Stochastic Image Denoising by Sampling from the Posterior Distribution

Bahjat Kawar Gregory Vaksman Michael Elad
Computer Science Department, The Technion - Israel Institute of Technology
{bahjat.kawar,grishav,elad}@cs.technion.ac.il

Abstract

Image denoising is a well-known and well studied problem, commonly targeting a minimization of the mean squared error (MSE) between the outcome and the original image. Unfortunately, especially for severe noise levels, such Minimum MSE (MMSE) solutions may lead to blurry output images. In this work we propose a novel stochastic denoising approach that produces viable and high perceptual quality results, while maintaining a small MSE. Our method employs Langevin dynamics that relies on a repeated application of any given MMSE denoiser, obtaining the reconstructed image by effectively sampling from the posterior distribution. Due to its stochasticity, the proposed algorithm can produce a variety of high-quality outputs for a given noisy input, all shown to be legitimate denoising results. In addition, we present an extension of our algorithm for handling the inpainting problem, recovering missing pixels while removing noise from partially given data.

1. Introduction

This work focuses on the image denoising task, a well-known and well-studied problem in the field of image processing. Various successful algorithms, both classically oriented and deep learning based, were proposed over the years for handling this task, such as NLM [13], KSVD [19], BM3D [17], EPLL [59], WNNM [21], TNRD [16], DnCNN [53], NLRN [29] and others [52, 36, 4, 45, 25, 54, 42, 44, 26, 57]. Nowadays, supervised deep learning-based schemes lead the image denoising field, showing state-of-the-art (SoTA) performance [53, 29, 57].

Common to these and many other algorithms is the fact that they minimize the expected distance, most notably the L_2 metric, between the original and the reconstructed images. This approach leads to a minimum mean squared error (MMSE) estimator. Unfortunately, high performance in terms of MSE does not necessarily mean good perceptual quality [47]. Since denoising is an ill-posed problem (*i.e*. a given input may have multiple correct solutions), the MMSE solutions tend to average these possible correct outcomes. In a high noise scenario, such an averaging strategy often leads to output images with blurry edges and unclear fine details. Many alternative distance measures to the MSE have been suggested, including SSIM [48], MS-SSIM [49], IFC [38], VIF [37], VSNR [14], and FSIM [55]. However, changing the distance measure might not solve the problem. The authors of [8, 9] have shown that there is an inherent contradiction between any mean distortion measure and perceptual quality. Due to this so-called "perception-distortion" trade-off, an image that minimizes the mean distance in any metric will necessarily suffer from a degradation in perceptual quality.

Figure 1. Top row, left to right: original image, noisy image with $\sigma = 0.406$ (pixel values are in the range $[0, 1]$), the denoised results using an MMSE denoiser,[1] and BM3D [17]. Bottom row: several outputs of our algorithm using the same MMSE denoiser.

What could be the remedy to the above-described problem? While maintaining the Bayesian point of view, denoising could still leverage the posterior distribution of the unknown image given the measurements, but avoid the averaging effect. Seeking the highest peak of the posterior distribution or sampling from it, both seem as good strategies for getting high-perceptual quality outcomes.

And indeed, many model-based classically-oriented algorithms seem to have chosen to apply a maximum a posteriori (MAP) estimator instead of MMSE (*e.g*. [19, 59]). MAP or closely related prior-based approaches are also ap-

[1]The MMSE denoiser used in all our experiments is based on NCSNv2 [40]. See subsection 2.2.2 for more details.

plied more recently in deep learning based methods [43, 20, 7, 58]. However, these methods do not attempt to recover a highly probable natural-looking solution to the problem, but rather attempt to leverage a prior assumption on the image distribution in order to improve MSE performance. This is evident in the fact that the main evaluation metric for these methods is almost always peak signal to noise ratio (PSNR). In fact, many of these methods incorporate certain techniques in addition to MAP estimation, such as early stopping, patch averaging, or extra regularization, all done in order to achieve better PSNR performance, getting as close as possible to MMSE performance.

An alternative strategy to MAP is a sampler from the posterior. Recently, generative adversarial networks (GANs) have achieved success in generating realistically looking images (*e.g*. [12, 24]), effectively sampling from the distribution of images. A GAN can serve our denoising task in one of two ways: either being trained to sample from the posterior directly, or by inverting its pre-trained generator. The first path refers to a variant of a conditional GAN, an approach that encounters difficulties in training, as exposed in [2, 3]. The inversion option is appealing [27, 1], but relies on an adversarial training, which is usually unstable, and there are currently no theoretical guarantees that its results are valid samples from the posterior distribution.

In this paper, we take a completely different approach towards high perceptual quality denoising that does not rely on GANs. We draw inspiration from an interesting line of work reported in [39, 40, 22, 41] that develops an alternative method for generating images. The authors of [39] propose an *annealed Langevin dynamics* algorithm in order to sample from the prior image distribution. This requires knowledge of the *(Stein) score function*, which is the gradient of the log of the prior. The work in [23] introduces an extremely valuable link between this score function and MMSE denoisers, showing how to synthesize images and solve a special class of inverse problems by leveraging a given MMSE denoiser.

In our work we propose a novel approach for handling the image denoising task by building on the above. Our method produces sharp output images bypassing the typical denoisers' blurriness problem. Instead of minimizing MSE, our *stochastic denoiser* samples its output from the posterior distribution given a corrupted image. The proposed algorithm stochastically picks a clean image consistent with the corrupted input one, instead of producing a single averaged output. Similar ideas arise in recently published papers [5, 31], which suggest that methods solving super resolution should not be deterministic, but rather allow for many possible outcomes. In addition, stochastic denoising has been suggested in [50], but their method utilizes a far more complicated posterior sampling, and they use it for estimating an MMSE denoiser. In contrast, our algorithm samples directly from the posterior, achieving better perceptual quality.

For implementing sampling from the posterior, we formulate a score function that corresponds to the posterior, employ an MMSE denoiser to assist in evaluating it, and harness the annealed Langevin dynamics for drawing samples from this distribution. For any noisy input image, the proposed algorithm can produce a variety of viable outputs. Besides being sharp and natural-looking, images produced by the proposed stochastic denoiser are close to the MMSE solution in terms of PSNR, and visually similar to the true clean image. In fact, our work shows that all reconstructed images are valid outcomes of the denoising procedure, *i.e*. the difference between any pair of noisy and reconstructed images is statistically fitted to an additive white Gaussian noise with the appropriate variance.

In addition, we introduce an extension of the stochastic denoising scheme for solving the noisy inpainting problem, in which the observed image is incomplete and contaminated by noise. As in the denoising case, the inpainting scheme can produce a variety of valid yet different outputs for any input image.

Instead of working with a specific model architecture, our denoising and inpainting schemes utilize any denoiser trained/designed for minimizing the MSE on a set of noise levels. Such high-performance denoisers are widely available due to the incredible advances in image denoising achieved in the past two decades[2]. Thus, our recovery schemes do not require any specific constraints on the model architecture or retraining of the MMSE denoiser. The only requirement is the ability to produce high-PSNR outputs for a set of noise levels. To summarize, this paper has two main contributions:

- We introduce a novel stochastic approach for the image denoising problem that leads to sharp and natural-looking reconstructions. Instead of reducing the restoration error, we propose to pick a probable solution by effectively sampling from the posterior distribution.

- We present stochastic algorithms for solving both the image denoising and inpainting problems. For any corrupted input, these algorithms can produce a wide range of outputs where each is a possible valid solution of the problem.

[2]Indeed, the extremely well-performing MMSE denoisers available today have led researchers to question whether we are nearing the optimal achievable noise reduction [15, 28].

2. Proposed Method: Foundations

2.1. Sampling from the prior distribution

One way to generate samples from a probability distribution $p(x)$, is using the Markov Chain Monte Carlo (MCMC) method with the Langevin transition rule [6, 34]

$$x_{t+1} = x_t + \alpha \nabla_x \log p(x_t) + \sqrt{2\alpha} z_t \ , \qquad (1)$$

where $z_t \sim \mathcal{N}(0, I)$ and α some appropriate small constant. The expression $\nabla_x \log p(x)$ is known as the *score function* [39] and is usually denoted as $s(x)$. The role of z_t is to allow stochastic sampling, avoiding a collapse to a maximum of the distribution. Initialized randomly, after a sufficiently large number of iterations, and under some conditions, this process converges to a sampling from the desired distribution $p(x)$ [34]. The work reported in [39] extends the aforementioned algorithm to *annealed Langevin dynamics*, which is handy for generating images from the implied prior distribution $p(x)$. The algorithm proposed by [39] works as follows: Initialized with a random image, it follows the direction of the score function in each step, as in Equation 1. The score function is defined as $\nabla_{\tilde{x}} \log p(\tilde{x})$ where $\tilde{x} = x + z$ and $z \sim \mathcal{N}\left(0, \sigma^2 I\right)$ for different values of σ. Their method starts by using score functions corresponding to a high σ, and gradually lowers it until $p(\tilde{x})$ is indistinguishable from the true data prior $p(x)$. This way, the algorithm flows the initial random image to ones with a higher prior probability, meaning that the output is a natural-looking image.

2.2. Sampling from the posterior distribution

We start by formulating our denoising task: Given a noisy input image $y = x + n$ where $x \sim p(x)$ is the true clean image and $n \sim \mathcal{N}\left(0, \sigma_0^2 I\right)$ is a random white Gaussian noise with a known variance, we attempt to recover x.[3] This task might have multiple possible solutions for x, and we would like to output one of them. While $p(x)$ is unknown, we assume that we have access to an MMSE denoiser operating on images from $p(x)$. We propose to recover x by sampling from the posterior distribution given the noisy input image, *i.e.*, $p(x \mid y)$.

Our proposed approach is an adaptation of the *annealed Langevin dynamics* algorithm [39] for our aforementioned task. *Annealed Langevin dynamics* produces samples from $p(x)$ by means of the score function $\nabla_{\tilde{x}} \log p(\tilde{x})$ where $\tilde{x} = x + z$ and $z \sim \mathcal{N}\left(0, \sigma^2 I\right)$ for different values of σ. In order to adapt it to our task, we need to estimate the score function of the posterior $\nabla_{\tilde{x}} \log p(\tilde{x} \mid y)$. Note that the work reported in [23, 41] formulates score functions for posteriors for several inverse problems, and samples from these posteriors using Langevin dynamics. However, the problems treated are limited to noise free cases.

[3] Throughout this work we consider a Gaussian noise corruption, which provides a good approximation for many use cases [11].

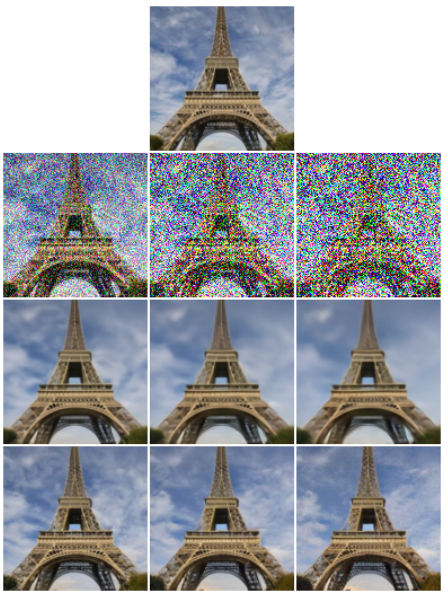

Figure 2. From top to bottom: original LSUN-tower image, noisy versions (σ_0 from left to right: 0.198, 0.403, 0.606), MMSE denoiser outputs, and instances of our algorithm's output.

In the following we derive a formula for obtaining $\nabla_{\tilde{x}} \log p(\tilde{x} \mid y)$, and then present the relation between this score function and the MMSE denoiser. In section 3 we present the stochastic denoiser algorithm for sampling from $p(x \mid y)$, which is based on the *annealed Langevin dynamics* algorithm [39]. Section 3 also includes an extension of our algorithm for handling the inpainting problem.

2.2.1. The score function of the posterior distribution

We fix a sequence of noise levels $\{\sigma_i\}_{i=0}^{L+1}$ such that $\sigma_0 > \sigma_1 > \ \cdots \ > \sigma_L > \sigma_{L+1} = 0$, where σ_0 is the noise level in y and σ_{L+1} is simply zero. We consider the process of adding white Gaussian noise with standard deviation σ_0 to x as a gradual sequence of noise additions $\{\tilde{x}_i\}_{i=1}^{L}$ starting from $\tilde{x}_L$ down to $\tilde{x}_0$:

$$\begin{aligned} \tilde{x}_L &= x + z_L \\ \tilde{x}_{L-1} &= \tilde{x}_L + z_{L-1} \\ \tilde{x}_{L-2} &= \tilde{x}_{L-1} + z_{L-2} \\ &\vdots \\ \tilde{x}_1 &= \tilde{x}_2 + z_1 \\ y = \tilde{x}_0 &= \tilde{x}_1 + z_0 \end{aligned} \qquad (2)$$

where $z_i \sim \mathcal{N}\left(0, \left(\sigma_i^2 - \sigma_{i+1}^2\right) I\right)$ for $0 \leq i \leq L$. Note that from the above we conclude that

$$y = \tilde{x}_0 = x + \sum_{i=0}^{L} z_i, \qquad (3)$$

where $\sum_{i=0}^{L} z_i \sim \mathcal{N}\left(0, \sigma_0^2 I\right)$, because a sum of independent Gaussian random variables is a Gaussian random variable with variance equal to the sum of their variances. This matches our original definition of y in the denoising task. We also notice that

$$y - \tilde{x}_i = \sum_{j=0}^{i-1} z_j, \tag{4}$$

where $\sum_{j=0}^{i-1} z_j \sim \mathcal{N}\left(0, \left(\sigma_0^2 - \sigma_i^2\right) I\right)$.

In the next calculations, which are valid for every i, we refer to $\tilde{x}_i$ as $\tilde{x}$ for simplicity. We move on to calculate $\nabla_{\tilde{x}} \log p\left(\tilde{x} \mid y\right)$ using the Bayes rule,

$$\begin{aligned}\nabla_{\tilde{x}} \log p\left(\tilde{x} \mid y\right) &= \nabla_{\tilde{x}} \log \left[\left(\frac{1}{p\left(y\right)}\right) p\left(y \mid \tilde{x}\right) p\left(\tilde{x}\right)\right] \\ &= \nabla_{\tilde{x}} \left[\log\left(\frac{1}{p\left(y\right)}\right) + \log p\left(y \mid \tilde{x}\right) + \log p\left(\tilde{x}\right)\right].\end{aligned}$$

Since y is a fixed observation that does not depend on $\tilde{x}$, the gradient of the first term vanishes, resulting in

$$\nabla_{\tilde{x}} \log p\left(\tilde{x} \mid y\right) = \nabla_{\tilde{x}} \log p\left(y \mid \tilde{x}\right) + \nabla_{\tilde{x}} \log p\left(\tilde{x}\right). \tag{5}$$

In order to calculate $\nabla_{\tilde{x}} \log p\left(y \mid \tilde{x}\right)$, we recall Equation 4 and obtain

$$\begin{aligned}\nabla_{\tilde{x}} \log p\left(y \mid \tilde{x}\right) &= \nabla_{\tilde{x}} \log p_{Y-\tilde{X}}\left(y - \tilde{x} \mid \tilde{x}\right) \\ &= \nabla_{\tilde{x}} \log \left[\frac{1}{\sqrt{2\pi\left(\sigma_0^2 - \sigma_i^2\right)}} \exp\left[-\frac{1}{2}\frac{\|y - \tilde{x}\|^2}{\left(\sigma_0^2 - \sigma_i^2\right)}\right]\right].\end{aligned}$$

Calculating this results in

$$\nabla_{\tilde{x}} \log p\left(y \mid \tilde{x}\right) = \frac{y - \tilde{x}}{\sigma_0^2 - \sigma_i^2}, \tag{6}$$

which when combined with Equation 5 gives

$$\nabla_{\tilde{x}} \log p\left(\tilde{x} \mid y\right) = \nabla_{\tilde{x}} \log p\left(\tilde{x}\right) + \frac{y - \tilde{x}}{\sigma_0^2 - \sigma_i^2}. \tag{7}$$

The first term is the same one used in [39], while the second can be computed easily. Therefore, we have obtained a tractable method for estimating the score function of the posterior distribution given the noisy image.

2.2.2. Estimating the score using an MMSE denoiser

A major step forward is provided in [23], exposing the following intricate and fascinating connection between the score function and MMSE denoisers:

$$\nabla_{\tilde{x}} \log p\left(\tilde{x}\right) = \frac{\hat{x}\left(\tilde{x}\right) - \tilde{x}}{\sigma_i^2}, \tag{8}$$

where $\hat{x}\left(\tilde{x}\right) = \mathbb{E}\left[x \mid \tilde{x}\right]$ is defined as the MMSE denoiser. This relation suggests that a network trained to estimate the score function (NCSNv2 [40]) can be interpreted as a denoiser estimating MMSE. Indeed, we have utilized this network as such, and it performs very well, as can be seen in the MMSE denoiser results presented in Figure 1 and Table 1.

Likewise, we can utilize in our scheme any denoiser trained/designed to minimize MSE (for various noise levels σ) in order to estimate the score function. A variety of such denoisers exist, each implicitly defining and serving a different prior distribution. Adopting a broader view, the fact that MMSE denoisers can be leveraged for different tasks is a fascinating phenomenon, which has already been exposed in recent work in different contexts [46, 35, 23].

Algorithm 1: Stochastic image denoiser

Input: $\{\sigma_i\}_{i=1}^{L}$, ϵ, T, y, σ_0
Initialize $x_0 \leftarrow y$
for $i \leftarrow 1$ *to* L **do**
  $\alpha_i \leftarrow \epsilon \cdot \sigma_i^2 / \sigma_L^2$
  for $t \leftarrow 1$ *to* T **do**
    Draw $z_t \sim \mathcal{N}\left(0, I\right)$
    $\Delta_t \leftarrow s\left(x_{t-1}, \sigma_i\right) + \left(y - x_{t-1}\right) / \left(\sigma_0^2 - \sigma_i^2\right)$
    $x_t \leftarrow x_{t-1} + \alpha_i \Delta_t + \sqrt{2\alpha_i} z_t$
  end
  $x_0 \leftarrow x_T$
end
Output: x_T

3. Proposed Algorithms

3.1. Stochastic denoising

In order to clean a given noisy image y, we propose to gradually reverse the noise addition process described in subsection 2.2.1. We do so stochastically, using a variation on the *annealed Langevin dynamics* [39] sampling algorithm. We denote by $s\left(x, \sigma\right)$ a function that estimates the score function of the prior, and we present our method in Algorithm 1. The algorithm follows the direction of the conditional score function, with a step size of α_i that is gradually tuned down as the noise level decreases (see [39]).

Our algorithm is initialized with the given noisy image y, and it follows the *annealed Langevin dynamics* scheme using the score function of the posterior as presented in Equation 7. This allows it to effectively sample from the posterior distribution $p\left(\tilde{x}_L \mid y\right) \approx p\left(x \mid y\right)$, and thus be considered a *stochastic image denoiser*.

3.2. Image inpainting

In the noisy inpainting problem, our observation is only a known subset M of the pixels of the noisy image $y = x + n$. We denote the pixels M of any image w as w^M and the re-

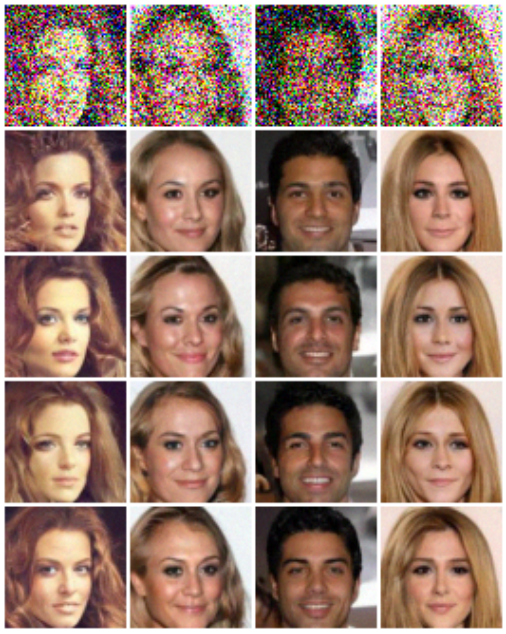

Figure 3. Top row: Several given noisy images ($\sigma_0 = 0.406$). Second row and below: Various outputs of our algorithm corresponding to each noisy image.

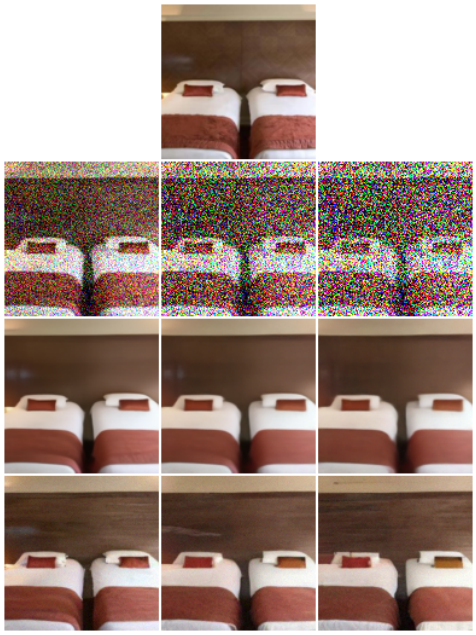

Figure 4. From top to bottom: original LSUN-bedroom image, noisy versions of it (σ_0 from left to right: 0.198, 0.403, 0.606), MMSE denoiser outputs, and instances of our algorithm's output.

maining pixels as w^R. With these notations, the visible observation is y^M. However, our approach remains the same as in the denoising problem, aiming to sample from the posterior distribution $p\left(x \mid y^M\right)$.

As our observation is incomplete, we initialize the recovery algorithm with an i.i.d. Gaussian noise image with a very strong variance (as in [40]), and proceed from there by sampling from the posterior distribution given y^M. More formally, we use a fixed sequence of noise levels $\{\sigma_i\}_{i=-K}^{L+1}$ such that $\sigma_{-K} > \sigma_{-(K-1)} > \cdots > \sigma_0 > \cdots > \sigma_L > \sigma_{L+1} = 0$, where σ_0 is the noise level of the observation. In calculating the score function $\nabla_{\tilde{x}} \log p\left(\tilde{x} \mid y^M\right)$, we divide our analysis into two cases, $i < 0$ in which the noise we handle is stronger than σ_0, and $i > 0$, in which the noise bypasses σ_0 and gradually decreases towards zero. We start our derivations with the second case, as it is simpler:

For the case where $i > 0$, we recall Equation 4 and deduce that

$$\begin{aligned}\nabla_{\tilde{x}} \log p\left(y^M \mid \tilde{x}\right) &= \nabla_{\tilde{x}} \log p_{Y^M - \tilde{X}^M}\left(y^M - \tilde{x}^M \mid \tilde{x}\right) \\ &= \nabla_{\tilde{x}} \log \left[\frac{1}{\sqrt{2\pi\left(\sigma_0^2 - \sigma_i^2\right)}} \exp\left[-\frac{1}{2}\frac{\left\|y^M - \tilde{x}^M\right\|^2}{\left(\sigma_0^2 - \sigma_i^2\right)}\right]\right].\end{aligned}$$

Calculating this results in

$$\begin{cases} \nabla_{\tilde{x}^M} \log p\left(y^M \mid \tilde{x}\right) = \frac{y^M - \tilde{x}^M}{\sigma_0^2 - \sigma_i^2} \\ \nabla_{\tilde{x}^R} \log p\left(y^M \mid \tilde{x}\right) = 0 \end{cases}, \tag{9}$$

which when combined with Equation 5 gives the score function to use:

$$\begin{cases} \nabla_{\tilde{x}^M} \log p\left(\tilde{x} \mid y^M\right) = \left[\nabla_{\tilde{x}} \log p\left(\tilde{x}\right)\right]^M + \frac{y^M - \tilde{x}^M}{\sigma_0^2 - \sigma_i^2} \\ \nabla_{\tilde{x}^R} \log p\left(\tilde{x} \mid y^M\right) = \left[\nabla_{\tilde{x}} \log p\left(\tilde{x}\right)\right]^R. \end{cases} \tag{10}$$

Since the noise level in this case is below σ_0, we effectively obtain the same score function as in the denoising task for the observed pixels M. As for the remaining pixels, R, the observation does not add any information, leaving us to rely only on the prior distribution.

For the other case where $i < 0$, we recall the definition of the conditional distribution,

$$\begin{aligned} p\left(\tilde{x}^R \mid \tilde{x}^M, y^M\right) &= \frac{p\left(\tilde{x}^M, \tilde{x}^R \mid y^M\right)}{p\left(\tilde{x}^M \mid y^M\right)} \\ \Rightarrow p\left(\tilde{x}^R \mid \tilde{x}^M, y^M\right) p\left(\tilde{x}^M \mid y^M\right) &= p\left(\tilde{x}^M, \tilde{x}^R \mid y^M\right) \end{aligned}$$

With that in mind, we present the following calculation of the log of the posterior function:

$$\begin{aligned} \log p\left(\tilde{x} \mid y^M\right) &= \log p\left(\tilde{x}^M, \tilde{x}^R \mid y^M\right) \\ &= \log\left[p\left(\tilde{x}^R \mid \tilde{x}^M, y^M\right) p\left(\tilde{x}^M \mid y^M\right)\right] \\ &= \log p\left(\tilde{x}^R \mid \tilde{x}^M, y^M\right) + \log p\left(\tilde{x}^M \mid y^M\right). \end{aligned} \tag{11}$$

Referring to the second term, we can conclude that, similar to Equation 4, we have $\left(\tilde{x}^M - y^M\right) \sim \mathcal{N}\left(0, \left(\sigma_i^2 - \sigma_0^2\right) I\right)$. This difference is independent of $\tilde{x}^R$, and thus, $p\left(\tilde{x}^R \mid \tilde{x}^M, y^M\right)$ can be expressed as either $p\left(\tilde{x}^R \mid \tilde{x}^M\right)$

Dataset	σ_0	BM3D	MMSE	Ours	Ratio
CelebA	0.100	30.18	32.58	29.39	2.07
	0.203	25.43	29.08	26.28	1.90
	0.406	19.73	25.78	23.24	1.79
	0.607	16.75	23.93	21.52	1.73
	0.702	15.87	23.27	20.90	1.72
LSUN bedroom	0.198	27.19	29.95	27.11	1.91
	0.403	21.31	26.50	24.00	1.78
	0.606	18.17	24.55	22.22	1.72

Table 1. Average PSNR results using 64 CelebA images and 64 LSUN images, including BM3D [17] as a baseline. The last column shows the MSE ratio between the MMSE and ours.

or $p\left(\tilde{x}^R \mid y^M\right)$. Equation 11 can be derived w.r.t. $\tilde{x}^M$:

$$\begin{aligned}
&\nabla_{\tilde{x}^M} \log p\left(\tilde{x} \mid y^M\right) \\
&\quad = \nabla_{\tilde{x}^M}\left[\log p\left(\tilde{x}^R \mid y^M\right) + \log p_{\tilde{X}^M - Y^M}\left(\tilde{x}^M - y^M \mid y^M\right)\right] \\
&\quad \approx \nabla_{\tilde{x}^M} \log p_{\tilde{X}^M - Y^M}\left(\tilde{x}^M - y^M \mid y^M\right).
\end{aligned}$$

More details on this approximation are in the supplementary material. This results in

$$\nabla_{\tilde{x}^M} \log p\left(\tilde{x} \mid y^M\right) = \frac{y^M - \tilde{x}^M}{\sigma_i^2 - \sigma_0^2}. \tag{12}$$

Deriving Equation 11 w.r.t. $\tilde{x}^R$ yields:

$$\begin{aligned}
&\nabla_{\tilde{x}^R} \log p\left(\tilde{x} \mid y^M\right) \\
&\quad = \nabla_{\tilde{x}^R}\left[\log p\left(\tilde{x}^R \mid \tilde{x}^M\right) + \log p\left(\tilde{x}^M \mid y^M\right)\right] \\
&\quad = \nabla_{\tilde{x}^R} \log p\left(\tilde{x}^R \mid \tilde{x}^M\right) = \nabla_{\tilde{x}^R} \log\left[\frac{p\left(\tilde{x}^R, \tilde{x}^M\right)}{p\left(\tilde{x}^M\right)}\right] \\
&\quad = \nabla_{\tilde{x}^R}\left[\log p\left(\tilde{x}\right) - \log p\left(\tilde{x}^M\right)\right] \\
&\quad = \nabla_{\tilde{x}^R} \log p\left(\tilde{x}\right) = \left[\nabla_{\tilde{x}} \log p\left(\tilde{x}\right)\right]^R,
\end{aligned}$$

resulting in

$$\nabla_{\tilde{x}^R} \log p\left(\tilde{x} \mid y^M\right) = \left[\nabla_{\tilde{x}} \log p\left(\tilde{x}\right)\right]^R. \tag{13}$$

As the noise level in this case is above σ_0, the score function for the known pixels M points to the direction of the observation y^M, which can be considered a denoised version of $\tilde{x}^M$. For the remaining pixels, R, the score function remains the same as in the previous case.

To conclude, by using an estimator for $\nabla_{\tilde{x}} \log p\left(\tilde{x}\right)$ and combining equations 10, 12, and 13, we obtain a tractable method for estimating the score function of the posterior distribution for the inpainting problem. By using this and starting Algorithm 1 with a very strong noise $\sigma_{-K} \gg \sigma_0$, we obtain a path towards solving the noisy inpainting problem, as presented in Algorithm 2.

Algorithm 2: Inpainting algorithm

Input: $\{\sigma_i\}_{i=-K}^{L}$, ϵ, T, y^M
Initialize x_0 with random noise
for $i \leftarrow -K$ *to* -1 **do**
  $\alpha_i \leftarrow \epsilon \cdot \sigma_i^2 / \sigma_L^2$
  for $t \leftarrow 1$ *to* T **do**
    Draw $z_t \sim \mathcal{N}(0, I)$
    $\Delta_t^M \leftarrow \left(y^M - x_{t-1}^M\right) / \left(\sigma_i^2 - \sigma_0^2\right)$
    $\Delta_t^R \leftarrow \left[s\left(x_{t-1}, \sigma_i\right)\right]^R$
    $x_t \leftarrow x_{t-1} + \alpha_i \Delta_t + \sqrt{2\alpha_i} z_t$
  end
  $x_0 \leftarrow x_T$
end
for $i \leftarrow 1$ *to* L **do**
  $\alpha_i \leftarrow \epsilon \cdot \sigma_i^2 / \sigma_L^2$
  for $t \leftarrow 1$ *to* T **do**
    Draw $z_t \sim \mathcal{N}(0, I)$
    $\Delta_t^M \leftarrow \left[s\left(x_{t-1}, \sigma_i\right)\right]^M$
    $\Delta_t^M \leftarrow \Delta_t^M + \left(y^M - x_{t-1}^M\right) / \left(\sigma_0^2 - \sigma_i^2\right)$
    $\Delta_t^R \leftarrow \left[s\left(x_{t-1}, \sigma_i\right)\right]^R$
    $x_t \leftarrow x_{t-1} + \alpha_i \Delta_t + \sqrt{2\alpha_i} z_t$
  end
  $x_0 \leftarrow x_T$
end
Output: x_T

4. Experimental Results

4.1. Denoising experiments

As our algorithm extends the work reported in [39] and [40], it is natural to use their denoiser network, named the noise conditional score network version 2 (NCSNv2) [40]. As is customary in the image synthesis literature, this network was trained on a specific class of images rather than generic natural ones. Previous work [32, 33] have also shown that denoising can benefit from training on a narrow class of images.

We perform experiments using the NCSNv2 network trained separately on CelebA [30], FFHQ [24], and the bedroom and tower categories in LSUN [51]. CelebA images are center cropped to 140×140 pixels, then resized to 64×64 pixels, FFHQ images are resized to 256×256 pixels, and LSUN images are center cropped and resized to 128×128 pixels, exactly as in [40]. We do not change the hyperparameters reported in [40], as they work well for our tasks. For CelebA experiments we pick $L = 127, 166, 204, 226, 234$ for $\sigma_0 = 0.100, 0.203, 0.406, 0.607, 0.702$ respectively. For FFHQ experiments we pick $L = 663, 816, 906$ for $\sigma_0 = 0.200, 0.400, 0.602$ respectively. For

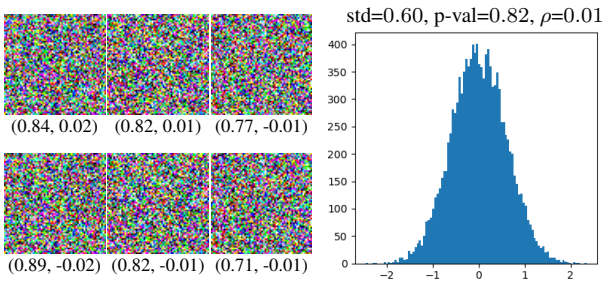

Figure 5. Left: Residual images, p-values and ρ values on CelebA with $\sigma_0 = 0.607$ for 3 different images. The standard deviation is 0.59 or 0.6 in all images. The top row shows our method's residuals, and the bottom one shows the MMSE denoiser's residuals. Right: A histogram for a residual image from our algorithm.

LSUN experiments we pick $L = 330, 408, 453$ for $\sigma_0 = 0.198, 0.403, 0.606$ respectively. We note that in each of our experiments we use the same pre-trained model for both the MMSE denoiser and in our algorithm.

As can be seen in Figure 2, 4, and 6, our stochatic denoising method achieves sharp and real-looking results, regardless of the noise level in the input image, whereas the MMSE denoiser suffers from more severe averaging artefacts as the noise level increases. The results' sharpness is also preserved across different stochastic variations, as can be seen in Figure 1, 3, and in the supplementary material.

We evaluate the perceptual quality of the results using LPIPS [56], in which our model performs significantly better than the MMSE denosier, as shown in the supplementary material. We also asses the similarity to the original clean image using the MSE metric (or its equivalent PSNR). While the MSE measure has clear drawbacks [47], and our algorithm inherently achieves poorer results than an MMSE denoiser, we still find value in reporting such results. It was proven in [8, 9] that we do not need to sacrifice more than a factor of 2 in MSE in order to achieve perfect perceptual quality, which serves as a good baseline for us to evaluate our results. As can be seen in Table 1, the aforementioned ratio is comfortably below 2 in all experiments but one.

4.2. Assesing the estimated noise

We now turn to show that all outputs of the presented algorithm are viable denoising results. A sample from $p(x \mid y)$ should both look real and fulfill the condition $(y - \hat{x}) \sim \mathcal{N}\left(0, \sigma_0^2 I\right)$. The latter is also a criterion for claiming that a given algorithm is a denoiser, as it suggests that the content removed from its input is indeed noise-like. MMSE denoisers, for example, fulfill this criterion.

In order to empirically test whether our algorithm is a stochastic image denoiser as we claim, we analyze the estimated noise – the difference between its input and output, as visualized in Figure 5. Our analysis includes three tests: for whiteness, for noise energy, and for the distribution. First,

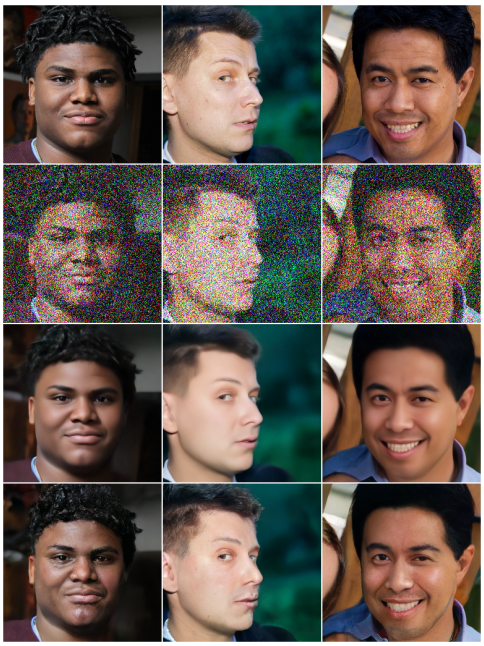

Figure 6. From top to bottom: original 256×256 FFHQ images, noisy versions with $\sigma_0 = 0.4$, and our algorithm's outputs.

we calculate Pearson's correlation coefficient (ρ) among adjacent pixels in all 8 directions, take the one with the maximum absolute value, and if it is sufficiently close to zero, we conclude that the noise is uncorrelated. We proceed by performing D'Agostino and Pearson's test of normality [18] in order to determine whether the difference is normally distributed. For a confidence level of 95%, we conclude that the tested signal is indeed Gaussian if the p-value is greater than 0.05. We conclude by evaluating the empirical standard deviation of the difference and comparing it to σ_0.

We perform these tests on several output images for each noisy input and different noise levels. In almost all of our tests, $|\rho|$ is smaller than 0.02, the p-values are comfortably above 0.05, often reaching more than 0.9, and the standard deviations match the input noise level σ_0 almost perfectly. We show one of the residual histograms in Figure 5, and defer the rest to the supplementary material. Based on these observations, we conclude that our sampled images can be regarded as viable stochastic denoising results. Figure 7 shows the intermediate images obtained along our algorithm, showing a gradual denoising effect, while preserving and even synthesizing details.

4.3. Inpainting experiments

Following the calculations shown in subsection 3.2, we adapt our stochastic denoiser algorithm for solving the noisy inpainting task. As in subsection 4.1, we utilize the NCSNv2 [40] network for estimating the score function of the prior distribution, and we perform experiments on the

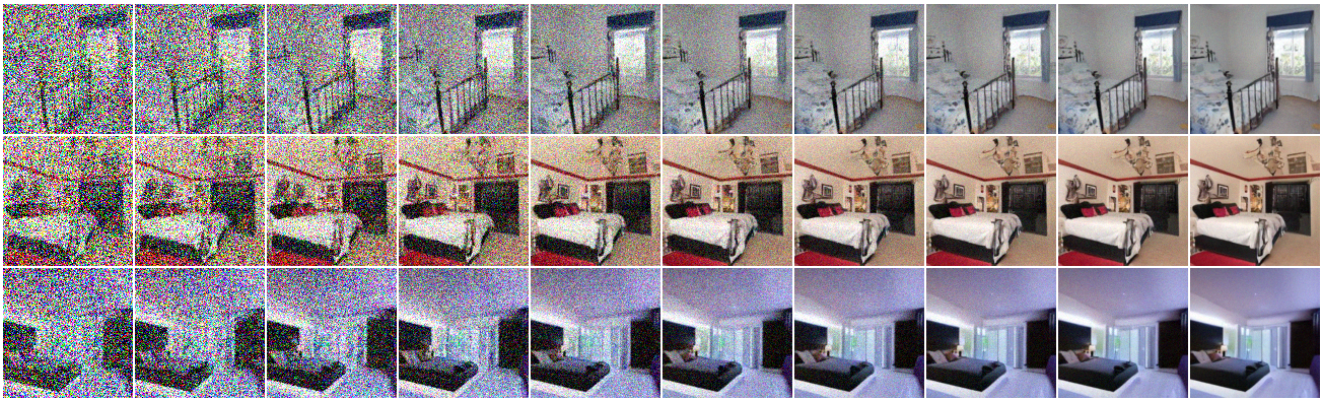

Figure 7. Intermediate results of Algorithm 1 on LSUN-bedroom images with $\sigma_0 = 0.606$.

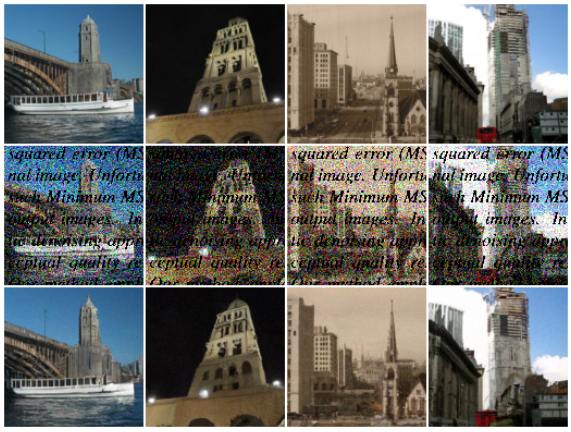

Figure 8. From top to bottom: original LSUN-tower images, the observations with a text overlay and additive noise ($\sigma_0 = 0.198$), and our inpainting algorithm's outputs.

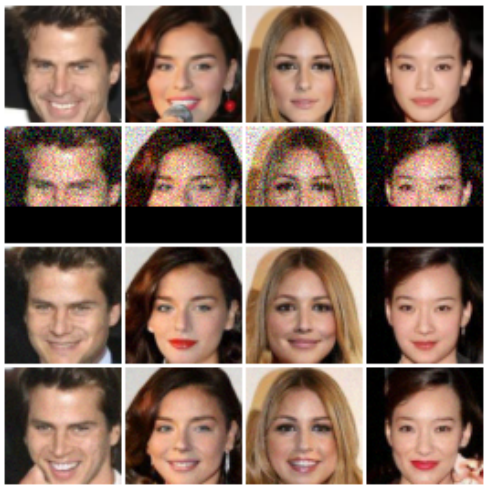

Figure 9. From top to bottom: original CelebA images, the observations with 20 missing rows and additive noise ($\sigma_0 = 0.1$), and two outputs of our inpainting algorithm.

CelebA [30] and LSUN [51] datasets. Here as well we do not change the hyperparameters reported in [40]. We showcase results of our algorithm in Figure 8, 9, and in the supplementary material.

5. Conclusion

In this work we present a new image denoising approach, which samples from the posterior distribution given the noisy image. We argue that in order to attain high perceptual quality, a denoising algorithm should be stochastic rather than deterministic, having multiple possible outcomes. We present a denoising algorithm along these lines, showcasing high-quality results. Our method relies on the annealed Langevin dynamics algorithm, requiring only an MMSE denoiser, without any additional retraining, constraints on the model architecture, nor extra model parameters. In addition, we extend our algorithm for handling the problem of noisy image inpainting.

Our algorithm takes a significant amount of time ($\sim$ 2 minutes for 8 CelebA images) in order to guarantee a proper convergence to a valid sampling result. Means to speed-up this procedure are therefore necessary. Our future work will focus on speeding this method by multi-scale denoising [10], deployment of denoisers of varying complexities, and other acceleration techniques. Other future research directions we consider include (i) treating general content images using general purpose denoisers, and handling much larger images (our current solution operates on images of up to size 256×256 pixels), (ii) assessing the image manifold as implicitly implied by varying denoisers; and (iii) developing uncertainty measures for the denoising solution to expose and quantify the diversity of the possible solutions.

6. Acknowledgement

We would like to thank Mauricio Belbracio and Peyman Milanfar from Google Research for drawing our attention to the work in [23, 39], which inspired our work.

References

[1] Aviad Aberdam, Dror Simon, and Michael Elad. When and how can deep generative models be inverted? *arXiv preprint arXiv:2006.15555*, 2020.

[2] Jonas Adler and Ozan Öktem. Deep bayesian inversion. *arXiv preprint arXiv:1811.05910*, 2018.

[3] Jonas Adler and Ozan Öktem. Deep posterior sampling: Uncertainty quantification for large scale inverse problems. In *International Conference on Medical Imaging with Deep Learning–Extended Abstract Track*, 2019.

[4] Michal Aharon and Michael Elad. Sparse and redundant modeling of image content using an image-signature-dictionary. *SIAM Journal on Imaging Sciences*, 1(3):228–247, 2008.

[5] Yuval Bahat and Tomer Michaeli. Explorable super resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2716–2725, 2020.

[6] Julian Besag. Markov chain monte carlo for statistical inference. *Center for Statistics and the Social Sciences*, 9:24–25, 2001.

[7] V Bhanumathi and S Lavanya. Image denoising by wavelet bayesian network based on map estimation. *International Journal of Mathematical and Computational Methods*, 2, 2017.

[8] Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. *arXiv preprint arXiv:1711.06077*, 2017.

[9] Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6228–6237, 2018.

[10] Adam Block, Youssef Mroueh, Alexander Rakhlin, and Jerret Ross. Fast mixing of multi-scale langevin dynamics underthe manifold hypothesis. *arXiv preprint arXiv:2006.11166*, 2020.

[11] Ajay Boyat and Brijendra Joshi. A review paper: Noise models in digital image processing. *Signal & Image Processing: An International Journal*, 6, 05 2015.

[12] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[13] Antoni Buades, Bartomeu Coll, and J-M Morel. A non-local algorithm for image denoising. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 60–65. IEEE, 2005.

[14] Damon M Chandler and Sheila S Hemami. Vsnr: A wavelet-based visual signal-to-noise ratio for natural images. *IEEE transactions on image processing*, 16(9):2284–2298, 2007.

[15] Priyam Chatterjee and Peyman Milanfar. Is denoising dead? *IEEE Transactions on Image Processing*, 19(4):895–911, 2009.

[16] Yunjin Chen and Thomas Pock. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE transactions on pattern analysis and machine intelligence*, 39(6):1256–1272, 2016.

[17] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8):2080–2095, 2007.

[18] Ralph D'Agostino and Egon S Pearson. Tests for departure from normality. empirical results for the distributions of b_2 and $\sqrt{b_1}$. *Biometrika*, 60(3):613–622, 1973.

[19] Michael Elad and Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image processing*, 15(12):3736–3745, 2006.

[20] Alona Golts, Daniel Freedman, and Michael Elad. Deep energy: Using energy functions for unsupervised training of dnns. *arXiv*, 2018.

[21] Shuhang Gu, Lei Zhang, Wangmeng Zuo, and Xiangchu Feng. Weighted nuclear norm minimization with application to image denoising. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2862–2869, 2014.

[22] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33, 2020.

[23] Zahra Kadkhodaie and Eero P Simoncelli. Solving linear inverse problems using the prior implicit in a denoiser. *arXiv preprint arXiv:2007.13640*, 2020.

[24] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8110–8119, 2020.

[25] Marc Lebrun, Antoni Buades, and Jean-Michel Morel. A nonlocal bayesian image denoising algorithm. *SIAM Journal on Imaging Sciences*, 6(3):1665–1688, 2013.

[26] Stamatios Lefkimmiatis. Universal denoising networks: a novel cnn architecture for image denoising. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3204–3213, 2018.

[27] Qi Lei, Ajil Jalal, Inderjit S Dhillon, and Alexandros G Dimakis. Inverting deep generative models, one layer at a time. In *Advances in Neural Information Processing Systems*, pages 13910–13919, 2019.

[28] Anat Levin, Boaz Nadler, Fredo Durand, and William T Freeman. Patch complexity, finite pixel correlations and optimal denoising. In *European Conference on Computer Vision*, pages 73–86. Springer, 2012.

[29] Ding Liu, Bihan Wen, Yuchen Fan, Chen Change Loy, and Thomas S Huang. Non-local recurrent network for image restoration. In *Advances in Neural Information Processing Systems*, pages 1673–1682, 2018.

[30] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015.

[31] Sachit Menon, Alexandru Damian, Shijia Hu, Nikhil Ravi, and Cynthia Rudin. Pulse: Self-supervised photo upsampling via latent space exploration of generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2437–2445, 2020.

[32] Tal Remez, Or Litany, Raja Giryes, and Alex M Bronstein. Deep class-aware image denoising. In *2017 international conference on sampling theory and applications (SampTA)*, pages 138–142. IEEE, 2017.

[33] Tal Remez, Or Litany, Raja Giryes, and Alex M Bronstein. Class-aware fully convolutional gaussian and poisson denoising. *IEEE Transactions on Image Processing*, 27(11):5707–5722, 2018.

[34] Gareth O Roberts, Richard L Tweedie, et al. Exponential convergence of langevin distributions and their discrete approximations. *Bernoulli*, 2(4):341–363, 1996.

[35] Yaniv Romano, Michael Elad, and Peyman Milanfar. The little engine that could: Regularization by denoising (red). *SIAM Journal on Imaging Sciences*, 10(4):1804–1844, 2017.

[36] Stefan Roth and Michael J Black. Fields of experts: A framework for learning image priors. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 2, pages 860–867. IEEE, 2005.

[37] Hamid R Sheikh and Alan C Bovik. Image information and visual quality. *IEEE Transactions on image processing*, 15(2):430–444, 2006.

[38] Hamid R Sheikh, Alan C Bovik, and Gustavo De Veciana. An information fidelity criterion for image quality assessment using natural scene statistics. *IEEE Transactions on image processing*, 14(12):2117–2128, 2005.

[39] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *Advances in Neural Information Processing Systems*, pages 11918–11930, 2019.

[40] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *arXiv preprint arXiv:2006.09011*, 2020.

[41] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[42] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. In *Proceedings of the IEEE international conference on computer vision*, pages 4539–4547, 2017.

[43] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9446–9454, 2018.

[44] Gregory Vaksman, Michael Elad, and Peyman Milanfar. Lidia: Lightweight learned image denoising with instance adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 524–525, 2020.

[45] Gregory Vaksman, Michael Zibulevsky, and Michael Elad. Patch ordering as a regularization for inverse problems in image processing. *SIAM Journal on Imaging Sciences*, 9(1):287–319, 2016.

[46] Singanallur V Venkatakrishnan, Charles A Bouman, and Brendt Wohlberg. Plug-and-play priors for model based reconstruction. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 945–948. IEEE, 2013.

[47] Zhou Wang and Alan C Bovik. Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE signal processing magazine*, 26(1):98–117, 2009.

[48] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[49] Zhou Wang, Eero P Simoncelli, and Alan C Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, pages 1398–1402. Ieee, 2003.

[50] Alexander Wong, Akshaya Mishra, Wen Zhang, Paul Fieguth, and David A Clausi. Stochastic image denoising based on markov-chain monte carlo sampling. *Signal Processing*, 91(8):2112–2120, 2011.

[51] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015.

[52] Guoshen Yu, Guillermo Sapiro, and Stéphane Mallat. Solving inverse problems with piecewise linear estimators: From gaussian mixture models to structured sparsity. *IEEE Transactions on Image Processing*, 21(5):2481–2499, 2011.

[53] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017.

[54] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing*, 27(9):4608–4622, 2018.

[55] Lin Zhang, Lei Zhang, Xuanqin Mou, and David Zhang. Fsim: A feature similarity index for image quality assessment. *IEEE transactions on Image Processing*, 20(8):2378–2386, 2011.

[56] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, 2018.

[57] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[58] Wenda Zhou and Shirin Jalali. Towards theoretically-founded learning-based denoising. In *2019 IEEE International Symposium on Information Theory (ISIT)*, pages 2714–2718. IEEE, 2019.

[59] Daniel Zoran and Yair Weiss. From learning models of natural image patches to whole image restoration. In *2011 International Conference on Computer Vision*, pages 479–486. IEEE, 2011.