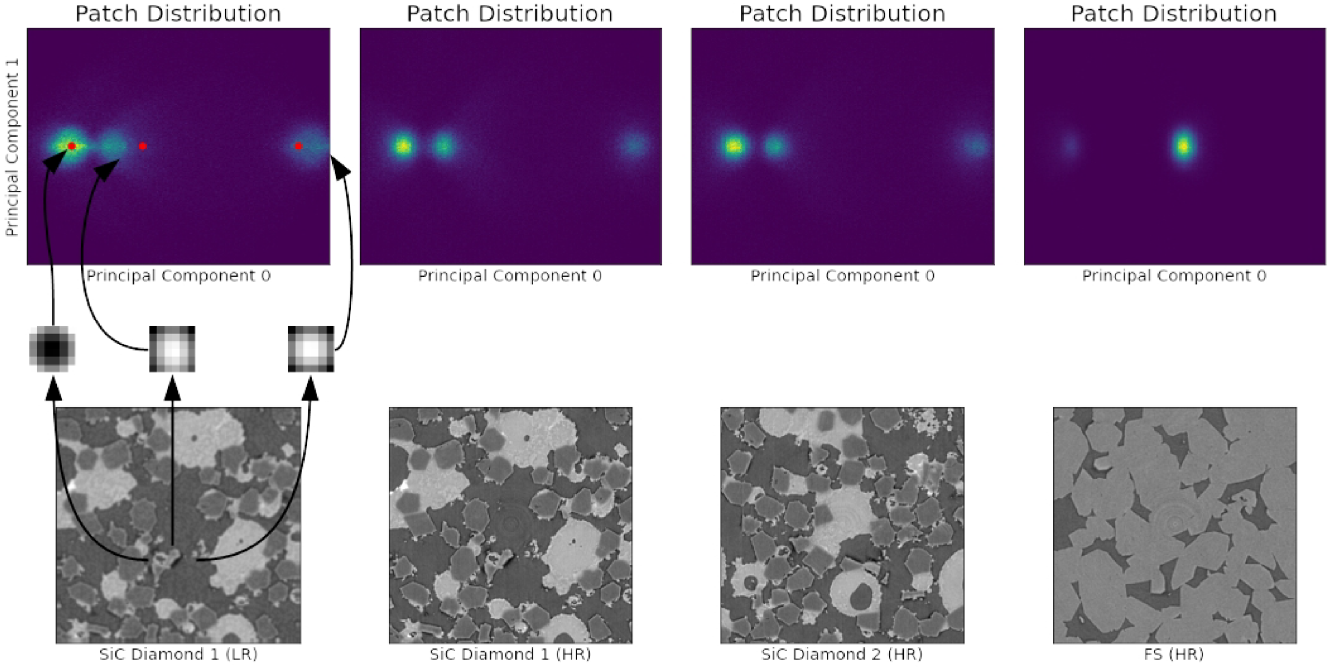


FIGURE 3 Left to right: Illustration of the patch distribution of a low resolution (LR) image (downsampled) from a composite of silicon and diamonds ("SiC Diamond"), two different high-resolution (HR) images from the same material and one from the material Fontainebleau sandstone ("FS"). Patches of size 6 × 6 from the respective images are extracted and a principal component analysis is applied to the corresponding vectors in $\mathbb{R}^{36}$ in order to project onto the plane spanned by the two principal directions of the largest patch variance. The empirical distribution of the first two principal components in the form of 2D histograms is depicted in the top row. All patches of HR images of the first material have a similar distribution, which is easy to distinguish from those of the second material. This is also true for the LR image but with a slightly larger spread of the clusters. Example patches (red dots) from the histogram are displayed in the upper row, left.

3.2 | Neural network filtered features

Several feature methods for image reconstruction, as, for example, in the "field of experts" framework [90] are based on features obtained from various linear filter responses possibly finally followed by an application of a nonlinear function. More recently, such techniques were further extended by using a pre-trained classification convolutional neural network, for example, a VGG architecture trained on the ImageNet dataset [101]. In each layer, multiple nonlinear filters (convolutions and component-wise nonlinear activation function) are applied to the downsampled result of the previous layer. Typically, the outputs of the first convolutional layer after a downsampling step are utilized as *hidden features*. This is illustrated in Figure 2 right. Since every nonlinear filter of a convolutional filter is applied locally, these extracted features represent nonlinear transformations of patches of different sizes. The use of such features has been pioneered by Gatys et al. [33], who used them to construct a loss function for the style transfer between two images based on a statistical distance between their hidden feature distributions [66]. Furthermore, such features have been utilized for texture synthesis in [32, 56] and for image similarity comparison in [96, 119]. Due to excessive pre-training, they may be able to capture semantic information which can be helpful in reconstruction tasks such as inpainting [69, 105], denoising [115] or super-resolution [31, 82]. However, unlike patches, the extraction of hidden features often relies on networks trained on large datasets. This hinders their application in the setting of small data sets.

3.3 | Internal image statistics in image priors

In the rest of the paper, we will concentrate on patches as features because their use requires no pre-training and is therefore favorable in the context of limited data availability. We assume that we are given a small number n of images from an image class, say, $n = 1$ high-resolution material image or $n = 6$ computed tomography scans. For simplicity, let us enumerate the patch operators by $\mathscr{P}_i$, $i = 1, \dots, N$. Further, let us denote by Q the *patch distribution*. Then we will follow two different strategies to incorporate them into the prior $\mathcal{R}$ of model (10), which we describe next.

1. **Patch maximum log-likelihood**: We approximate the patch distribution Q by a distribution Q_θ with density q_θ depending on some parameter θ. Then, we learn its parameter via a *maximum log-likelihood* (ML) estimator:

$$\hat{\theta} = \arg\max_\theta \left\{ \prod_{j=1}^{n} \prod_{i=1}^{N} q_\theta\big(\mathscr{P}_i(x_j)\big) \right\} = \arg\max_\theta \left\{ \sum_{j=1}^{n} \sum_{i=1}^{N} \log\Big(q_\theta\big(\mathscr{P}_i(x_j)\big)\Big) \right\} \tag{19}$$

$$= \arg\min_\theta \left\{ \underbrace{-\sum_{j=1}^{n} \sum_{i=1}^{N} \log\Big(q_\theta\big(\mathscr{P}_i(x_j)\big)\Big)}_{=:\mathcal{L}(\theta)} \right\}. \tag{20}$$

Once the optimal parameter $\hat{\theta}$ is determined by minimizing the *loss function* $\mathcal{L}(\theta)$, we can use

$$\mathcal{R}(x) := -\frac{1}{N} \sum_{i=1}^{N} \log\big(q_{\hat{\theta}}(\mathscr{P}_i(x))\big) \tag{21}$$

as a prior in our minimization problem (10). Indeed this value should become small, if the patches of the wanted image x are distributed according to $q_{\hat{\theta}}$. Concrete choices for families of probability distributions $\{q_\theta\}_{\theta \in \Theta}$ are presented in Section 4.

The above model can be derived from another point of view using the *Kullback–Leibler* (KL) *divergence* between Q and Q_θ. As a measure divergence, the KL is non-negative and becomes zero if and only if both measures coincide. For Q and Q_θ on $\mathbb{R}^d$, $d = p^2$ with densities q and q_θ, respectively, the KL divergence is given (if it exists) by

$$\mathrm{KL}(Q, Q_\theta) = \int_{\mathbb{R}^d} \log\left(\frac{q(x)}{q_\theta(x)}\right) q(x)\,\mathrm{d}x \tag{22}$$

$$= \int_{\mathbb{R}^d} \underbrace{\log(q(x))q(x)\,\mathrm{d}x}_{\text{const}} - \int_{\mathbb{R}^d} \log(q_\theta(x))q(x)\,\mathrm{d}x. \tag{23}$$

Skipping the constant part with respect to θ, this becomes

$$\mathrm{KL}(Q, Q_\theta) \propto -\int_{\mathbb{R}^d} \log(q_\theta(x))q(x)\,\mathrm{d}x = -\mathbb{E}_{x\sim X}\big[\log(q_\theta(x))\big].$$

Here $\propto$ denotes equality up to an additive constant. Replacing the expectation value by the empirical one and neglecting the factor $\frac{1}{nN}$, we arrive exactly at the loss function $\mathcal{L}(\theta)$ in (20).

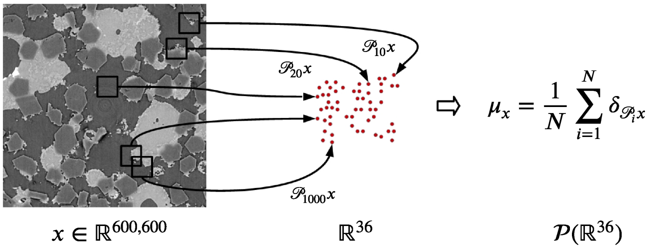

$\mathscr{P}_{10}x$
$\mathscr{P}_{20}x$
$\mathscr{P}_{1000}x$
$\Rightarrow \quad \mu_x = \frac{1}{N}\sum_{i=1}^{N}\delta_{\mathscr{P}_i x}$
$x \in \mathbb{R}^{600,600}$
$\mathbb{R}^{36}$
$\mathcal{P}(\mathbb{R}^{36})$

FIGURE 4 From patches to measures.

2. **Divergences between empirical patch measures**: We can associate empirical measures to the image patches by

$$\nu := \frac{1}{nN}\sum_{j=1}^{n}\sum_{i=1}^{N}\delta_{\mathscr{P}_i(x_j)} \quad \text{and} \quad \mu_x := \frac{1}{N}\sum_{i=1}^{N}\delta_{\mathscr{P}_i(x)} \tag{24}$$

as illustrated in Figure 4, where μ_x relates to the target image x and ν is constructed from a small set of training images. Then we use a prior

$$\mathcal{R}(x) := \mathrm{dist}(\mu_x, \nu), \tag{25}$$

with some distance, respectively divergence, between measures.

Our distances of choice in (25) will be Wasserstein-like distances. Let $\mathcal{P}_p(\mathbb{R}^d)$, $p \in [1, \infty)$, denote the set of probability measures with finite pth moments. The Wasserstein-p distance $W_p : \mathcal{P}_p(\mathbb{R}^d) \times \mathcal{P}_p(\mathbb{R}^d) \to \mathbb{R}$ is defined by

$$W_p^p(\mu, \nu) := \inf_{\gamma \in \Pi(\mu,\nu)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|x - y\|^p \, \mathrm{d}\pi(x, y), \tag{26}$$

where $\Pi(\mu, \nu) := \{\pi \in \mathcal{P}(\mathbb{R}^d \times \mathbb{R}^d) : (\mathrm{proj}_1)_\# \pi = \mu, (\mathrm{proj}_2)_\# \pi = \nu\}$ is the set of all couplings with marginals μ and ν. Here proj_i, $i = 1, 2$ denote the projection onto the ith marginals. Further, we used the notation of a push-forward measure. In general, for a measurable function $T : \mathbb{R}^d \to \mathbb{R}^{\tilde{d}}$ and a measure μ on $\mathbb{R}^d$, the *push-forward measure* of μ by T on $\mathbb{R}^{\tilde{d}}$ is defined as

$$T_\#\mu(A) = \mu\big(T^{-1}(A)\big), \quad \text{i.e.,} \quad \int_A g(y) \, \mathrm{d}(T_\#\mu)(y) = \int_{T^{-1}(A)} g(T(x)) \, \mathrm{d}\mu(x)$$

for all $g \in C_0(\mathbb{R}^{\tilde{d}})$ and for all Borel measurable sets $A \subseteq \mathbb{R}^{\tilde{d}}$. The push-forward measure and the corresponding densities p_μ and $p_{T_\#\mu}$ for a differentiable and invertible function T are related by the *transformation formula*

$$p_{T_\#\mu}(x) = p_\mu\big(T^{-1}(x)\big) \left|\det\big(\nabla T^{-1}(x)\big)\right|. \tag{27}$$

An example of the Wasserstein-2 distance is given in Figure 7.

Both of these approaches allow us to define a regularization on the space of images by identifying each image with its patch distribution. Note that both strategies can easily be generalized to multi-scale regularization by adding the composition $\mathcal{R} \circ D$ for a downsampling operator D.

4 | PATCH MAXIMUM LOG-LIKELIHOOD PRIORS

For the prior in (21), it remains to find an appropriate parameterized function $p(\cdot|\theta)$. In the following, we present three different regularizers, namely obtained via Gaussian mixture models (GMM-EPLL), normalizing flows (patchNR), and adversarial neural networks (ALR).

We will compare their performance in inverse problems later in the experimental section.

4.1 | Gaussian mixture model

A classical approach assumes that the patch distribution can be approximated by a GMM (13), that is,

$$q_\theta(x) = \sum_{k=1}^{K} \alpha_k \varphi(x \mid m_k, \Sigma_k), \quad \theta = (\alpha_k, m_k, \Sigma_k)_{k=1}^{K}. \tag{28}$$

This is justified by the fact that any probability distribution can be approximated arbitrarily well in the Wasserstein distance by a GMM [22]. However, the number of modes K has to be fixed in advance. Then the maximization problem becomes

$$\hat{\theta} = \arg\max_{\theta} \left\{ \sum_{j=1}^{n} \sum_{i=1}^{N} \log\left(\sum_{k=1}^{K} \alpha_k \varphi\left(\mathcal{P}_i(x_j) \,|\, m_k, \Sigma_k\right) \right) \right\}.$$

This is typically solved by the Expectation-Maximization (EM) Algorithm 1, with the guarantee of convergence to a local maximizer, see [23]. The corresponding regularizer (21) becomes

$$\mathcal{R}(x) = \mathrm{EPLL}(x) := \tfrac{1}{N} \sum_{i=1}^{N} -\log\left(\sum_{k=1}^{K} \alpha_k \varphi(\mathcal{P}_i(x) \,|\, m_k, \Sigma_k) \right).$$

It was suggested for solving inverse problems under the name expected patch log-likelihood (EPLL) by Zoran and Weiss [121].

Algorithm 1. Expectation-maximization for Gaussian mixture model

```
Input: Patches $\{x_1, \dots, x_M\}$, $M = Nn$, number of GMM components $K$, stopping criterion
Output: GMM parameters $\left\{m_k, \Sigma_k, \alpha_k\right\}_{k=1}^{K}$
Initialization: $\left\{m_k^{(0)}, \Sigma_k^{(0)}, \alpha_k^{(0)}\right\}_{k=1}^{K}$
for $r = 0, 1, \dots$ until stopping criterion do
    1. Expectation step: For $k = 1, \dots, K$ and $i = 1, \dots, M$ compute
           $$\beta_{i,k}^{(r)} = \frac{\alpha_k^{(r)} \varphi\left(x_i | m_k^{(r)}, \Sigma_k^{(r)}\right)}{\sum_{j=1}^{K} \alpha_j^{(r)} \varphi\left(x_i | m_j^{(r)}, \Sigma_k^{(r)}\right)}.$$
    2. Maximization step: For $k = 1, \dots, K$ and $i = 1, \dots, M$update parameters
           $$m_k^{(r+1)} = \frac{\sum_{i=1}^{M} \beta_{i,k}^{(r)} x_i}{\sum_{i=1}^{M} \beta_{i,k}^{(r)}},$$
           $$\Sigma_k^{(r+1)} = \frac{\sum_{i=1}^{M} \beta_{i,k}^{(r)} \left(x_i - m_k^{(r+1)}\right)\left(x_i - m_k^{(r+1)}\right)^{\intercal}}{\sum_{i=1}^{M} \beta_{i,k}^{(r)}},$$
           $$\alpha_k^{(r+1)} = \frac{1}{M} \sum_{i=1}^{M} \beta_{i,k}^{(r)}$$
end for
```

Remark 1. By the relation (9) the EPLL defines a prior distribution $p_X(x) = C_\beta \exp(-\beta \mathrm{EPLL}(x))$ on the space of images. The integrability of the function p_X can be shown by similar arguments as in the proof of [3, Prop. 6].

While originally the variational formulation (10) with the EPLL was solved using *half quadratic splitting*, in our implementation we use a stochastic gradient descent for minimizing (10). Meanwhile there exist many extensions and improvements: GMMs may be replaced with other families of distributions [21, 49, 50, 77] or multiple image scales can be included [80]. The intrinsic dimension of the Gaussian components can be restricted as in [54, 111] or in the PCA reduced GMM model [52]. Finally, image restoration can be accelerated by introducing flat-tail Gaussian components, balanced search trees, and restricting the sum of the EPLL to a stochastically chosen subset of patch indices [81]. For the inclusion of learned local features into the model, we refer to [116, 117].

In the next subsections, we will see that machine learning based models can further improve the performance.

4.2 | Patch normalizing flow regularizer

Another successful approach models the patch distribution using *normalizing flows* (NFs) [3]. NFs are invertible differentiable mappings. Currently, there are two main structures that achieve invertibility of a neural network, namely invertible residual networks [13] and directly invertible networks [7, 24]. For the patchNR, the directly invertible networks are of interest. The invertibility is ensured by the special network structure which in the simplest case consists of a concatenation of K invertible, differentiable mappings $T_{\theta_k} : \mathbb{R}^d \to \mathbb{R}^d$ (and some permutation matrices which are skipped for simplicity)

$$T_\theta = T_{\theta_K} \circ \cdots \circ T_{\theta_1}.$$

The invertibility is ensured by a special splitting structure, namely for $d_1 + d_2 = d$, we set

$$T_{\theta_k}(z_1, z_2) = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} := \begin{pmatrix} z_1 \\ z_2 \odot \exp\Big(s_{\theta_{k_1}}(z_1)\Big) + t_{\theta_{k_2}}(z_1) \end{pmatrix}, \quad z_i, x_i \in \mathbb{R}^{d_i}, \; i = 1, 2, \tag{29}$$

where $s_{\theta_{k_1}}, t_{\theta_{k_2}}$ are arbitrary neural networks and $\odot$ denotes the component-wise multiplication. Then the inverse of each of the K blocks can be simply computed by

$$T_{\theta_k}^{-1}(x_1, x_2) = \begin{pmatrix} z_1 \\ z_2 \end{pmatrix} = \begin{pmatrix} x_1 \\ \Big(x_2 - t_{\theta_{k_2}}(x_1)\Big) \odot \exp\Big(-s_{\theta_{k_1}}(x_1)\Big) \end{pmatrix}. \tag{30}$$

This is the simplest *Real NVP* network architecture [24]. A more sophisticated one is given in [7]. Now the idea is to approximate our unknown patch distribution Q on $\mathbb{R}^d$, $d = p^2$, using the push-forward by T_θ of a measure P_Z, where it is easy to sample from as, for example, the d-dimensional standard normal distribution $Z \sim \mathcal{N}(0, I_d)$. Our goal becomes

$$Q \approx (T_\theta)_\# P_Z = Q_\theta.$$

The NF between (samples of) the standard normal distribution in $\mathbb{R}^{36}$ and the distribution of material image patches is illustrated in Figure 5. Let us take the KL approach to find the parameters of $q_\theta = q_{(T_\theta)_\# P_Z}$, that is,

$$\mathrm{KL}(Q, (T_\theta)_\# P_Z) = \int_{\mathbb{R}^d} \log\left(\frac{q(x)}{q_{(T_\theta)_\# P_Z}(x)}\right) q(x) \, \mathrm{d}x \tag{31}$$

$$= \underbrace{\int_{\mathbb{R}^d} \log(q(x)) q(x) \, \mathrm{d}x}_{\text{const}} - \int_{\mathbb{R}^d} \log\Big(q_{(T_\theta)\,_\# P_Z}(x)\Big) q(x) \, \mathrm{d}x. \tag{32}$$

Using the transformation formula (27), we obtain (up to a constant)

$$\begin{aligned} \mathrm{KL}(Q, (T_\theta)_\# P_Z) &\propto -\int_{\mathbb{R}^d} \log\Big(p_Z\big((T_\theta)^{-1}(x)\big) \; |\det \nabla T_\theta^{-1}(x)|\Big) q(x) \, \mathrm{d}x \\ &= -\mathbb{E}_{x \sim Q}\Big[\log p_Z\big(T_\theta^{-1}(x)\big) + \log\big(|\det \nabla T_\theta^{-1}(x)|\big)\Big] \end{aligned}$$

and since P_Z is standard normally distributed further

$$\mathrm{KL}(Q, (T_\theta)_\# P_Z) \propto \mathbb{E}_{x \sim Q}\Big[\frac{1}{2} \big\|T_\theta^{-1}(x)\big\|^2 - \log\big(\big|\det \nabla T_\theta^{-1}(x)\big|\big)\Big]. \tag{33}$$

Taking the empirical expectation provides us with the ML loss function

$$\mathcal{L}(\theta) = \sum_{j=1}^{n} \sum_{i=1}^{N} \frac{1}{2} \|T_\theta^{-1}\big(\mathcal{P}_i(x_j)\big)\|^2 - \log\big(|\det \nabla T_\theta^{-1}\big(\mathcal{P}_i(x_j)\big)|\big).$$

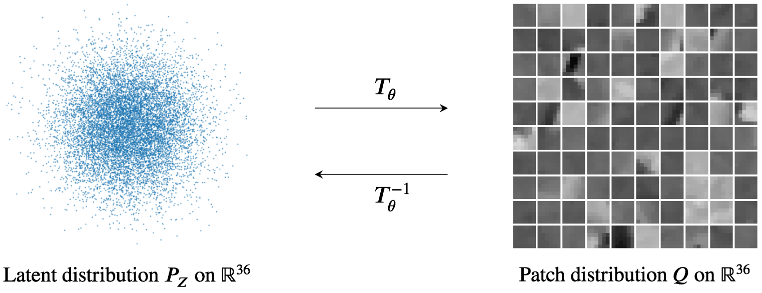

T_θ
T_θ^{-1}
Latent distribution P_Z on $\mathbb{R}^{36}$
Patch distribution Q on $\mathbb{R}^{36}$

FIGURE 5 NF between (samples of) standard normal distribution (projection on $\mathbb{R}^2$) and distribution of 6×6 patches.

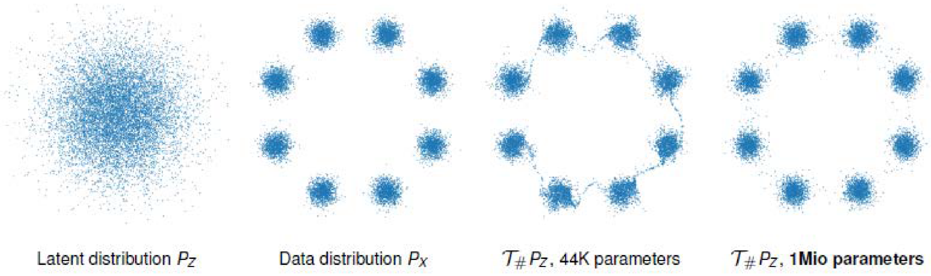


FIGURE 6 NFs between (samples of) 2D standard Gaussian distribution and multimodal distribution. A good approximation is only possible with the rightmost NF which has a higher number of parameters and here also a higher Lipschitz constant.

To minimize this function we use a stochastic gradient descent algorithm, where the special structure (29) of the network can be utilized for the gradient computations. Once good network parameters $\hat{\theta}$ are found, we introduce in (10) the *patchNR*

$$\mathcal{R}(x) = \mathrm{patchNR}(x) := \frac{1}{N}\sum_{i=1}^{N} \frac{1}{2}\|T_{\hat{\theta}}^{-1}(\mathscr{P}_i(x))\|^2 - \log\Big(|\det \nabla T_{\hat{\theta}}^{-1}(\mathscr{P}_i(x))|\Big). \tag{34}$$

Remark 2. By the relation (9) the patchNR defines a prior distribution $p_X(x) = C_\beta \exp(-\beta \mathrm{patchNR}(x))$ on the space of images. The integrability of the function p_X is shown in [3, Prop. 6].

Remark 3. The KL divergence of measures is neither symmetric nor fulfills a triangular inequality. Concerning symmetry, the setting $\mathrm{KL}(Q, Q_\theta)$ is called forward KL. Changing the order of the measures gives the *backward* (or reverse) KL in $\mathrm{KL}(Q_\theta, Q)$. These settings have different properties as being mode seeking or mode covering and the loss functions rely on different data inputs, see [46]. There are also mixed variants $\alpha\mathrm{KL}(Q, Q_\theta) + (1-\alpha)\mathrm{KL}(Q_\theta, Q)$, $\alpha \in (0, 1)$ as well as the Jensen–Shannon divergence [38].

Unfortunately, NFs mapping unimodal to multimodal distributions suffer from exploding Lipschitz constants and are therefore sensitive to adversarial attacks [14, 48, 57, 92]. This is demonstrated in Figure 6. A remedy is the use of GMMs for latent distribution [48] or of *stochastic NFs* [46, 47, 114].

4.3 | Adversarial local regularizers

The adversarial local regularizer (ALR) proposed by Prost et al. [85] makes use of a discriminative model. Originally, the ALR was not formulated with a loss of the form (20), but by an adversarial approach similar to Wasserstein generative adversarial networks (WGANs) [9]. The basic idea goes back to Lunz et al. [71], who suggested learning regularizers through corrupted data. More precisely, the regularizer is a neural network trained to discriminate between the distribution of ground truth images and the distribution of unregularized reconstructions. The ALR is based on the same idea, but it operates on patches instead of whole images. Here a *discriminator* D_θ between unpaired samples from the original patch distribution Q and a degraded one, say $\tilde{Q}$, is trained using the Wasserstein-1 distance. Conveniently, the Wasserstein-1 distance has the dual formulation

$$W_1(Q,\tilde{Q}) = \sup_{f \in Lip_1} \left\{ \mathbb{E}_{x \sim Q}[f(x)] - \mathbb{E}_{\tilde{x} \sim \tilde{Q}} \left[f(\tilde{x}) \right] \right\},$$

where Lip_1 denotes the set of all Lipschitz continuous functions on $\mathbb{R}^d$ with Lipschitz constant not larger than 1. A maximizing function is called optimal *Kantorovich potential* and can be considered as a good separation between the two distributions. Unfortunately, obtaining such a potential is computationally intractable. Nevertheless, it can be approximated by functions from a parameterized family $\mathcal{F}$ as, for example, neural networks with a fixed architecture

$$\arg\max_{\mathrm{D}_\theta \in \mathcal{F} \cap \mathrm{Lip}_1} \left\{ \mathbb{E}_{x \sim Q}[\mathrm{D}_\theta(x)] - \mathbb{E}_{\tilde{x} \sim \tilde{Q}}[\mathrm{D}_\theta(\tilde{x})] \right\}. \tag{35}$$

One possibility to relax the Lipschitz condition is the addition of a gradient penalty, see [43], to arrive at

$$\hat{\theta} = \arg\max_{\theta} \left\{ \mathbb{E}_{x \sim Q}[\mathrm{D}_\theta(x)] - \mathbb{E}_{\tilde{x} \sim \tilde{Q}}[\mathrm{D}_\theta(\tilde{x})] - \lambda \mathbb{E}_{x \sim Q_\alpha} \left[(\|\nabla \mathrm{D}_\theta(x)\| - 1)^2 \right] \right\}, \quad \lambda > 0, \tag{36}$$

where $X_\alpha \sim Q_\alpha$ fulfills $X_\alpha = \alpha X + (1-\alpha)\tilde{X}$ for $X \sim Q$ and $\tilde{X} \sim \tilde{Q}$ and α is uniformly distributed in $[0,1]$. This can be solved using a stochastic gradient descent algorithm. Finally, to solve our inverse problem, we can use the ALR

$$\mathcal{R}(x) = ALR(x) := \frac{1}{N} \sum_{i=1}^{N} \mathrm{D}_\theta(\mathscr{P}_i(x)).$$

This parameter estimation is different from the ML estimation of the previous two models, since it employs a discriminative approach.

Remark 4. The original GAN architecture utilizes the Jensen–Shannon divergence [38] and can be replaced with the (forward) KL divergence (22). This would lead to some form of maximum likelihood estimation in a discriminative setting. Furthermore, GAN architectures within an explicit maximum likelihood framework exist [42]. On closer inspection, however, this still takes a similar form as the EPLL or the patchNR. We assign a value to each patch in an image and sum over the set of resulting values. Moreover, higher values are assigned to patches that are more likely to stem from the true patch distribution. As a result, we could interpret the assigned patch score as the log-likelihood of a given patch.

5 | DIVERGENCES BETWEEN EMPIRICAL PATCH MEASURES

In the previous section, we have constructed patch-based regularizers using sums over all patches within an ML approach. This calls for independently drawn patches from the underlying distribution, an assumption that does not hold true, for example, for overlapping patches. In particular, the same value may be assigned for an image which is a combination of very likely and very unlikely patches. This makes it desirable to address the patch distribution as a whole by assigning empirical measures to the patches as in (24). In the following, we will consider three different "distances" between these empirical measures, namely the Wasserstein-2 distance, the regularized Wasserstein-2 distance, and an unbalanced variant.

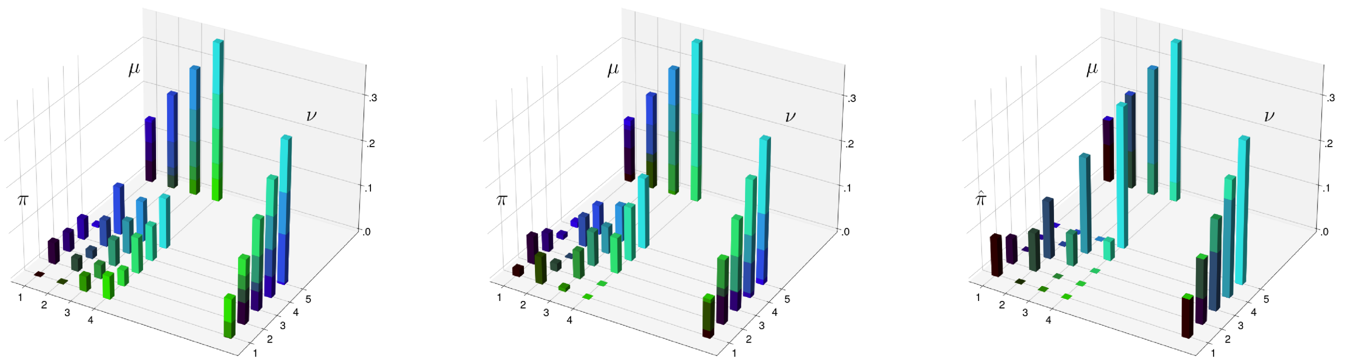
5.1 | Wasserstein patch prior

To keep the notation simple, let us rewrite the empirical measures in (24) as

$$\nu = \frac{1}{nN} \sum_{j=1}^{n} \sum_{i=1}^{N} \delta_{\mathscr{P}_i(x_j)} = \frac{1}{M} \sum_{k=1}^{M} \delta_{y_k} \quad \text{and} \quad \mu_x = \frac{1}{N} \sum_{i=1}^{N} \delta_{\mathscr{P}_i(x)} = \frac{1}{N} \sum_{i=1}^{N} \delta_{x_i}. \tag{37}$$

Then the admissible plans in the Wasserstein-2 distance (26) have the form

$$\pi = \sum_{i=1}^{N} \sum_{k=1}^{M} \pi_{i,k} \delta_{i,k}, \qquad \sum_{i=1}^{N} \pi_{i,k} = \frac{1}{M}, \ \sum_{k=1}^{M} \pi_{i,k} = \frac{1}{N}, \ i = 1, \dots, N, k = 1, \dots, M.$$



Cost: 3.335

Cost: 2.145

Cost: $\mathrm{W}_2^2(\mu, \nu) = 0.714$

FIGURE 7 Admissible couplings between $\mu = \sum_{i=1}^{4} \mu_i \delta_i = \frac{2}{14}\delta_1 + \frac{3}{14}\delta_2 + \frac{4}{14}\delta_3 + \frac{5}{14}\delta_4$ and $\nu = \sum_{j=1}^{5} \nu_j \delta_j = \frac{3}{35}\delta_1 + \frac{5}{35}\delta_2 + \frac{7}{35}\delta_3 + \frac{9}{35}\delta_4 + \frac{11}{35}\delta_5$ (weights are visualized as bars). Any admissible coupling takes the form $\pi = \sum_{i,j=1}^{4,5} \pi_{i,j}\delta_{(i,j)}$, where $\sum_{i=1}^{4} \pi_{i,j} = \nu_j$ and $\sum_{j=1}^{5} \pi_{i,j} = \mu_i$. The weights of three admissible couplings are visualized. The Wasserstein-2 distance is characterized by those π which minimize the costs $= \sum_{i,j=1}^{4,5} \pi_{i,j}(i-j)^2$. The squared Wasserstein-2 distance has the sparsest coupling.

Obviously, they are determined by the weight matrix $\boldsymbol{\pi} := (\pi_{i,k})_{i,k=1}^{N,M}$. Then, with the cost matrix $C := (\|x_i - y_k\|^2)_{i,k=1}^{N,M}$, the Wasserstein-2 distance becomes

$$W_2^2(\mu_x, \nu) = \min_{\boldsymbol{\pi} \in \Pi} \langle C, \boldsymbol{\pi} \rangle, \quad \Pi = \left\{ \boldsymbol{\pi} \in \mathbb{R}_{\geq 0}^{N,M} \,:\, \mathbb{1}_N^{\mathrm{T}} \boldsymbol{\pi} = \frac{1}{M}\mathbb{1}_M, \boldsymbol{\pi}\mathbb{1}_M = \frac{1}{N}\mathbb{1}_N \right\}. \tag{38}$$

Here $\mathbb{1}_M \in \mathbb{R}^M$ denotes the vector with all entries one. An example of the Wasserstein-2 distance for two discrete measures is given in Figure 7. The dual formulation of the linear optimization problem (38) reads as

$$W_2^2(\mu_x, \nu) = \max_{\phi(x_i)+\psi_k \leq c_{i,k}} \frac{1}{N}\sum_{i=1}^{N} \phi(x_i) + \frac{1}{M}\sum_{k=1}^{M} \psi_k \tag{39}$$

$$= \max_{\psi \in \mathbb{R}^M} \frac{1}{N}\sum_{i=1}^{N} \psi^c(x_i) + \frac{1}{M}\sum_{k=1}^{M} \psi_k \tag{40}$$

with the *c-conjugate function*

$$\psi^c(x_i) := \min_k \left\{ \|x_i - y_k\|^2 - \psi_k \right\}, \tag{41}$$

see [93]. The maximization problem (39) is concave, and for large-scale problems, a gradient ascent algorithm as in [34] can be used to find a global maximizer $\hat{\psi}$. As in [5, 51], the optimal vector $\hat{\psi}$ allows for the computation of the gradient of

$$\mathcal{R}(x) = \mathrm{WPP(x)} := W_2^2(\mu_x, \nu) \tag{42}$$

in our inverse problem (10). More precisely, with the minimizer $\sigma(i) \in \arg\min_k \{\|x_i - y_k\|^p - \hat{\psi}_k\}$ in (41) we obtain

$$W_2^2(\mu_x, \nu) = \frac{1}{N}\sum_{i=1}^{N} \|x_i - y_{\sigma(i)}\|^2 - \hat{\psi}_{\sigma(i)} + \frac{1}{M}\sum_{k=1}^{M} \hat{\psi}_k,$$

so that if the gradient with regard to the support point x_i of μ_x exists, it reads as

$$\nabla_{x_i} W_2^2(\mu_x, \nu) = \frac{1}{N}\nabla_{x_i}\|x_i - y_{\sigma(i)}\|_2^2 = \frac{2}{N}(x_i - y_{\sigma(i)}). \tag{43}$$

Remark 5. By the relation (9) the WPP defines a prior distribution $p_X(x) = C_\beta \exp(-\beta \mathrm{WPP}(x))$ on the space of images. The integrability of the function p_X is shown in [5, Prop. 4.1].

Remark 6. Wasserstein patch priors were originally introduced by Gutierrez et al. [44] and Houdard et al. [55] for texture generation, where a direct minimization of the regularizer without a data fidelity term was used. Their use was adopted for regularization in inverse problems by Hertrich et al. [51]. Note that this stands in contrast to the previous EPLL-based regularizers, where a direct minimization of these regularizers would result in the synthesis of images with almost equally likely patches. In practice, this would lead to single-color images.

5.2 Sinkhorn patch prior

To lower the computational burden in the WPP approach, a combination of the Wasserstein distance with the KL of the coupling and the product measure $\mu_x \otimes \nu$ can be used

$$W_{2,\varepsilon}^2(\mu_x,\nu) = \inf_{\pi\in\Pi(\mu_x,\nu)} \int_{\mathbb{R}^d\times\mathbb{R}^d} \|x-y\|^2 \, d\pi(x,y) + \varepsilon \mathrm{KL}(\pi,\mu_x\otimes\nu) \tag{44}$$

$$= \inf_{\boldsymbol{\pi}\in\Pi(\mu_x,\nu)} \langle C,\boldsymbol{\pi}\rangle + \varepsilon \sum_{i,k=1}^{N,M} \pi_{i,k} \log\left(MN\pi_{i,k}\right). \tag{45}$$

In Figure 8 we give an example of $W_{2,\varepsilon}^2$ for different choices of ε and the same discrete measures as in Figure 7. The dual formulation reads as

$$W_{2,\varepsilon}^2(\mu_x,\nu) = \max_{\phi\in\mathbb{R}^N,\psi\in\mathbb{R}^M} \frac{1}{N}\sum_{i=1}^N \phi_i + \frac{1}{M}\sum_{k=1}^M \psi_k - \frac{\varepsilon}{MN}\sum_{i=1}^N\sum_{k=1}^M \exp\left(\frac{\phi_i+\psi_k-\|x_i-y_k\|^2}{\varepsilon}\right) + \varepsilon. \tag{46}$$

This problem can be efficiently solved using the *Sinkhorn algorithm* which employs a fixed-point iteration. To this end, fix $\psi^{(r)}$, respectively, $\phi^{(r)}$ and set the gradient with respect to the other variable in (46) to zero. This results in the iterations

$$\phi_i^{(r+1)} = -\varepsilon \log\left(\sum_{k=1}^M \exp\left(\frac{\psi_k^{(r)}-\|x_i-y_k\|^2}{\varepsilon}\right)\right) + \varepsilon \log M,$$
$$\psi_k^{(r+1)} = -\varepsilon \log\left(\sum_{i=1}^N \exp\left(\frac{\phi_i^{(r)}-\|x_i-y_k\|^2}{\varepsilon}\right)\right) + \varepsilon \log N,$$

which converge linearly to the fixed points $\hat{\phi}$ and $\hat{\psi}$, see, for example, [29]. Then, noting that by construction of $\hat{\phi}$ and $\hat{\psi}$ we have

$$-\frac{\varepsilon}{MN}\sum_{i=1}^N\sum_{k=1}^M \exp\left(\frac{\hat{\phi}_i+\hat{\psi}_k-\|x_i-y_k\|^2}{\varepsilon}\right) + \varepsilon = 0, \tag{47}$$

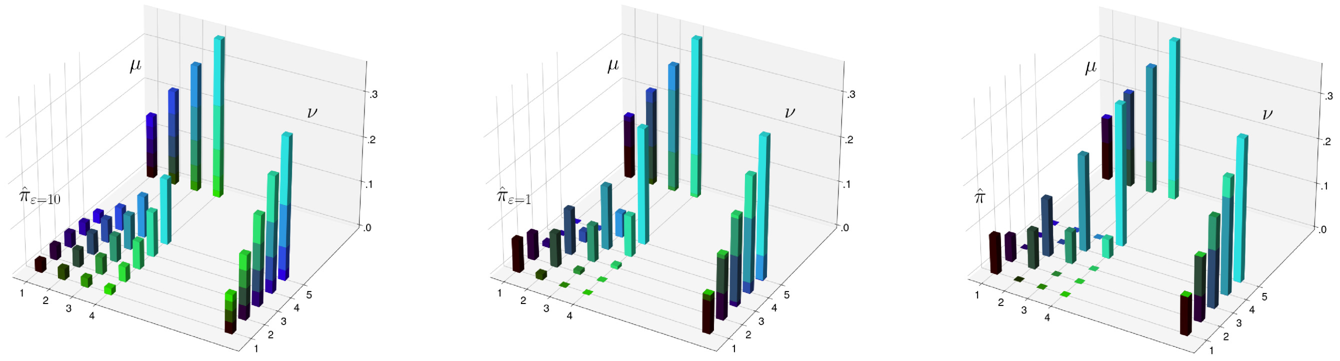
the regularized Wasserstein distance becomes

$$W_{2,\varepsilon}^2(\mu_x,\nu) = -\frac{\varepsilon}{N}\left(\sum_{i=1}^N \log\left(\sum_{k=1}^M \exp\left(\frac{\hat{\psi}_k-\|x_i-y_k\|^2}{\varepsilon}\right)\right) - \log M\right) + \frac{1}{M}\sum_{k=1}^M \hat{\psi}_k, \tag{48}$$

which is differentiable with respect to the support points and the gradient is given by

$$\nabla_{x_i} W_{2,\varepsilon}^2(\mu_x,\nu) = \frac{2}{N}\left(\sum_{k=1}^M \exp\left(\frac{\hat{\psi}_k-\|x_i-y_k\|^2}{\varepsilon}\right)\right)^{-1} \sum_{k=1}^M \exp\left(\frac{\hat{\psi}_k-\|x_i-y_k\|^2}{\varepsilon}\right)(x_i-y_k). \tag{49}$$

If the Wasserstein gradient from (43) exists, it is recovered for $\varepsilon \to 0$. Computation of the gradient can, for example, be achieved by means of algorithmic differentiation through the Sinkhorn iterations or on the basis of the optimal dual



$\mathrm{W}^2_{2,10}(\mu,\nu) = 2.935$ $\mathrm{W}^2_{2,1}(\mu,\nu) = 1.544$ $\mathrm{W}^2_{2}(\mu,\nu) = 0.714$

FIGURE 8 Optimal couplings of $W_{2,\varepsilon}$ for $\varepsilon = 10, 1, 0$ and the measures from Figure 7. With decreasing ε the coupling matrices become sparser.

potentials through the Sinkhorn algorithm. Finally, we can use the Sinkhorn patch prior (WPP$_\varepsilon$) first used in [73] as a regularizer in our inverse problem

$$\mathcal{R}(x) = WPP_\varepsilon(x) := W^2_{2,\varepsilon}(\mu_x, \nu).$$

Remark 7. By the relation (9) the Sinkhorn regularizer defines a prior distribution $p_X(x) = C_\beta \exp(-\beta \mathrm{WPP}_\varepsilon(x))$ on the space of images. The integrability of the function p_X follows from the integrability of the WPP [5, Prop. 4.1] and the relation $\mathrm{WPP}_\varepsilon(x) \geq \mathrm{WPP}(x)$. This can be seen immediately since $\mathrm{KL}(\pi, \mu_x \otimes \nu) \geq 0$.

Remark 8. The regularized Wasserstein-2 distance is no longer a distance. It does not fulfill the triangular inequality and is moreover biased, that is, $W_{2,\varepsilon}(\mu,\nu)$ does not take its smallest value if and only if $\mu = \nu$. As a remedy, the debiased regularized Wasserstein distance or Sinkhorn divergence

$$S^2_{2,\varepsilon}(\mu,\nu) = W^2_{2,\varepsilon,}(\mu,\nu) - \frac{1}{2} W^2_{2,\varepsilon}(\mu,\mu) - \frac{1}{2} W^2_{\varepsilon}(\nu,\nu)$$

can be used, which is now indeed a statistical distance. Computation with the Sinkhorn divergence is similar to above so it can be used as a regularizer as well. For more information see [35, 76].

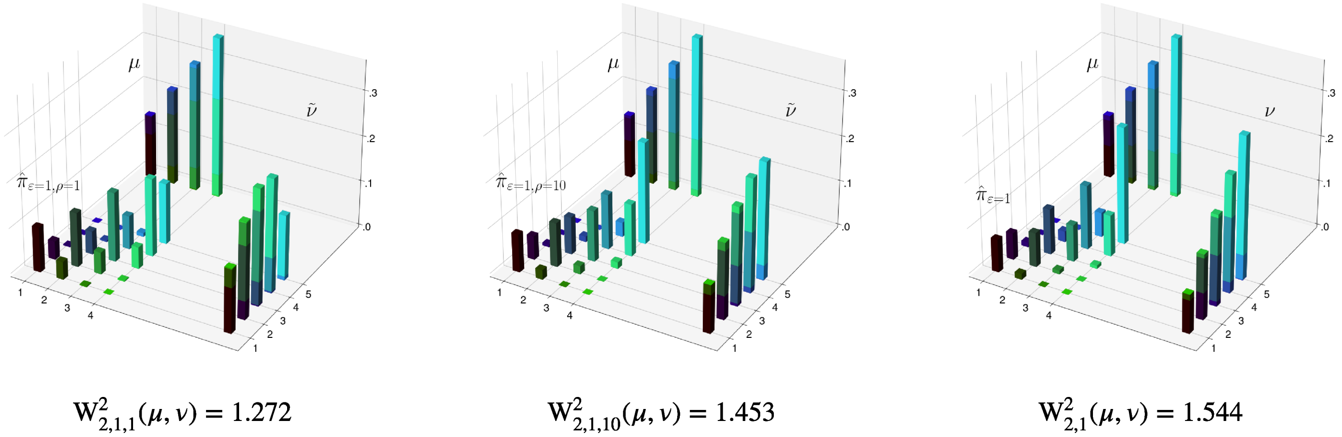
5.3 | Semi-unbalanced Sinkhorn patch prior

The optimal transport framework for regularizing the patch distribution was extended by Mignon et al. [73] to the *semi-unbalanced case*, where the marginal of the coupling only approximates the target distribution for

$$\begin{aligned} W^2_{2,\varepsilon,\rho}(\mu_x,\nu) &= \inf_{\substack{\pi \in \mathcal{P}(\mathbb{R}^d \times \mathbb{R}^d) \\ (\mathrm{proj}_1)_\# \pi = \mu_x}} \int\limits_{\mathbb{R}^d \times \mathbb{R}^d} \|x-y\|^2 \, \mathrm{d}\pi(x,y) + \varepsilon \mathrm{KL}(\pi, \mu_x \otimes \nu) + \rho \mathrm{KL}((\mathrm{proj}_2)_\# \pi, \nu) \\ &= \inf_{\substack{\pi \in \mathbb{R}^{N,M} \\ \pi \mathbb{1}_M = \frac{1}{N} \mathbb{1}_N}} \langle C, \boldsymbol{\pi} \rangle + \varepsilon \sum_{i,k=1}^{N,M} \pi_{i,k} \log\big(MN\pi_{i,k}\big) + \rho \sum_{k=1}^{M} \big(\mathbb{1}_N^{\mathrm{T}} \boldsymbol{\pi}\big)_k \log\Big(M\big(\mathbb{1}_N^{\mathrm{T}} \boldsymbol{\pi}\big)_k\Big). \end{aligned}$$

In this setting, probability mass can be added to ν or removed from ν. This behavior is controlled by the parameter ρ and leads to a decreased sensitivity with regard to isolated areas in the second distribution. An example of $W^2_{2,\varepsilon,\rho}$ for different choices of ρ and the same measures as in Figure 7 is given in Figure 9. The dual formulation becomes

$$W^2_{2,\varepsilon,\rho}(\mu_x,\nu) = \max_{\phi \in \mathbb{R}^M, \psi \in \mathbb{R}^M} \frac{1}{N} \sum_{i=1}^{N} \phi_i + \frac{1}{M} \sum_{k=1}^{M} \rho \left(\exp\left(\frac{\psi_k}{\rho} \right) - 1 \right) - \frac{\varepsilon}{MN} \sum_{i=1}^{N} \sum_{k=1}^{M} \exp\left(\frac{\phi_i + \psi_k - \|x_i - y_k\|^2}{\varepsilon} \right) + \varepsilon,$$



$\mathrm{W}^2_{2,1,1}(\mu,\nu) = 1.272$ $\mathrm{W}^2_{2,1,10}(\mu,\nu) = 1.453$ $\mathrm{W}^2_{2,1}(\mu,\nu) = 1.544$

FIGURE 9 Optimal couplings of $W_{2,\varepsilon,\rho}$ for $\varepsilon = 1$ and $\rho = 1, 10, \infty$ and the measures from Figure 7. The marginal $\tilde{\nu}$ (left, middle) of the coupling matrix is only an approximation of the original measure ν (right). Note the gradually increased probability mass placed on 5 for $\tilde{\nu}$. By increasing the balancing parameter, we move the approximation $\tilde{\nu}$ towards ν (left to middle).

see, for example, [73]. This maximization problem can be solved by the following adapted Sinkhorn iterations

$$\phi_i^{(r+1)} = -\varepsilon \log\left(\sum_{k=1}^{M} \exp\left(\frac{\psi_k^{(r)} - \|x_i - y_k\|^2}{\varepsilon}\right)\right) + \varepsilon \log M,$$
$$\psi_k^{(r+1)} = -\frac{\varepsilon\rho}{\rho+\varepsilon} \log\left(\sum_{i=1}^{N} \exp\left(\frac{\phi_i^{(r)} - \|x_i - y_k\|^2}{\varepsilon}\right)\right) + \varepsilon \log N.$$

Note that the fixed point $\hat{\phi}$ fulfills the fixed point condition from Section 5.2 and consequently $\hat{\phi}$ and $\hat{\psi}$ fulfill (47). The semi-unbalanced regularized Wasserstein distance becomes

$$W^2_{2,\varepsilon,\rho}(\mu_x,\nu) = -\frac{\varepsilon}{N}\left(\sum_{i=1}^{N} \log\left(\sum_{k=1}^{M} \exp\left(\frac{\hat{\psi}_k - \|x_i - y_k\|^2}{\varepsilon}\right)\right) - \log M\right) + \frac{1}{M}\sum_{k=1}^{M} \rho\left(\exp\left(\frac{\hat{\psi}_k}{\rho}\right) - 1\right).$$

This expression equals the expression (48) up to the second term, which does not depend on the support points. As a result, the gradient takes the same form as in (49), but for a $\hat{\psi}$ depending on ρ. For $\rho \to \infty$ we recover the balanced formulation and hence the gradient from (49). Finally, we can use a semi-unbalanced Sinkhorn patch prior ($\mathrm{WPP}_{\varepsilon,\rho}$) defined as

$$\mathcal{R}(x) = WPP_{\varepsilon,\rho}(x) := W^2_{2,\varepsilon,\rho}(\mu_x,\nu).$$

This was proposed as an extension of the WPP in [73].

Remark 9. By the relation (9) the $\mathrm{WPP}_{\varepsilon,\rho}$ defines a prior distribution $p_X(x) = C_\beta \exp(-\beta \mathrm{WPP}_{\varepsilon,\rho}(x))$ on the space of images. This can be seen as follows: Using the auxiliary variable $\tilde{\nu} = (\mathrm{proj}_2)_\#\pi$ we rewrite $W^2_{2,\varepsilon,\rho}(\mu_x,\nu)$ by

$$\inf_{\substack{\pi\in\Pi(\mu_x,\tilde{\nu})\\ \mathrm{supp}(\tilde{\nu})\subseteq\mathrm{supp}(\nu)}} W_2^2(\mu_x,\tilde{\nu}) + \varepsilon \mathrm{KL}(\pi, \mu_x \otimes \nu) + \rho \mathrm{KL}(\tilde{\nu},\nu) \geq \inf_{\substack{\pi\in\Pi(\mu_x,\tilde{\nu})\\ \mathrm{supp}(\tilde{\nu})\subseteq\mathrm{supp}(\nu)}} W_2^2(\mu_x,\tilde{\nu}).$$

The constraint $\mathrm{supp}(\tilde{\nu}) \subseteq \mathrm{supp}(\nu)$ is due to the term $\mathrm{KL}(\tilde{\nu},\nu)$ which otherwise would be infinite. Exploiting the discrete structure $\nu = \frac{1}{M}\sum_{k=1}^{M}\delta_{y_k}$, such a measure $\tilde{\nu}$ needs to be of the form $\tilde{\nu} = \sum_{k=1}^{M} a_k \delta_{y_k}$, for $a \in \mathbb{R}^M_{\geq 0}$ with

$\sum_{k=1}^{M} a_k = 1$. The dual formulation (39) yields

$$\inf_{\substack{\pi\in\Pi(\mu_x,\tilde{\nu})\\ \mathrm{supp}(\tilde{\nu})\subseteq\mathrm{supp}(\nu)}} W_2^2(\mu_x,\tilde{\nu}) = \inf_{\substack{a\in\mathbb{R}^M_{\geq 0}\\ \sum_{k=1}^{M} a_k=1}} \left(\max_{\psi(a)\in\mathbb{R}^M} \frac{1}{N}\sum_{i=1}^{N}\psi(a)^c(x_i) + \sum_{k=1}^{M} a_k\psi(a)_k \right) \geq \frac{1}{N}\sum_{i=1}^{N}\psi_0^c(x_i),$$

where the last inequality follows from inserting $\psi(a) = \psi_0 = 0$ for all $a \in \mathbb{R}^M$. Now, the statement follows from the proof of [5, Prop. 4.1].

Remark 10. By construction, the semi-unbalanced regularized Wasserstein distance is not symmetric anymore. Moreover, it is again biased. Similarly, as for the balanced case, the semi-unbalanced regularized Wasserstein distance can be transformed into a (non-symmetric) semi-unbalanced Sinkhorn divergence

$$S^2_{2,\varepsilon,\rho}(\mu,\nu) = W^2_{2,\varepsilon,\rho}(\mu,\nu) - \frac{1}{2}W^2_{2,\varepsilon}(\mu,\mu) - \frac{1}{2}W^2_{2,\varepsilon,\rho,\rho}(\nu,\nu)$$

with the fully unbalanced regularized Wasserstein distance

$$W^2_{2,\varepsilon,\rho,\rho}(\mu,\nu) = \inf_{\pi\in\mathcal{M}^+(\mathbb{R}^d\times\mathbb{R}^d)} \int\limits_{\mathbb{R}^d\times\mathbb{R}^d} \|x-y\|^2\,\mathrm{d}\pi(x,y) + \varepsilon\mathrm{KL}(\pi,\mu_x\otimes\nu) + \rho\mathrm{KL}((\mathrm{proj}_2)_\#\pi,\nu) + \rho\mathrm{KL}((\mathrm{proj}_1)_\#\pi,\mu).$$

Here, $\mathcal{M}^+(\mathbb{R}^d\times\mathbb{R}^d)$ denotes the set of positive measures on $\mathbb{R}^d\times\mathbb{R}^d$. For more information, see [95].

6 | UNCERTAINTY QUANTIFICATION VIA POSTERIOR SAMPLING

In contrast to the MAP approaches, which just give point estimates for the most likely solution of the inverse problem, see Paragraph 1 of Section 2, we want to approximate the whole posterior measure $P_{X|Y=y}$ now. More precisely, we intend to sample from the approximate posterior to get multiple possible reconstructions of the inverse problem and to quantify the uncertainty in our reconstruction. By Bayes' law and relation (9), we know that

$$p_{X|Y=y}(x) \propto p_{Y|X=x}(y)p_X(x), \quad p_X(x) = C_\beta \exp(-\beta\mathcal{R}(x)).$$

While the likelihood $p_{Y|X=x}$ is determined by the noise model and the forward operator, the idea is now to choose a prior from the previous sections, that is,

$$\mathcal{R} \in \{\mathrm{EPLL}, \mathrm{patchNR}, \mathrm{ALR}, \mathrm{WPP}, \mathrm{WPP}_\varepsilon, \mathrm{WPP}_{\varepsilon,\rho}\}, \tag{50}$$

By the Remarks 1,2,5,7 and 9 we have ensured that the corresponding functions p_X are indeed integrable on the image space $\mathbb{R}^d$, except for ALR, where this is probably not the case. Nevertheless, we will use ALR in our computations even without the theoretical foundation. Techniques to enforce the integrability of a given regularizer by utilizing a projection onto a compact set, for example, $[0,1]^d$, exist in the literature [18, 61].

Even if the density of a distribution is known you can in general not sample from this distribution, except for the uniform and the Gaussian distribution. Established methods for posterior sampling are Markov chain Monte Carlo (MCMC) methods such as Gibbs sampling [88]. We want to focus on Langevin Monte Carlo methods [75, 89, 112], which have shown good performance for image applications and come with theoretical guarantees [18, 61]. In particular, in [30] the EPLL was used in combination with Gibbs sampling for posterior reconstruction of natural images, and in [18] the patchNR was used in combination with Langevin sampling for posterior reconstruction in limited-angle CT.

Consider the overdamped Langevin stochastic differential equation (SDE)

$$\mathrm{d}X_t = \nabla\log p_{X|Y=y}(X_t)\mathrm{d}t + \sqrt{2}\mathrm{d}B_t, \tag{51}$$

where B_t is the d-dimensional Brownian motion. If $p_{X|Y=y}$ is proper, smooth and $x \mapsto \nabla\log p_{X|Y=y}(x)$ is Lipschitz continuous, then Roberts and Tweedie [89] have shown that, for any initial starting point, the SDE (51) has a unique strong solution and $p_{X|Y=y}$ is the unique stationary density. For a discrete time approximation, the *Euler-Maruyama discretization* with step size δ leads to the *unadjusted Langevin algorithm* (ULA)

$$X_{k+1} = X_k + \delta \nabla \log p_{X|Y=y}(X_k) + \sqrt{2\delta} Z_{k+1} \tag{52}$$

$$= X_k + \delta \nabla \log p_{Y|X=X_k}(y) + \delta \nabla \log p_X(X_k) + \sqrt{2\delta} Z_{k+1}, \tag{53}$$

where $Z_k \sim \mathcal{N}(0, I)$, $k \in \mathbb{N}$. The step size δ provides control between accuracy and convergence speed. The error made due to the discretization step in (52) can be asymptotically removed by a Metropolis-Hastings correction step [89]. The corresponding *Metropolis-adjusted Langevin algorithm* (MALA) comes with additional computational cost and will not be considered here. Now using an approximation (50) for the prior, we get up to an additive constant

$$X_{k+1} = X_k + \delta \nabla \log p_{Y|X=X_k}(y) - \delta\beta \nabla \log \mathcal{R}(X_k) + \sqrt{2\delta} Z_{k+1}. \tag{54}$$

In Section 7.5, we will use this iteration for posterior sampling in image inpainting.

6.1 | Other methods for sampling from the posterior distribution

Alternatively to MCMC methods, posterior sampling can be done by conditional neural networks. While conditional variational auto-encoders (VAEs) [59, 68, 102] approximate the posterior distribution by learning conditional stochastic encoder and decoder networks, conditional generative adversarial networks (GANs) [2, 9, 38, 70] learn a conditional generator via adversarial training. Conditional diffusion models [12, 103, 104] map the posterior distribution to an approximate Gaussian distribution and reverse the noising process for sampling from the posterior distribution. For the reverse noising process, the conditional model needs to approximate the *score* $\nabla_x \log p_{X|Y=y}$. Conditional normalizing flows [5, 6, 8, 113] aim to approximate the posterior distribution using diffeomorphisms. In particular, in [5] the WPP was used as the prior distribution for training the normalizing flow with the backward KL. Recently, gradient flows of the maximum mean discrepancy and the sliced Wasserstein distance were successfully used for posterior sampling [26, 45].

7 | EXPERIMENTS

In this section, we first use the MAP approach

$$x_{\mathrm{MAP}}(y) \in \underset{x \in \mathbb{R}^d}{\arg\min}\{\mathcal{D}(F(x), y) + \beta \mathcal{R}(x)\}, \quad \beta > 0,$$

with our different regularizers

$$\mathcal{R} \in \{\text{EPLL, patchNR, ALR, WPP, WPP}_\varepsilon, \text{WPP}_{\varepsilon,\rho}\}$$

on 6×6 image patches for solving various inverse problems. Since the data term $\mathcal{D}$ depends on the forward operator and the noise model, we have to describe both for each application. We consider the following problems:

- computed tomography (CT) in a low-dose and a limited-angle setting, where we learn the regularizer from just $n = 6$ “clean” images shown in Figure 10. The transformed images are corrupted by Poisson noise
- super-resolution, where the regularizer is first learned from just $n = 1$ “clean” image and second from the corrupted image. The later setting is known as zero-shot super-resolution. Here we have a Gaussian noise model
- image inpainting from the corrupted image in a noise-free setting

Second, we provide examples for sampling from the posterior in image inpainting and for uncertainty quantification in CT.

The code for all experiments is implemented in PyTorch and is available online. [2] You can also find all hyperparameters in the GitHub repository. In the experiments, we minimize the variational formulation (10) using the Adam optimizer [58]. The presented experimental set-up for super-resolution and computed tomography is closely related to the set-up of

[2] https://github.com/MoePien/PatchbasedRegularizer.

Altekrüger et al. [3]. However, before comparing these approaches, we should give some comments on error measures in image processing.

7.1 Error measures

There does not exist an ultimate measure for the visual quality of images, since this depends heavily on the human visual perception. Nevertheless, there are some frequently used quality measures between the original image $x \in \mathbb{R}^{d_1,d_2}$ and the reconstructed, deteriorated one $\hat{x}$. The *peak signal-to-noise ratio* (PSNR) is defined by

$$PSNR(\hat{x}) = 10 \cdot \log_{10}\left(\frac{d_1 d_2 \max^2(x)}{\|x - \hat{x}\|^2}\right),$$

where $\max(x)$ denotes the highest possible pixel value of an image, for example, 255 for 8 bit representations. Unfortunately, small changes in saturation and brightness of the image have a large impact on the PSNR despite a small impact on the visual quality. An established alternative meant to alleviate this issue is the *structural similarity index* (SSIM) [53]. It is based on a comparison of pixel means and variances of various local windows of the images. Still, this is a rather simple model for human vision and small pixel shifts heavily influence its value, see [87]. Recently, the development of improved similarity metrics has revolved around the importance of low-level features for human visual impressions. Hence, alternative metrics focus on the comparison of extracted image features. Prominent examples include the *Feature Similarity Index* (FSIM) [94, 118] based on hand-crafted features and the *Learned Perceptual Image Patch Similarity* (LPIPS) [119] based on the features learned by a convolutional neural network.

All these different metrics behave very differently for image distortions as visualized in Figure 11. The image in Figure 11B is obtained by corrupting the original image in Figure 11A by 5% salt-and-pepper noise. The other images were generated with a gradient descent algorithm starting in Figure 11B for customized loss functions that penalize one

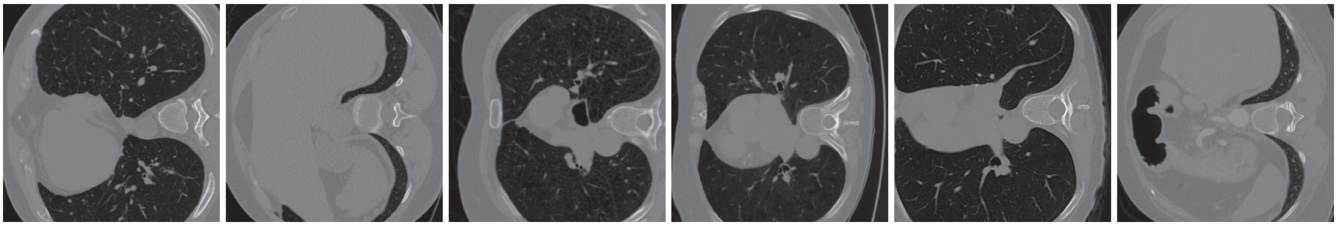
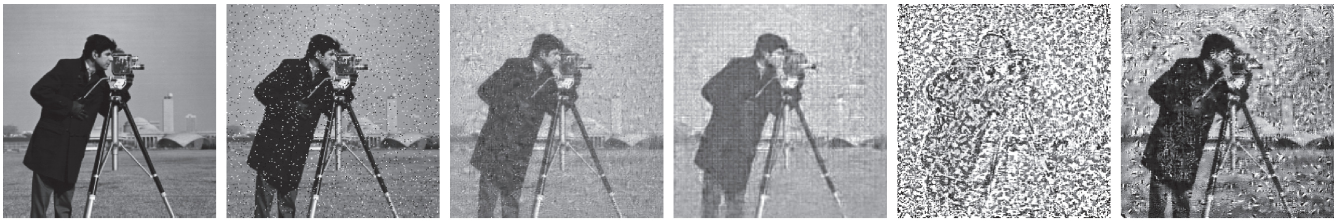


FIGURE 10 CT training images used to train EPLL, patchNR, and ALR as well as for reference patch distribution for WPP, WPP$_\varepsilon$ and WPP$_{\varepsilon,\rho}$.



(A) Original

(B) **PSNR: 20.3**
SSIM: 0.44
LPIPS: 0.55
FSIM: 0.79

(C) **PSNR: 20.3**
SSIM: 0.35
LPIPS: 2.27
FSIM: 0.69

(D) PSNR: 17.6
SSIM: 0.44
LPIPS: 1.14
FSIM: 0.76

(E) PSNR: 3.21
SSIM: 0.15
LPIPS: 0.55
FSIM: 0.36

(F) PSNR: 13.49
SSIM: 0.21
LPIPS: 2.22
FSIM: 0.79

FIGURE 11 Quality measures for different deteriorated images of the original image (A). For PSNR, SSIM, FSIM the largest value is best, for LPIPS the smallest one. The best values for all measures appear in image (B). In the other images, one of the measures is fixed.

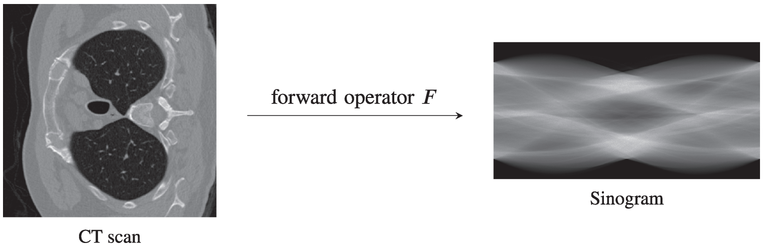


FIGURE 12 Application of the discrete radon transformation.

quality metric and deviation from the initial values for the other quality metric. As a result, the evaluation of reconstructions depends on the chosen metrics, where a single metric may not be suitable for all problems since the requirements differ, for example, for natural and medical images.

7.2 | Computed tomography

In CT, we want to reconstruct a CT scan from a given measurement, which is called a sinogram. We used the LoDoPaB dataset [65][3] for low-dose CT imaging with images of size 362 × 362. The ground truth images are based on scans of the Lung Image Database Consortium and Image Database Resource Initiative [10] and the measurements are simulated. The LoDoPab dataset uses a two-dimensional parallel beam geometry with 513 equidistant detector bins, which results in a linear forward operator F for the discretized Radon transformation. A CT scan and its corresponding sinogram is visualized in Figure 12. The noise model follows a Poisson distribution. Recall that $\text{Pois}(\lambda)$ has probability $p(k|\lambda) = \frac{\lambda^k \exp(-\lambda)}{k!}$ with mean (= variance) λ. More specifically, we assume that the pixels y_i are corrupted independently and we have for each pixel

$$Y = -\frac{1}{\mu}\log\left(\frac{\tilde{Y}}{N_0}\right), \quad \tilde{Y} \sim \text{Pois}(N_0 \exp(-F(x)\mu)),$$

where $N_0 = 4096$ is the mean photon count per detector bin without attenuation and $\mu = 81.35858$ is a normalization constant. Then we obtain pixel-wise

$$\exp(-Y\mu)N_0 = \tilde{Y} \sim \text{Pois}(N_0 \exp(-F(x)\mu)),$$

and consequently for the whole data term

$$\mathcal{D}(F(x), y) = -\log \prod_{i=1}^{\tilde{d}} p(\exp(-y_i\mu)N_0 \,|\, \exp(-F(x)_i\mu)N_0) \tag{55}$$

$$\propto \sum_{i=1}^{\tilde{d}} \exp(-F(x)_i\mu)N_0 + \exp(-y_i\mu)N_0\big(F(x)_i\mu - \log(N_0)\big). \tag{56}$$

For the initialization, we use the Filtered Backprojection (FBP) described by the adjoint Radon transform [86]. We used the ODL implementation [1] with the filter type "Hann" and a frequency scaling of 0.641.

7.2.1 | Low-dose CT

First, we consider a low-dose CT example with 1000 angles between 0 and π. In Figure 13, we compare the different regularizers. The ALR tends to oversmooth the reconstructions and the WPP, WPP_ϵ and $\text{WPP}_{\epsilon,\rho}$ are not able to reconstruct

[3] Available at https://zenodo.org/records/3384092#.Ylglz3VBwgM.

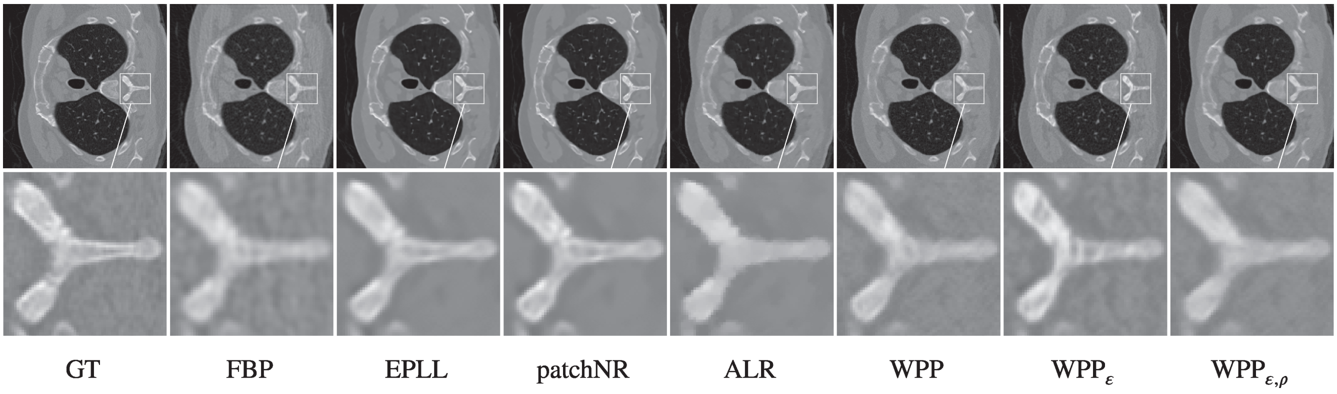


FIGURE 13 Comparison of different methods for low-dose CT reconstruction. The zoomed-in part is marked with a white box. *Top*: full image. *Bottom*: zoomed-in part.

TABLE 1 Low-dose CT.

	FBP	EPLL	patchNR	ALR	WPP	WPP$_\varepsilon$	WPP$_{\varepsilon,\rho}$
PSNR	30.37 ± 2.95	34.89 ± 4.41	**35.19** ± 4.52	33.59 ± 3.73	32.61 ± 3.12	31.34 ± 4.22	32.79 ± 3.27
SSIM	0.739 ± 0.141	0.821 ± 0.154	**0.829** ± 0.152	0.808 ± 0.146	0.777 ± 0.121	0.757 ± 0.158	0.791 ± 0.131
FSIM	0.941 ± 0.037	0.935 ± 0.073	0.935 ± 0.080	0.945 ± 0.061	0.950 ± 0.048	0.936 ± 0.064	**0.951** ± 0.049

Note: Averaged quality measures and standard deviations of the high-resolution reconstructions. Evaluated on the first 100 test images of LoDoPab dataset. Best values are marked in bold.

sharp edges. Both, the EPLL and the patchNR perform well, while the patchNR gives slightly more accurate and realistic reconstructions. This can be also seen quantitatively in Table 1, where we evaluated the methods on the first 100 test images of the dataset. Here the patchNR gives the best results with respect to PSNR and SSIM. The weak performance of WPP, WPP$_\varepsilon$ and WPP$_{\varepsilon,\rho}$ can be explained by the diversity of the CT dataset, leading to very different patch distributions. Therefore, defining the reference patch distribution as a mixture of patch distributions of the given 6 reference images is not sufficient for a good reconstruction. Note that for CT data the LPIPS is not meaningful, since the feature-extracting network is trained on natural images, which differ substantially from the CT scans. Therefore, we cannot expect informative results from LPIPS.

7.2.2 | Limited-angle CT

Next, we consider a limited-angle CT setting, that is, instead of using 1000 equidistant angles between 0 and π, we cut off the first and last 100 angles such that we consider 144° instead of 180°. This leads to a much worse FBP due to the missing part in the measurement. In Figure 14, we compare the different regularizers. Again, the ALR smooths out the reconstruction and the WPP, WPP$_\varepsilon$ and WPP$_{\varepsilon,\rho}$ are not able to reconstruct the missing parts well. In contrast, the EPLL and the patchNR give good reconstructions, although the patchNR gives sharper edges as can be seen in the right part of the zoomed-in part. In Table 2, a quantitative comparison is given. Again, the patchNR gives the best results with respect to PSNR and SSIM.

7.3 | Super-resolution

For image super-resolution, we want to recover a high-resolution image from a given low-resolution image. The forward operator F consists of a convolution with a 16×16 Gaussian blur kernel of a certain standard deviation specified below and a subsampling process. For the noise model, we consider additive Gaussian noise with standard deviation $\Xi \sim \mathcal{N}(0, \sigma^2 I)$ with standard deviation $\sigma = 0.01$. Consequently, we want to minimize the variational problem (12) with $\alpha = \beta\sigma^2$.

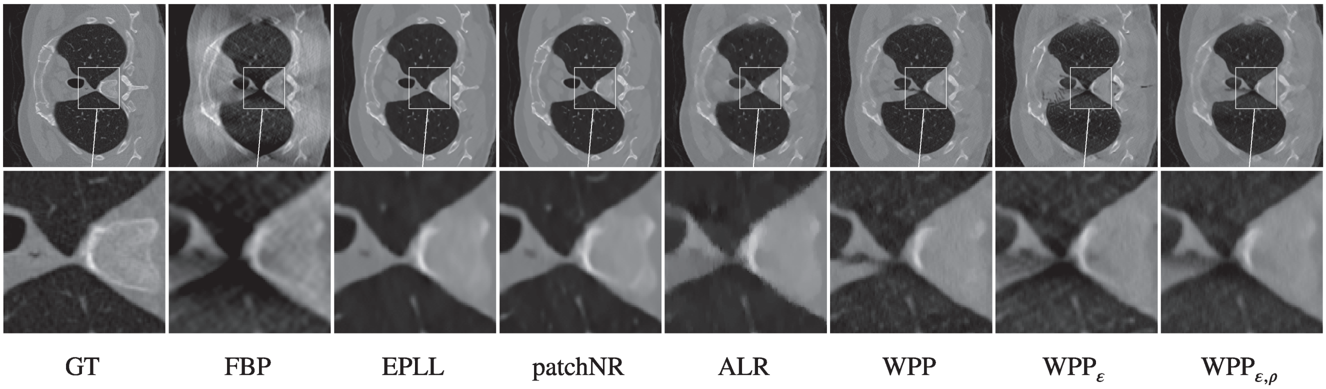


FIGURE 14 Comparison of different methods for limited-angle CT reconstruction. The zoomed-in part is marked with a white box. *Top*: full image. *Bottom*: zoomed-in part. As visualized in the zoomed-in part, only EPLL and patchNR are able to reconstruct the missing part in the middle. The patchNR gives the sharpest results on the right of the zoomed-in part.

TABLE 2 Limited-angle CT.

	FBP	EPLL	patchNR	ALR	WPP	WPP$_\varepsilon$	WPP$_{\varepsilon,\rho}$
PSNR	21.96 ± 2.25	33.14 ± 3.58	**33.26** ± 3.58	31.27 ± 2.94	29.92 ± 2.36	25.10 ± 3.98	30.00 ± 2.55
SSIM	0.531 ± 0.097	0.804 ± 0.154	**0.811** ± 0.151	0.783 ± 0.143	0.737 ± 0.114	0.642 ± 0.144	0.753 ± 0.125
FSIM	0.913 ± 0.032	0.920 ± 0.071	0.921 ± 0.077	**0.929** ± 0.053	0.920 ± 0.048	0.846 ± 0.111	0.922 ± 0.048

Note: Averaged quality measures and standard deviations of the high-resolution reconstructions. Evaluated on the first 100 test images of LoDoPab dataset. Best values are marked in bold.

We consider two different types of super-resolution: First, we deal with the super-resolution of material data. Here we assume that we are given one high-resolution reference image of the material which we can use as prior knowledge. Second, we consider zero-shot super-resolution of natural images, where no reference data is known.

7.3.1 | Material data

The dataset consists of 2D slices of size 600 × 600 from a 3D material image of size 2560 × 2560 × 2120. This has been acquired by synchrotron micro-computed tomography at the SLS beamline TOMCAT. More specifically, we consider a composite ("SiC Diamond") obtained by microwave sintering of silicon and diamonds, see [110]. We assume that we are given one high-resolution reference image of size 600 × 600. The blur kernel of the forward operator F has standard deviation 2 and we consider a subsampling factor of 4 (in each direction).

In Figure 15, we compare the different regularizers, where we choose the bicubic interpolation as initialization. In the reconstruction of the ALR and the WPP, we can observe a significant blur, in particular in the regions between the edges. In contrast, the EPLL and the patchNR reconstructions are sharper and more realistic. The WPP, WPP$_\varepsilon$ and WPP$_{\varepsilon,\rho}$ reconstructions have quite similar lower quality. A quantitative comparison is given in Table 3.

7.3.2 | Zero-shot super-resolution

We consider the grayscale BSD68 dataset [72]. The blur kernel of the forward operator F has standard deviation 1 and we consider a subsampling factor of 2.

We assume that no reference data is given, so that we need to extract our prior information from the given low-resolution observation. Here we exploit the concepts of zero-shot super-resolution by internal learning. The main observation is that the patch distribution of natural images is self-similar across the scales [37, 99, 120]. Thus the patch distributions of the same image are similar at different resolutions. An illustrative example with two images from the BSD68 dataset is given in Figure 16. The reconstruction of the unknown high-resolution image using the different regularizers is visualized in Figure 17. Here ALR and EPLL smooth out parts of the reconstruction, in particular when these

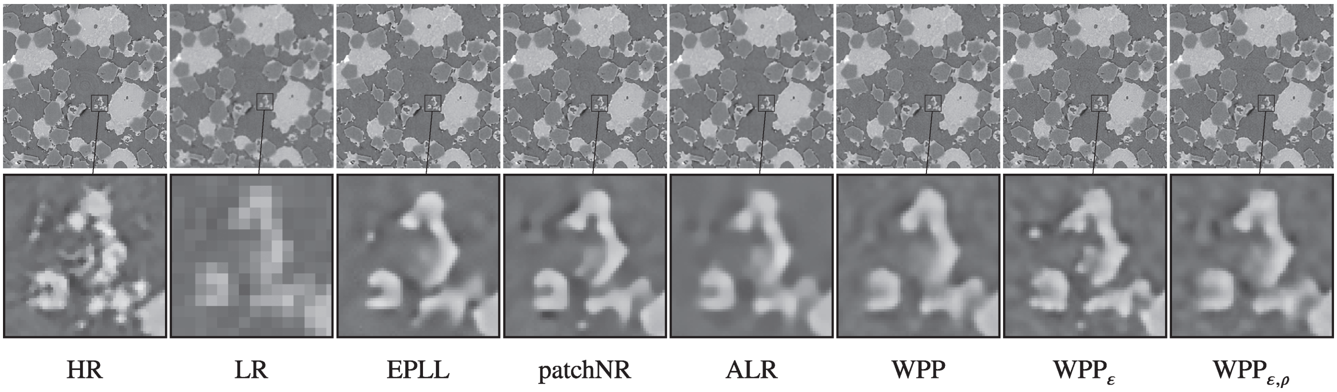


FIGURE 15 Comparison of different methods for super-resolution. The zoomed-in part is marked with a black box. *Top*: full image. *Bottom*: zoomed-in part.

TABLE 3 Super-resolution.

	bicubic	**EPLL**	**patchNR**	**ALR**	**WPP**	**WPP$_\varepsilon$**	**WPP$_{\varepsilon,\rho}$**
PSNR	25.63 ± 0.56	28.34 ± 0.50	**28.53** ± 0.49	27.76 ± 0.52	27.55 ± 0.46	26.60 ± 0.30	27.46 ± 0.48
SSIM	0.699 ± 0.012	0.770 ± 0.008	**0.780** ± 0.008	0.758 ± 0.005	0.737 ± 0.007	0.698 ± 0.020	0.727 ± 0.006
LPIPS	0.414 ± 0.011	0.175 ± 0.009	**0.161** ± 0.007	0.187 ± 0.005	0.188 ± 0.008	0.177 ± 0.027	0.186 ± 0.007
FSIM	0.878 ± 0.005	0.933 ± 0.005	**0.940** ± 0.004	0.932 ± 0.003	0.937 ± 0.003	0.919 ± 0.007	0.938 ± 0.003

Note: Averaged quality measures and standard deviations of the high-resolution reconstructions. Evaluated on the material data test set. Best values are marked in bold.

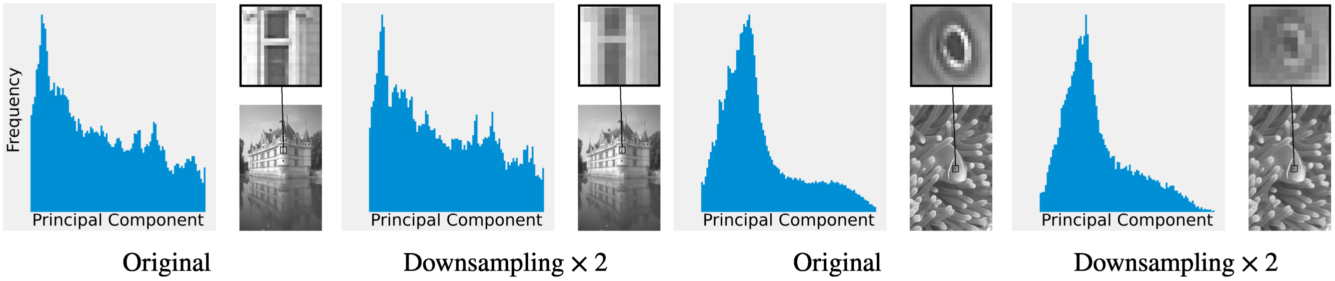


FIGURE 16 We utilize the same method as in Figure 3 to visualize the patch distribution for two natural images from the BSD68 data set [72]. Both images are downsampled by a factor of 2 using a Gaussian blur operator in combination with additive Gaussian noise. We project all patches onto the direction of the first principal component and illustrate the resulting distribution with a histogram. For every image, the first principal component explains more than 70% of the total variance. Visually, the projected patch distributions of the original images and their downsampled versions are highly similar.

parts are blurry in the low-resolution part, see, for example, the stripes of the zebra or the fur pattern of the giraffe. In contrast, WPP, WPP$_{\varepsilon,\rho}$ and patchNR are able to reconstruct well and without blurred parts. The WPP$_\varepsilon$ reconstructions admit structured noise, which can be seen in the upper right corner of the zoomed-in part of the giraffe. A quantitative comparison is given in Table 4. Again, the patchNR performs best in terms of quality measures.

7.3.3 | Zero-shot super-resolution with VGG-16 features

We repeat the previous experiment but replace patches with neural network features as laid out in Section 3. More specifically we test the possibility of utilizing the internal image statistics of the downsampled image by extracting features of a VGG-16 classification network [101]. This network has been trained to assign images of resolution 469 × 387 into 1000

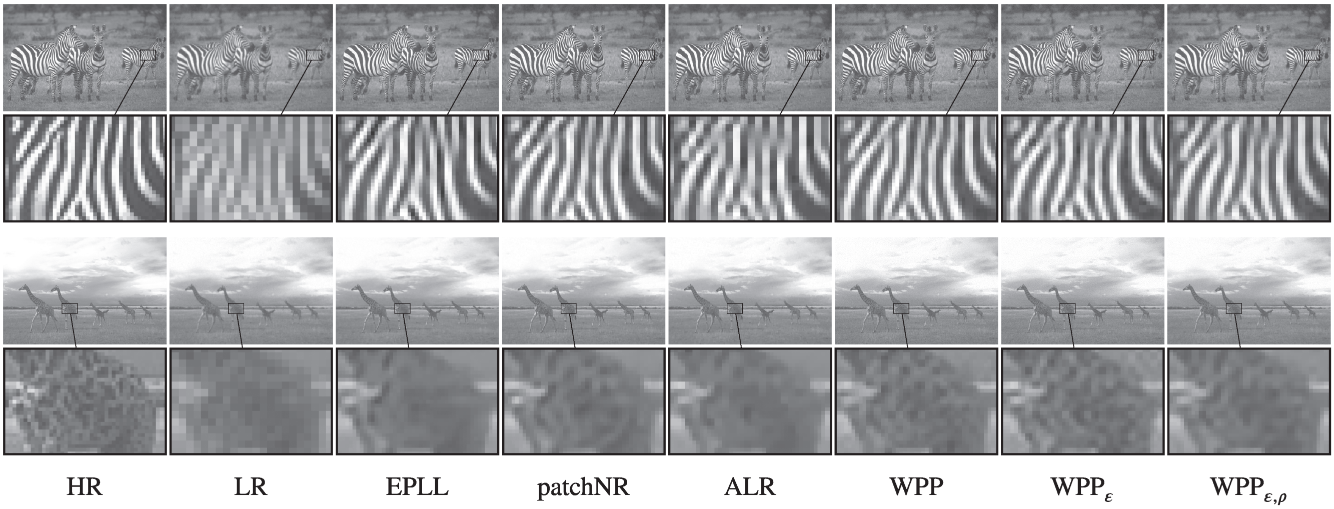
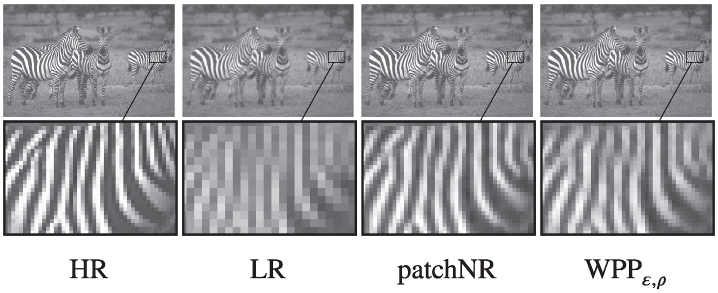

HR
LR
EPLL
patchNR
ALR
WPP
$\mathrm{WPP}_{\varepsilon}$
$\mathrm{WPP}_{\varepsilon,\rho}$

FIGURE 17 Comparison of different methods for zero-shot super-resolution. The zoomed-in part is marked with a black box. *Top*: full image. *Bottom*: zoomed-in part. The ALR and the EPLL smooth out parts of the reconstruction when these parts are blurry in the low-resolution part, see, for example, the stripes of the zebra or the fur pattern of the giraffe.

TABLE 4 Zero-shot super-resolution.

	bicubic	**EPLL**	**patchNR**	**ALR**	**WPP**	**$\mathbf{WPP}_{\varepsilon}$**	**$\mathbf{WPP}_{\varepsilon,\rho}$**
PSNR	27.06 ± 3.39	28.90 ± 3.53	**29.08** ± 3.58	28.61 ± 3.51	28.20 ± 3.03	27.92 ± 2.81	28.36 ± 3.24
SSIM	0.782 ± 0.075	0.838 ± 0.065	**0.846** ± 0.061	0.829 ± 0.066	0.790 ± 0.055	0.771 ± 0.056	0.806 ± 0.055
LPIPS	0.327 ± 0.075	0.204 ± 0.079	0.203 ± 0.075	**0.196** ± 0.072	0.251 ± 0.062	0.259 ± 0.066	0.245 ± 0.065
FSIM	0.952 ± 0.023	0.977 ± 0.010	**0.980** ± 0.008	0.974 ± 0.013	0.969 ± 0.023	0.964 ± 0.029	0.973 ± 0.016

Note: Averaged quality measures and standard deviations of the high-resolution reconstructions. Evaluated on BSD68. Best values are marked in bold.



	bicubic	patchNR	$\mathrm{WPP}_{\varepsilon,\rho}$
PSNR	27.06 ± 3.39	**27.97**±3.00	**27.97**±3.25
SSIM	0.782 ± 0.075	0.775±0.051	**0.805**±0.054
LPIPS	0.327 ± 0.075	0.252±0.060	**0.233**±0.059
FSIM	0.952 ± 0.023	0.967±0.022	**0.974**±0.0120

FIGURE 18 *Left*: Comparison of different methods for zero-shot super-resolution with VGG16 features instead of patches. The zoomed-in part is marked with a black box. *Right*: Averaged quality measures and standard deviations of the high-resolution reconstructions. Evaluated on BSD68. Best values are marked in bold.

predefined classes. To extract low-level local information, we take the output of the first convolutional layer ("conv1-1"). The resulting features have dimension 64. We take the best-performing likelihood-based regularizer, the patchNR, and the best-performing divergence-based regularizer, the $WPP_{\varepsilon,\rho}$, from the previous experiment and replace the utilized patch distributions with the resulting feature distributions. An exemplary result is visualized in Figure 18 (left) which also contains a table with a quantitative comparison for the full BSD68 dataset (right). Overall the feature-based methods perform worse than the patch-based methods, but we still improve upon the bicubic baseline despite using features geared towards classification on an unrelated dataset. The use of techniques from self-supervised learning might enable the extraction of more meaningful features. Such features could potentially carry semantic information or have even lower dimensions than patches. This might accelerate image reconstruction or lead to improved performance. We leave the development of such features as future work.

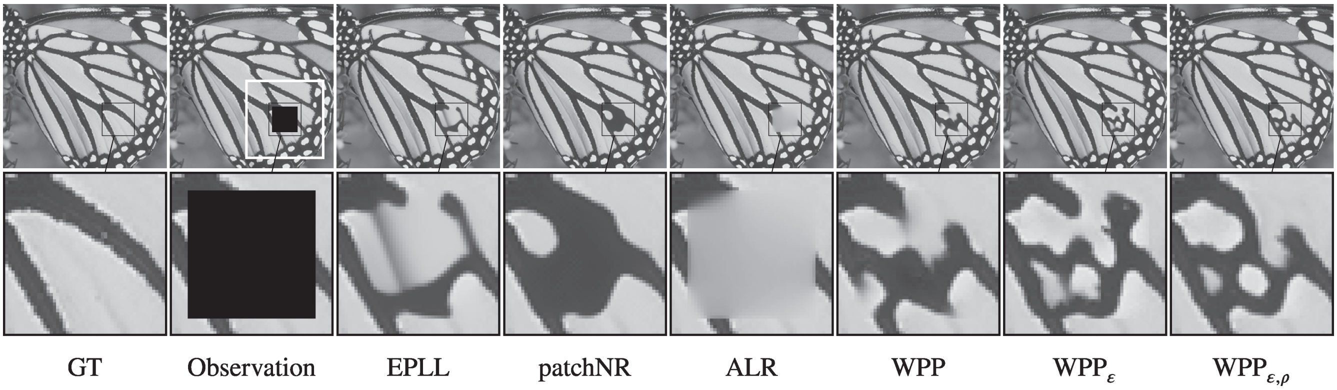


FIGURE 19 Comparison of different methods for inpainting. The black boxes mark missing parts. The reference patches for the regularizers are obtained from the region within the white border. The zoomed-in part is marked with a black box. *Top*: full image. *Bottom*: zoomed-in part.

7.4 | Inpainting

The task of image inpainting is to reconstruct missing data in the observation. For a given inpainting mask $m \in \{0,1\}^n$, the forward operator F is given by $F(x) = x \odot m$. In this subsection, we focus on region inpainting, where large regions of data are missing in the observation. We assume that there is no additional noise in the observation, leading to the negative log-likelihood

$$-\log(p_{Y|X=x}(y)) = \begin{cases} 0, & \text{if } F(x) = y, \\ +\infty, & \text{else.} \end{cases} \tag{57}$$

Consequently, we are searching for

$$\arg\min x \in \mathbb{R}^d \{\mathcal{R}(x) : F(x) = y\}.$$

We consider the Set5 dataset [17] and assume that no reference data is given, such that we extract the prior information from a predefined area around the missing part of the observation.

In Figure 19, we compare the results of the different regularizers for the inpainting task. The missing part is the black rectangle in the observation and the reference patches are extracted around the missing part, which is visualized with the larger white box. We observe that ALR fails completely. In contrast, EPLL and patchNR are able to connect the lower missing black line. Here, the patchNR gives visually better results, in particular, the black lines are much sharper. Further, WPP, WPP$_\varepsilon$ and WPP$_{\varepsilon,\rho}$ fill out the missing part in a different way, as they aim to match the patch distribution between the reference part and the missing part. Obviously, the filled area is influenced by the patch distribution of the lower right corner in the reference part.

7.5 | Posterior sampling

In this section, we apply ULA (52). First, we use it for posterior sampling in image inpainting, where we can expect a high variety in the reconstructions due to the highly ill-posed problem. Then we quantify the uncertainty in limited-angle CT reconstructions.

7.5.1 | Posterior sampling for image inpainting

We apply ULA (52) with the different regularizers for the same task of image inpainting as in Section 7.4. Again, the data-fidelity term vanishes, so that (54) simplifies to

$$X_{k+1} = X_k - \delta\lambda\nabla\mathcal{R}(X_k) + \sqrt{2\delta}Z_{k+1}.$$

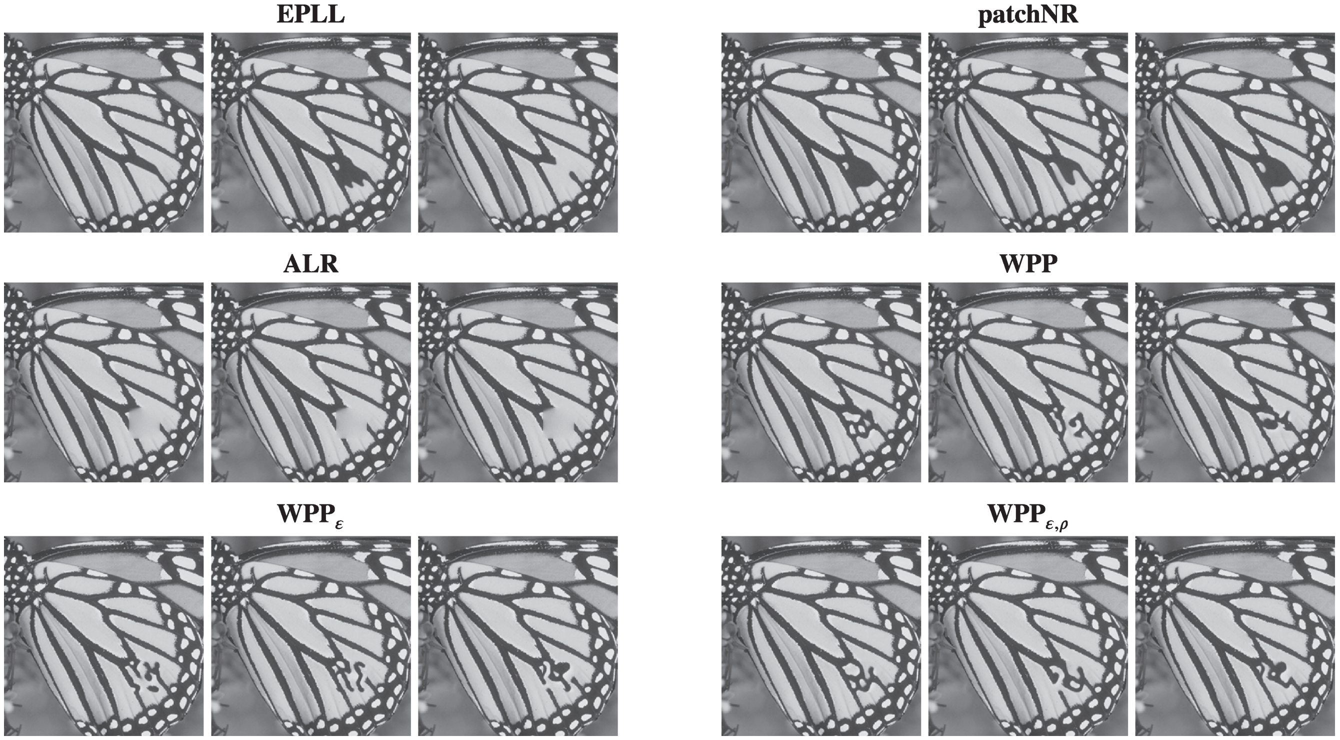

EPLL
patchNR
ALR
WPP
$\mathrm{WPP}_{\varepsilon}$
$\mathrm{WPP}_{\varepsilon,\rho}$

FIGURE 20 Comparison of regularizers for posterior sampling for inpainting. The ground truth and the observation images are the same as in Figure 19.

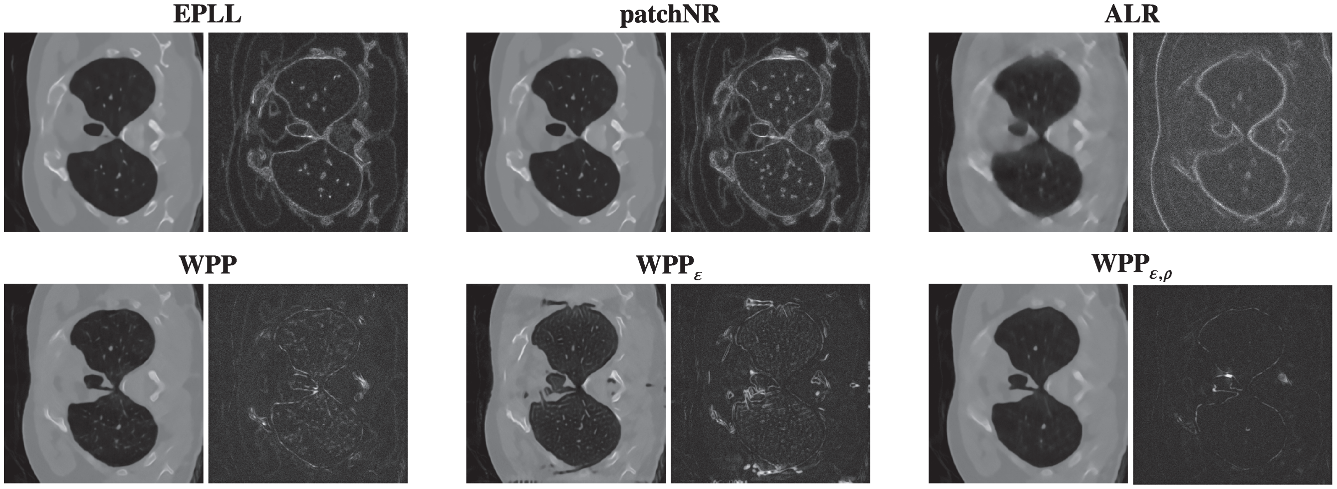

EPLL
patchNR
ALR
WPP
$\mathrm{WPP}_{\varepsilon}$
$\mathrm{WPP}_{\varepsilon,\rho}$

FIGURE 21 Comparison of regularizers for uncertainty quantification for limited-angle CT. Mean image (left) and pixel-wise standard deviation (right) of the reconstructions. The ground truth and the FBP are the same as in Figure 14.

Since the inverse problem is highly ill-posed due to the missing part, we can expect a high variety in the reconstructions. In Figure 20, we compare the different methods. The ground truth and the observation are the same as in Figure 19. We illustrated three different reconstruction samples. Again, we observe differences between the regularizers of Sections 4 and 5. First, we note that the ALR is, similar to MAP inpainting, not able to give meaningful reconstructions. On the other hand, the EPLL and the patchNR can reconstruct well, although the EPLL reconstructions look more realistic and are more diverse. The regularizers from Section 5give the most diverse reconstructions. Here the reconstruction quality is similar for WPP, $\mathrm{WPP}_{\varepsilon}$ and $\mathrm{WPP}_{\varepsilon,\rho}$.

7.5.2 | Uncertainty quantification for limited-angle CT

Finally, we consider the limited-angle CT reconstruction as in Section 7.2. The negative log-likelihood is given by (55) so that (54) reads as

$$X_{k+1} = X_k + \delta \nabla \sum_{i=1}^{d} e^{-F(X_k)_i \mu} N_0 + e^{-y_i \mu} N_0 \big(F(X_k)_i \mu - \log(N_0)\big) - \delta \alpha \nabla \mathcal{R}(X_k) + \sqrt{2\delta} Z_{k+1}.$$

In Figure 21, we compare the reconstructions of the different regularizers. We illustrate the mean image (left) and the pixel-wise standard deviation (right) of the corresponding regularizers for 10 reconstructions. The standard deviation can be seen as the uncertainty in the reconstruction and the brighter a pixel is, the less secure is the model in its reconstruction. As in the MAP reconstruction, EPLL and patchNR are able to reconstruct best. Moreover, the standard deviation of EPLL and patchNR are most meaningful and the highest uncertainty is in regions, where the FBP has missing parts. In contrast, the ALR smooths out the reconstruction. While the reconstructions of WPP and WPP$_{\varepsilon,\rho}$ appear almost similar at first glance, the WPP admits more uncertainty in its reconstructions. Nevertheless, both regularizers are not able to reconstruct the corrupted parts in the FBP. The WPP$_{\varepsilon}$ has a lot of artifacts in its reconstructions. Moreover, most of the uncertainty is observable in the artificially reconstructed artifacts.

ACKNOWLEDGEMENTS

M.P. and G.S. acknowledge funding from the German Research Foundation (DFG) within the project BIOQIC (GRK2260/289347353). F.A., A.W., and G.S. acknowledge support from the DFG through Germany's Excellence Strategy—The Berlin Mathematics Research Center MATH+ under project AA5-6. P.H. acknowledges support from the DFG within the SPP 2298 "Theoretical Foundations of Deep Learning" (STE 571/17-1). J.H. acknowledges support from the DFG within the project STE 571/16-1 and by the EPSRC program grant "The Mathematics of Deep Learning" with reference EP/V026259/1.

The material data presented in Section 7.3 was obtained as part 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 TOMCAT beamline of the Swiss Light Source (SLS), performed by A. Saadaldin, D. Bernard, and F. Marone Welford. We express our gratitude to the Paul Scherrer Institut, Villigen, Switzerland, for providing synchrotron radiation beamtime at the TOMCAT beamline X02DA of the SLS.

ORCID

Moritz Piening https://orcid.org/0009-0003-3877-4511

REFERENCES

[1] J. Adler, H. Kohr, A. Ringh, J. Moosmann, S. Banert, M. J. Ehrhardt, G. R. Lee, K. Niinimaki, B. Gris, O. Verdier, J. Karlsson, W. J. Palenstijn, O. Öktem, C. Chen, H. A. Loarca, and M. Lohmann. Operator discretization library (ODL). 2018 https://doi.org/10.5281/zenodo.1442734.

[2] J. Adler and O. Öktem. Deep Bayesian inversion. 2018. arXiv preprint, arXiv:1811.05910.

[3] F. Altekrüger, A. Denker, P. Hagemann, J. Hertrich, P. Maass, and G. Steidl, PatchNR: Learning from very few images by patch normalizing flow regularization, Inverse Problems **39** (2023), no. 6, 064006.

[4] Altekrüger, F., P. Hagemann, and G. Steidl, 2023: *Conditional generative models are provably robust: pointwise guarantees for Bayesian inverse problems*. Transactions on Machine Learning Research. Journal of Machine Learning Research Inc. (JMLR) New York.

[5] F. Altekrüger and J. Hertrich, WPPNets and WPPFlows: The power of Wasserstein patch priors for superresolution, SIAM J. Imag. Sci. **16** (2023), no. 3, 1033–1067.

[6] A. Andrle, N. Farchmin, P. Hagemann, S. Heidenreich, V. Soltwisch, and G. Steidl, "*Invertible neural networks versus MCMC for posterior reconstruction in grazing incidence x-ray fluorescence*," *Int. Conf. Scale Space and Variational Methods in Computer Vision*, Springer International Publishing, Basel. 2021, pp. 528–539.

[7] L. Ardizzone, J. Kruse, C. Rother, and U. Köthe, "*Analyzing inverse problems with invertible neural networks*," *Int. Conf. Learn. Represent.*, Curran Associates's, Red Hook. 2018.

[8] L. Ardizzone, C. Lüth, J. Kruse, C. Rother, and U. Köthe. Guided image generation with conditional invertible neural networks. 2019. arXiv preprint, arXiv:1907.02392.

[9] M. Arjovsky, S. Chintala, and L. Bottou, "*Wasserstein generative adversarial networks*," *Int. Conf. Mach. Learn.*, Curran Associates's, Red Hook. 2017, pp. 214–223.

[10] S. G. Armato, G. McLennan, L. Bidaut, M. F. McNitt-Gray, C. R. Meyer, A. P. Reeves, B. Zhao, D. R. Aberle, C. I. Henschke, E. A. Hoffman, E. A. Kazerooni, H. MacMahon, E. J. R. van Beek, D. Yankelevitz, A. M. Biancardi, P. H. Bland, M. S. Brown, R. M. Engelmann, G. E. Laderach, D. Max, R. C. Pais, D. P.-Y. Qing, and R. Y. Roberts, 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** (2011), no. 2, 915–931. https://doi.org/10.1118/1.3528204.

[11] S. Arridge, P. Maass, O. Öktem, and C. Schönlieb, Solving inverse problems using data-driven models, Acta Numer. **28** (2019), 1–174.

[12] G. Batzolis, J. Stanczuk, C.-B. Schönlieb, and C. Etmann. Conditional image generation with score-based diffusion models. 2021. arXiv preprint, arXiv:2111.13606.

[13] J. Behrmann, W. Grathwohl, R. Chen, D. Duvenaud, and J.-H. Jacobsen, "*Invertible residual networks*," *Int. Conf. Mach. Learn.*, Curran Associates's, Red Hook. 2019, pp. 573–582.

[14] J. Behrmann, P. Vicol, K.-C. Wang, R. Grosse, and J.-H. Jacobsen. Understanding and mitigating exploding inverses in invertible neural networks. 2020. arXiv preprint, arXiv:2006.09347.

[15] M. Benning and M. Burger, Modern regularization methods for inverse problems, Acta Numer. **27** (2018), 1–111.

[16] M. Bertero, P. Boccacci, and C. De Mol, *Introduction to inverse problems in imaging*, CRC Press, Boca Raton. 2021.

[17] M. Bevilacqua, A. Roumy, C. M. Guillemot, and M.-L. Alberi-Morel, "*Low-complexity single-image super-resolution based on nonnegative neighbor embedding*," *British Machine Vision Conf*., BMVA Press, Durham, UK. 2012.

[18] Z. Cai, J. Tang, S. Mukherjee, J. Li, C. B. Schönlieb, and X. Zhang. NF-ULA: Langevin Monte Carlo with normalizing flow prior for imaging inverse problems. 2023. arXiv preprint, arXiv:2304.08342.

[19] T. Q. Chen and M. Schmidt, "*Fast patch-based style transfer of arbitrary style*," *Advances in neural information processing systems*, Curran Associates's, Red Hook. 2016.

[20] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian, Image denoising by sparse 3-d transform-domain collaborative filtering, IEEE Trans. Image Process. **16** (2007), no. 8, 2080–2095.

[21] C.-A. Deledalle, S. Parameswaran, and T. Q. Nguyen, Image denoising with generalized Gaussian mixture model patch priors, SIAM J. Imag. Sci. **11** (2018), no. 4, 2568–2609.

[22] J. Delon and A. Desolneux, A Wasserstein-type distance in the space of Gaussian mixture models, SIAM J. Imag. Sci. **13** (2020), no. 2, 936–970.

[23] A. P. Dempster, N. M. Laird, and D. B. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. R. Stat. Soc. Ser. B **39** (1977), no. 1, 1–22.

[24] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "*Density estimation using Real NVP*,"*Int. Conf. Learn. Represent*., Curran Associates's, Red Hook. 2016.

[25] W. Dong, X. Li, L. Zhang, and G. Shi, "*Sparsity-based image denoising via dictionary learning and structural clustering*,"*IEEE Conf. Comput. Vision Pattern Recognit*., The Institute of Electrical and Electronics Engineers (IEEE), New York City. 2011, pp. 457–464.

[26] C. Du, T. Li, T. Pang, S. Yan, and M. Lin, "*Nonparametric generative modeling with conditional sliced-Wasserstein flows*,"*Int. Conf. Mach. Learn*., Curran Associates's, Red Hook. 2023, pp. 8565–8584.

[27] A. Elnekave and Y. Weiss, Generating natural images with direct patch distributions matching, Eur. Conf. Comput. Vision (2022), 544–560.

[28] H. W. Engl, M. Hanke, and A. Neubauer, *Regularization of inverse problems*, Vol **375**, Springer Science & Business Media, Berlin/Heidelberg. 1996.

[29] J. Feydy, T. Séjourné, F.-X. Vialard, S.-i. Amari, A. Trouvé, and G. Peyré, "*Interpolating between optimal transport and MMD using Sinkhorn divergences*,"*Int. Conf. Artificial Intell. Stat*., Proceedings of Machine Learning Research (PMLR), New York City. 2019, pp. 2681–2690.

[30] R. Friedman and Y. Weiss. Posterior sampling for image restoration using explicit patch priors. 2021. arXiv preprint, arXiv:2104.09895.

[31] Z. Gao, E. Edirisinghe, and S. Chesnokov, "*Image super-resolution using cnn optimised by self-feature loss*,"*Int. Conf. Image Process*., The Institute of Electrical and Electronics Engineers (IEEE), New York City. 2019, pp. 2816–2820.

[32] L. Gatys, A. S. Ecker, and M. Bethge, Texture synthesis using convolutional neural networks, Adv. Neural Inf. Proces. Syst. **28** (2015), pp. 262–270.

[33] L. Gatys, A. S. Ecker, and M. Bethge, Image style transfer using convolutional neural networks, IEEE Conf. Comput. Vision Pattern Recognit. **29** (2016), 2414–2423.

[34] A. Genevay, M. Cuturi, G. Peyré, and F. Bach, Stochastic optimization for large-scale optimal transport, Adv. Neural Inf. Proces. Syst. **29** (2016), pp. 3432–3440.

[35] A. Genevay, G. Peyré, and M. Cuturi, "*Learning generative models with Sinkhorn divergences*,"*Int. Conf. Artificial Intell. Stat*., 2018, Proceedings of Machine Learning Research (PMLR), New York City. pp. 1608–1617.

[36] D. Gilton, G. Ongie, and R. Willett, "*Learned patch-based regularization for inverse problems in imaging*,"*IEEE Int. Workshop Comput. Adv. Multi-Sensor Adapt. Process*., The Institute of Electrical and Electronics Engineers (IEEE), New York City. 2019, pp. 211–215.

[37] D. Glasner, S. Bagon, and M. Irani, "*Super-resolution from a single image*,"*IEEE Int. Conf. Comput. Vision*, Curran Associates's, Red Hook. 2009, pp. 349–356.

[38] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, Generative adversarial nets, Adv. Neural Inf. Proces. Syst. **27** (2014), pp. 2672–2680.

[39] D. Grana, T. Fjeldstad, and H. Omre, Bayesian Gaussian mixture linear inversion for geophysical inverse problems, Math. Geosci. **49** (2017), no. 4, 493–515.

[40] N. Granot, B. Feinstein, A. Shocher, S. Bagon, and M. Irani, Drop the GAN: In defense of patches nearest neighbors as single image generative models, IEEE/CVF Conference on Computer Vision and Pattern Recognition **35** (2022), 13460–13469.

[41] A. Griewank and A. Walther, *Evaluating derivatives: principles and techniques of algorithmic differentiation*, SIAM, Philadelphia, PA. 2008.

[42] A. Grover, M. Dhar, and S. Ermon, Flow-GAN: Combining maximum likelihood and adversarial learning in generative models, Proc. AAAI Conf. Artificial Intell. **32** (2018), no. 1, pp. 3069–3076.

[43] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville, Improved training of Wasserstein GANs, Adv. Neural Inf. Proces. Syst. **30** (2017), pp. 5769–5779.

[44] J. Gutierrez, J. Rabin, B. Galerne, and T. Hurtut, "*Optimal patch assignment for statistically constrained texture synthesis*,"*Int. Conf. Scale Space and Variational Methods in Comput. Vision*, Springer International Publishing, Basel. 2017, pp. 172–183.

[45] P. Hagemann, J. Hertrich, F. Altekrüger, R. Beinert, J. Chemseddine, and G. Steidl. Posterior sampling based on gradient flows of the MMD with negative distance kernel. 2023a. arXiv preprint, arXiv:2310.03054.

[46] P. Hagemann, J. Hertrich, and G. Steidl, Stochastic normalizing flows for inverse problems: A Markov chains viewpoint, SIAM/ASA J. Uncertain. Quant. **10** (2022), no. 3, 1162–1190.

[47] P. Hagemann, J. Hertrich, and G. Steidl, *Generalized normalizing flows via Markov Chains*, Foundations and Applications, Cambridge University Press, Elements in Non-local Data Interactions, 2023b.

[48] P. Hagemann and S. Neumayer, Stabilizing invertible neural networks using mixture models, Inverse Problems **37** (2021), no. 8, 085002.

[49] M. Hasannasab, J. Hertrich, F. Laus, and G. Steidl, Alternatives to the EM algorithm for ML estimation of location, scatter matrix, and degree of freedom of the Student-t distribution, Numer. Algorithms **87** (2021), no. 1, 77–118.

[50] W. He, R. Yu, Y. Zheng, and T. Jiang, "*Image denoising using asymmetric Gaussian mixture models*,"*IEEE Int. Symp. Sensing Inst. IoT Era*, The Institute of Electrical and Electronics Engineers (IEEE), New York City. 2018, pp. 1–4.

[51] J. Hertrich, A. Houdard, and C. Redenbach, Wasserstein patch prior for image superresolution, IEEE Trans. Comput. Imaging **8** (2022a), 693–704.

[52] J. Hertrich, D. P. L. Nguyen, J.-F. Aujol, D. Bernard, Y. Berthoumieu, A. Saadaldin, and G. Steidl, PCA reduced Gaussian mixture models with application in superresolution, Inverse Problems Imaging **16** (2022b), no. 2, 341–366.

[53] A. Hore and D. Ziou, "*Image quality metrics: PSNR vs SSIM*,"*Int. Conf. Pattern Recognit.*, The Institute of Electrical and Electronics Engineers (IEEE), New York City. 2010, pp. 2366–2369.

[54] A. Houdard, C. Bouveyron, and J. Delon, High-dimensional mixture models for unsupervised image denoising (HDMI), SIAM J. Imag. Sci. **11** (2018), no. 4, 2815–2846.

[55] A. Houdard, A. Leclaire, N. Papadakis, and J. Rabin, "*Wasserstein generative models for patch-based texture synthesis*,"*Int. Conf. Scale Space and Variational Methods in Comput. Vision*, Springer International Publishing, Basel. 2021, pp. 269–280.

[56] A. Houdard, A. Leclaire, N. Papadakis, and J. Rabin, A generative model for texture synthesis based on optimal transport between feature distributions, J. Math. Imaging Vision **65** (2023), no. 1, 4–28.

[57] P. Jaini, I. Kobyzev, Y. Yu, and M. Brubaker, "*Tails of Lipschitz triangular flows*,"*Int. Conf. Mach. Learn.*, Vol **119**, Proceedings of Machine Learning Research (PMLR), New York City. 2020, pp. 4673–4681.

[58] D. P. Kingma and J. Ba, "*ADAM: a method for stochastic optimization*,"*Int. Conf. Learn. Represent.*, Curran Associates's, Red Hook. 2015.

[59] D. P. Kingma and M. Welling. Auto-encoding variational Bayes. 2013. arXiv preprint, arXiv:1312.6114.

[60] J. Latz, On the well-posedness of Bayesian inverse problems, SIAM/ASA J. Uncertain. Quant. **8** (2020), no. 1, 451–482.

[61] R. Laumont, V. D. Bortoli, A. Almansa, J. Delon, A. Durmus, and M. Pereyra, Bayesian imaging using Plug & Play priors: When Langevin meets Tweedie, SIAM J. Imag. Sci. **15** (2022), no. 2, 701–737.

[62] F. Laus, M. Nikolova, J. Persch, and G. Steidl, A nonlocal denoising algorithm for manifold-valued images using second order statistics, SIAM J. Imag. Sci. **10** (2017), no. 1, 416–448.

[63] M. Lebrun, A. Buades, and J.-M. Morel, A nonlocal Bayesian image denoising algorithm, SIAM J. Imag. Sci. **6** (2013), no. 3, 1665–1688.

[64] M. Lebrun, M. Colom, A. Buades, and J. Morel, Secrets of image denoising cuisine, Acta Numer. **21** (2012), 475–576.

[65] J. Leuschner, M. Schmidt, D. O. Baguer, and P. Maass, LoDoPaB-CT, a benchmark dataset for low-dose computed tomography reconstruction, Sci. Data **8** (2021), no. 109, 109–109..

[66] Y. Li, N. Wang, J. Liu, and X. Hou, "*Demystifying neural style transfer*,"*Int. Joint Conf. Artificial Intell.*, Curran Associates's, Red Hook. 2017, pp. 2230–2236.

[67] L. Liang, C. Liu, Y.-Q. Xu, B. Guo, and H.-Y. Shum, Real-time texture synthesis by patch-based sampling, ACM Trans. Graph. **20** (2001), no. 3, 127–150.

[68] J. Lim, S. Ryu, J. W. Kim, and W. Y. Kim, Molecular generative model based on conditional variational autoencoder for de novo molecular design, J. Chem. **10** (2018), no. 1, 1–9.

[69] H. Liu, B. Jiang, Y. Song, W. Huang, and C. Yang, Rethinking image inpainting via a mutual encoder-decoder with feature equalizations, Eur. Conf. Comput. Vision Part II 16 (2020), 725–741.

[70] S. Liu, X. Zhou, Y. Jiao, and J. Huang. Wasserstein generative learning of conditional distribution. 2021. arXiv preprint, arXiv:2112.10039.

[71] S. Lunz, O. Öktem, and C.-B. Schönlieb, "*Adversarial regularizers in inverse problems*,"*Adv. Neural Informat. Process. Syst.*, Curran Associates's, Red Hook. 2018.

[72] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "*A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics*,"*IEEE Int. Conf. Comput. Vision*, 2001, The Institute of Electrical and Electronics Engineers (IEEE), New York City. pp. 416–423.

[73] S. Mignon, B. Galerne, M. Hidane, C. Louchet, and J. Mille, "*Semi-unbalanced regularized optimal transport for image restoration*,"*Euro. Signal Processing Conf*., 2023, The Institute of Electrical and Electronics Engineers (IEEE), New York City. pp. 466–470.

[74] S. Mukherjee, A. Hauptmann, O. Öktem, M. Pereyra, and C.-B. Schönlieb, Learned reconstruction methods with convergence guarantees: A survey of concepts and applications, IEEE Signal Process. Mag. **40** (2023), no. 1, 164–182.

[75] R. Neal, Bayesian learning via stochastic dynamics, Adv. Neural Inf. Proces. Syst. **5** (1992), 475–482.

[76] S. Neumayer and G. Steidl, "*From optimal transport to discrepancy*,"*Handbook of mathematical models and algorithms in computer vision and imaging*, Springer International Publishing, Basel. 2021, pp. 1–36.

[77] D.-P.-L. Nguyen, J. Hertrich, J.-F. Aujol, and Y. Berthoumieu, Image super-resolution with PCA reduced generalized Gaussian mixture models in materials science, Inverse Problems Imaging **17** (2023), no. 6, 1165–1192.

[78] G. Ongie, A. Jalal, C. A. Metzler, R. G. Baraniuk, A. G. Dimakis, and R. Willett, Deep learning techniques for inverse problems in imaging, IEEE J. Selected Areas Informat. Theory **1** (2020), no. 1, 39–56.

[79] S. Osher, Z. Shi, and W. Zhu, Low dimensional manifold model for image processing, SIAM J. Imag. Sci. **10** (2017), no. 4, 1669–1690.

[80] V. Papyan and M. Elad, Multi-scale patch-based image restoration, IEEE Trans. Image Process. **25** (2015), no. 1, 249–261.

[81] S. Parameswaran, C.-A. Deledalle, L. Denis, and T. Q. Nguyen, Accelerating GMM-based patch priors for image restoration: Three ingredients for a 100× speed-up, IEEE Trans. Image Process. **28** (2018), no. 2, 687–698.

[82] S.-J. Park, H. Son, S. Cho, K.-S. Hong, and S. Lee, Srfeat: Single image super-resolution with feature discrimination, Eur. Conf. Comput. Vision Part XVI (2018), 439–455.

[83] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, PyTorch: An imperative style, high-performance deep learning library, Adv. Neural Inf. Proces. Syst. **32** (2019), 8024.

[84] G. Peyré, S. Bougleux, and L. Cohen, Non-local regularization of inverse problems, Eur. Conf. Comput. Vision Part III (2008), 57–68.

[85] J. Prost, A. Houdard, A. Almansa, and N. Papadakis, "*Learning local regularization for variational image restoration*,"*Int. Conf. Scale Space and Variational Methods in Comput. Vision*, Springer International Publishing, Basel. 2021, pp. 358–370.

[86] J. Radon, On the determination of functions from their integral values along certain manifolds, IEEE Trans. Med. Imaging **5** (1986), no. 4, 170–176. https://doi.org/10.1109/TMI.1986.4307775.

[87] A. R. Reibman, R. M. Bell, and S. Gray, "*Quality assessment for super-resolution image enhancement*,"*IEEE Int. Conf. Image Process*. The Institute of Electrical and Electronics Engineers (IEEE), New York City. 2006, pp. 2017–2020.

[88] G. O. Roberts and J. S. Rosenthal, General state space Markov chains and MCMC algorithms, Probab. Surv. **1** (2004), 20–71.

[89] G. O. Roberts and R. L. Tweedie, Exponential convergence of Langevin distributions and their discrete approximations, Bernoulli **2** (1996), no. 4, 341–363.

[90] S. Roth and M. J. Black, Fields of experts: A framework for learning image priors, IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit. **2** (2005), 860–867.

[91] L. I. Rudin, S. Osher, and E. Fatemi, Nonlinear total variation based noise removal algorithms, Phys. D Nonlinear Phenom. **60** (1992), no. 1-4, 259–268.

[92] A. Salmona, V. D. Bortoli, J. Delon, and A. Desolneux, Can push-forward generative models fit multimodal distributions? Adv. Neural Inf. Proces. Syst. **36** (2022), 10766–10779.

[93] F. Santambrogio, *Optimal transport for applied mathematicians calculus of variations, PDEs, and modeling*, Vol **55**, Springer, Basel. 2015.

[94] U. Sara, M. Akter, and M. S. Uddin, Image quality assessment through FSIM, SSIM, MSE and PSNR—A comparative study, J. Comput. Commun. **7** (2019), no. 3, 8–18.

[95] T. Séjourné, G. Peyré, and F.-X. Vialard, "*Unbalanced optimal transport, from theory to numerics*,"*Handbook of Numerical Analysis*, Vol **24**, Elsevier B.V., Amsterdam. 2023, pp. 407–471.

[96] T. R. Shaham, T. Dekel, and T. Michaeli, "*SINGAN: Learning a generative model from a single natural image*,"*IEEE/CVF Int. Conf. Comput. Vision*, The Institute of Electrical and Electronics Engineers (IEEE), New York City. 2019, pp. 4570–4580.

[97] N. Shlezinger, J. Whang, Y. C. Eldar, and A. G. Dimakis, Model-based deep learning, Proc. IEEE, New York, (2023).

[98] A. Shocher, S. Bagon, P. Isola, and M. Irani, "*InGAN: Capturing and retargeting the "DNA" of a natural image*,"*IEEE/CVF Int. Conf. Comput. Vision*, The Institute of Electrical and Electronics Engineers (IEEE), New York City. 2019, pp. 4492–4501.

[99] A. Shocher, N. Cohen, and M. Irani, "Zero-shot" super-resolution using deep internal learning, IEEE Conf. Comput. Vision Pattern Recognit. **31** (2018), 3118–3126.

[100] E. P. Simoncelli and B. A. Olshausen, Natural image statistics and neural representation, Annu. Rev. Neurosci. **24** (2001), no. 1, 1193–1216.

[101] K. Simonyan and A. Zisserman, "*Very deep convolutional networks for large-scale image recognition*,"*Int. Conf. Learn. Represent*., Curran Associates's, Red Hook. 2015.

[102] K. Sohn, H. Lee, and X. Yan, Learning structured output representation using deep conditional generative models, Adv. Neural Inf. Proces. Syst. **28** (2015), 3483–3491.

[103] Y. Song, C. Durkan, I. Murray, and S. Ermon, Maximum likelihood training of score-based diffusion models, Adv. Neural Inf. Proces. Syst. **34** (2021a), 1415–1428.

[104] Y. Song, J. Sohl-Dickstein, D. P. Kingma, A. Kumar, S. Ermon, and B. Poole, "*Score-based generative modeling through stochastic differential equations*,"*Int. Conf. Learn. Represent*., Curran Associates's, Red Hook. 2021b.

[105] Y. Song, C. Yang, Z. Lin, X. Liu, Q. Huang, H. Li, and C.-C. J. Kuo, "*Contextual-based image inpainting: Infer, match, and translate*,"*Euro. Conf. Comput. Vision*, Springer International Publishing, Basel. 2018, pp. 3–19.

[106] B. Sprungk, On the local Lipschitz stability of Bayesian inverse problems, Inverse Problems **36** (2020), no. 5, 055015.
[107] A. M. Stuart, Inverse problems: A Bayesian perspective, Acta Numer. **19** (2010), 451–559.
[108] A. N. Tikhonov, On the solution of ill-posed problems and the method of regularization, Dokl. Akad. Nauk **151** (1963), no. 3, 501–504.
[109] A. Torralba and A. Oliva, Statistics of natural image categories, Netw. Comput. Neural Syst. **14** (2003), no. 3, 391.
[110] S. Vaucher, P. Unifantowicz, C. Ricard, L. Dubois, M. Kuball, J.-M. Catala-Civera, D. Bernard, M. Stampanoni, and R. Nicula, On-line tools for microscopic and macroscopic monitoring of microwave processing, Phys. B Condens. Matter **398** (2007), no. 2, 191–195.
[111] Y.-Q. Wang and J.-M. Morel, SURE guided Gaussian mixture image denoising, SIAM J. Imag. Sci. **6** (2013), no. 2, 999–1034.
[112] M. Welling and Y. W. Teh, "*Bayesian learning via stochastic gradient Langevin dynamics*,"*Int. Conf. Mach. Learn.*, Curran Associates's Red Hook. 2011, pp. 681–688.
[113] C. Winkler, D. Worrall, E. Hoogeboom, and M. Welling. Learning likelihoods with conditional normalizing flows. 2019. arXiv preprint, arXiv:1912.00042.
[114] H. Wu, J. Köhler, and F. Noé, Stochastic normalizing flows, Adv. Neural Inf. Proces. Syst. **33** (2020), 5933–5944.
[115] Q. Yang, P. Yan, M. K. Kalra, and G. Wang. CT image denoising with perceptive deep neural networks. 2017. arXiv preprint, arXiv:1702.07019.
[116] Q. Yu, G. Cao, H. Shi, Y. Zhang, and P. Fu, EPLL image denoising with multi-feature dictionaries, Digit. Signal Process. **137** (2023), 104019.
[117] M. Zach, T. Pock, E. Kobler, and A. Chambolle, "*Explicit diffusion of Gaussian mixture model based image priors*,"*Int. Conf. Scale Space and Variational Methods in Computer Vision*, Springer International Publishing, Basel, 2023, pp. 3–15.
[118] L. Zhang, L. Zhang, X. Mou, and D. Zhang, FSIM: A feature similarity index for image quality assessment, IEEE Trans. Image Process. **20** (2011), no. 8, 2378–2386.
[119] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, The unreasonable effectiveness of deep features as a perceptual metric, IEEE Conf. Comput. Vision Pattern Recognit. **31** (2018), 586–595.
[120] M. Zontak and M. Irani, Internal statistics of a single natural image, IEEE Conf. Comput. Vision Pattern Recognit. (2011), 977–984.
[121] D. Zoran and Y. Weiss, "*From learning models of natural image patches to whole image restoration*,"*Int. Conf. Compu. Vision*, The Institute of Electrical and Electronics Engineers (IEEE), New York, 2011, pp. 479–486.

How to cite this article: M. Piening, F. Altekrüger, J. Hertrich, P. Hagemann, A. Walther, and G. Steidl, *Learning from small data sets: Patch-based regularizers in inverse problems for image reconstruction*, GAMM-Mitteilungen. (2024), e202470002. https://doi.org/10.1002/gamm.202470002