Image Restoration using Autoencoding Priors

Siavash Arjomand Bigdeli[1] and Matthias Zwicker[1,2]
[1]*University of Bern, Bern, Switzerland*
[2]*University of Maryland, College Park, U.S.A.*
{*bigdeli, zwicker*}*@inf.unibe.ch*

Keywords: Image Restoration, Denoising Autoencoders, Mean Shift.

Abstract: We propose to leverage denoising autoencoder networks as priors to address image restoration problems. We build on the key observation that the output of an optimal denoising autoencoder is a local mean of the true data density, and the autoencoder error (the difference between the output and input of the trained autoencoder) is a mean shift vector. We use the magnitude of this mean shift vector, that is, the distance to the local mean, as the negative log likelihood of our natural image prior. For image restoration, we maximize the likelihood using gradient descent by backpropagating the autoencoder error. A key advantage of our approach is that we do not need to train separate networks for different image restoration tasks, such as non-blind deconvolution with different kernels, or super-resolution at different magnification factors. We demonstrate state of the art results for non-blind deconvolution and super-resolution using the same autoencoding prior.

1 INTRODUCTION

Deep learning has been successful recently at advancing the state of the art in various low-level image restoration problems including image super-resolution, deblurring, and denoising. The common approach to solve these problems is to train a network end-to-end for a specific task, that is, different networks need to be trained for each noise level in denoising, or each magnification factor in super-resolution. This makes it hard to apply these techniques to related problems such as non-blind deconvolution, where training a network for each blur kernel would be impractical.

A standard strategy to approach image restoration problems is to design suitable priors that can successfully constrain these underdetermined problems. Classical techniques include priors based on edge statistics, total variation, sparse representations, or patch-based priors. In contrast, our key idea is to leverage denoising autoencoder (DAE) networks (Vincent et al., 2008) as natural image priors. We build on the key observation by Alain et al. (Alain and Bengio, 2014) that for each input, the output of an optimal denoising autoencoder is a local mean of the true natural image density. The weight function that defines the local mean is equivalent to the noise distribution used to train the DAE. Our insight is that the autoencoder error, which is the difference between the output and input of the trained autoencoder, is a mean shift vector (Comaniciu and Meer, 2002), and the noise distribution represents a mean shift kernel.

Hence, we leverage neural DAEs in an elegant manner to define powerful image priors: Given the trained autoencoder, our natural image prior is based on the magnitude of the mean shift vector. For each image, the mean shift is proportional to the gradient of the true data distribution smoothed by the mean shift kernel, and its magnitude is the distance to the local mean in the distribution of natural images. With an optimal DAE, the energy of our prior vanishes exactly at the stationary points of the true data distribution smoothed by the mean shift kernel. This makes our prior attractive for maximum a posteriori (MAP) estimation.

For image restoration, we include a data term based on the known image degradation model. For each degraded input image, we maximize the likelihood of our solution using gradient descent by backpropagating the autoencoder error and computing the gradient of the data term. Intuitively, this means that our approach iteratively moves our solution closer to its local mean in the natural image density, while satisfying the data term. This is illustrated in Figure 1.

A key advantage of our approach is that we do not need to train separate networks for different image restoration tasks, such as non-blind deconvolution with different kernels, or super-resolution at different magnification factors. Even though our autoencoding

Figure 1: We propose a natural image prior based on a denoising autoencoder, and apply it to image restoration problems like non-blind deblurring. The output of an optimal denoising autoencoder is a local mean of the true natural image density, and the autoencoder error is a mean shift vector. We use the magnitude of the mean shift vector as the negative log likelihood of our prior. To restore an image from a known degradation, we use gradient descent to iteratively minimize the mean shift magnitude while respecting a data term. Hence, step-by-step we shift our solution closer to its local mean in the natural image distribution.

prior is trained on a denoising problem, it is highly effective at removing these different degradations. We demonstrate state of the art results for non-blind deconvolution and super-resolution using the same autoencoding prior.

In subsequent research, Bigdeli et al. (Bigdeli et al., 2017) built on this work by incorporating our proposed prior in a Bayes risk minimization framework, which allows them to perform noise-blind restoration.

2 RELATED WORK

Image restoration, including deblurring, denoising, and super-resolution, is an underdetermined problem that needs to be constrained by effective priors to obtain acceptable solutions. Without attempting to give a complete list of all relevant contributions, the most common successful techniques include priors based on edge statistics (Fattal, 2007; Tappen et al., 2003), total variation (Perrone and Favaro, 2014), sparse representations (Aharon et al., 2006; Yang et al., 2010), and patch-based priors (Zoran and Weiss, 2011; Levin et al., 2012; Schmidt et al., 2016a). While some of these techniques are tailored for specific restoration problems, recent patch-based priors lead to state of the art results for multiple applications, such as deblurring and denoising (Schmidt et al., 2016a).

Solving image restoration problems using neural networks seems attractive because they allow for straightforward end-to-end learning. This has led to remarkable success for example for single image super-resolution (Dong et al., 2014; Gu et al., 2015; Dong et al., 2016; Liu et al., 2016; Kim et al., 2016) and denoising (Burger et al., 2012; Mao et al., 2016). A disadvantage of the end-to-end learning is that, in principle, it requires training a different network for each restoration task (e.g., each different noise level or magnification factor). While a single network can be effective for denoising different noise levels (Mao et al., 2016), and similarly a single network can perform well for different super-resolution factors (Kim et al., 2016), it seems unlikely that in non-blind deblurring, the same network would work well for arbitrary blur kernels. Additionally, experiments by Zhang et al. (Zhang et al., 2016) show that training a network for multiple tasks reduces performance compared to training each task on a separate network. Previous research addressing non-blind deconvolution using deep networks includes the work by Schuler et al. (Schuler et al., 2013) and more recently Xu et al. (Xu et al., 2014), but they require end-to-end training for each blur kernel.

A key idea of our work is to train a neural autoencoder that we use as a prior for image restoration. Autoencoders are typically used for unsupervised representation learning (Vincent et al., 2010). The focus of these techniques lies on the descriptive strength of the learned representation, which can be used to address classification problems for example. In addition, generative models such as generative adversarial networks (Goodfellow et al., 2014) or variational autoencoders (Kingma and Welling, 2014) also facilitate sampling the representation to generate new data. Their network architectures usually consist of an encoder followed by a decoder, with a bottleneck that is interpreted as the data representation in the middle. The ability of autoencoders and generative models to create images from abstract representations makes them attractive for restoration problems. Notably, the encoder-decoder architecture in Mao et al.'s image restoration work (Mao et al., 2016) is highly reminiscent of autoencoder architectures, although they train their network in a supervised manner.

A denoising autoencoder (Vincent et al., 2008) is an autoencoder trained to reconstruct data that was corrupted with noise. Previously, Alain and Bengio (Alain and Bengio, 2014) and Nguyen et al. (Nguyen et al., 2016) used DAEs to construct generative models. We are inspired by the observation of Alain and Bengio that the output of an optimal DAE is a local mean of the true data density. Hence, our insight is that the autoencoder error (the difference between its output and input) is a mean shift vector (Comaniciu and Meer, 2002). This motivates

using the magnitude of the autoencoder error as our prior.

Our work has an interesting connection to the plug-and-play priors introduced by Venkatakrishnan et al. (Venkatakrishnan et al., 2013). They solve regularized inverse (image restoration) problems using ADMM (alternating directions method of multipliers), and they make the key observation that the optimization step involving the prior is a denoising problem, that can be solved with any standard denoiser. Using this framework, CNN-based denoisers have been employed (Zhang et al., 2017) for image restoration. While their use of a denoiser is a consequence of ADMM, our work shines a light on how a trained denoiser is directly related to the underlying data density (the distribution of natural images). Our approach also leads to a different, simpler gradient descent optimization that does not rely on ADMM.

In summary, the main contribution of our work is that we make the connection between DAEs and mean shift, which allows us to show the relation of an optimal DAE to the underlying data distribution, and to leverage DAEs to define a prior for image restoration problems. We train a DAE and demonstrate that the resulting prior is effective for different restoration problems, including deblurring with arbitrary kernels and super-resolution with different magnification factors.

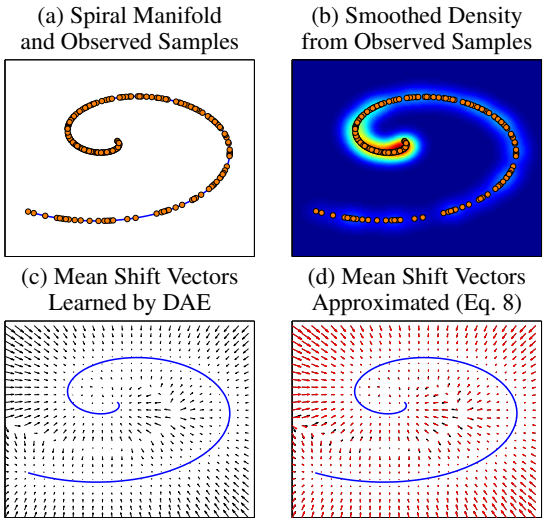

Figure 2: Visualization of a denoising autoencoder using a 2D spiral density. Given input samples of a true density (a), the autoencoder is trained to pull each sample corrupted by noise back to its original location. Adding noise to the input samples smooths the density represented by the samples (b). Assuming an infinite number of input samples and an autoencoder with unlimited capacity, for each input, the output of the optimal trained autoencoder is the local mean of the true density. The local weighting function corresponds to the noise distribution that was used during training, and it represents a mean shift kernel (Comaniciu and Meer, 2002). The difference between the output and the input of the autoencoder is a mean shift vector (c), which vanishes at local extrema of the true density smoothed by the mean shift kernel. Due to practical limitations (Section 4.2), we approximate the mean shift vectors (d, red) using Equation 8. The difference between the true mean shift vectors (d, black) and our approximate vectors (d, red) vanishes as we get closer to the manifold.

3 PROBLEM FORMULATION

We formulate image restoration in a standard fashion as a maximum a posteriori (MAP) problem (Joshi et al., 2009). We model degradation including blur, noise, and downsampling as

$$B = D(I \otimes K) + \xi, \tag{1}$$

where B is the degraded image, D is a down-sampling operator using point sampling, I is the unknown image to be recovered, K is a known, shift-invariant blur kernel, and $\xi \sim \mathcal{N}(0, \sigma_d^2)$ is the per-pixel i.i.d. degradation noise. The posterior probability of the unknown image is $p(I|B) = p(B|I)p(I)/p(B)$, and we maximize it by minimizing the corresponding negative log likelihoods L,

$$\underset{I}{\operatorname{argmax}}\, p(I|B) = \underset{I}{\operatorname{argmin}}\, [L(B|I) + L(I)]\,. \tag{2}$$

Under the Gaussian noise model, the negative data log likelihood is

$$L(B|I) = \|B - D(I \otimes K)\|^2 / \sigma_d^2. \tag{3}$$

Note that this implies that the blur kernel K is given at the higher resolution, before down-sampling by point sampling with D. Our contribution now lies in a novel image prior $L(I)$, which we introduce next.

4 DENOISING AUTOENCODER AS NATURAL IMAGE PRIOR

We will leverage a neural autoencoder to define a natural image prior. In particular, we are building on denoising autoencoders (DAE) (Vincent et al., 2008) that are trained using Gaussian noise and an expected quadratic loss. Inspired by the results by Alain et al. (Alain and Bengio, 2014), we relate the optimal DAE to the underlying data density and exploit this relation to define our prior.

4.1 Denoising Autoencoders

We visualize the intuition behind DAEs in Figure 2. Let us denote a DAE as A_{σ_η}. Given an input image I, its output is an image $A_{\sigma_\eta}(I)$. A DAE A_{σ_η} is trained to minimize (Vincent et al., 2008)

$$\mathcal{L}_{\text{DAE}} = \mathbb{E}_{\eta,I}\left[\|I - A_{\sigma_\eta}(I+\eta)\|^2\right], \tag{4}$$

where the expectation is over all images I and Gaussian noise η with variance σ_η^2, and A_{σ_η} indicates that the DAE was trained with noise variance σ_η^2. It is important to note that the noise variance σ_η^2 here is not related to the degradation noise and its variance σ_d^2, and it is not a parameter to be learned. Instead, it is a user specified parameter whose role becomes clear with the following proposition. Let us denote the true data density of natural images as $p(I)$. Alain et al. (Alain and Bengio, 2014) show that the output $A_{\sigma_\eta}(I)$ of the optimal DAE (assuming unlimited capacity) is related to the true data density $p(I)$ as

$$\begin{aligned} A_{\sigma_\eta}(I) &= \frac{\mathbb{E}_\eta\left[p(I-\eta)(I-\eta)\right]}{\mathbb{E}_\eta\left[p(I-\eta)\right]} \\ &= \frac{\int g_{\sigma_\eta^2}(\eta)p(I-\eta)(I-\eta)d\eta}{\int g_{\sigma_\eta^2}(\eta)p(I-\eta)d\eta}. \end{aligned} \tag{5}$$

This reveals an interesting connection to the mean shift algorithm (Comaniciu and Meer, 2002):

Proposition 1. *The autoencoder error, that is the difference between the output and the input of the autoencoder $A_{\sigma_\eta}(I) - I$ is an exact mean shift vector. More precisely, the mean shift vector ((Comaniciu and Meer, 2002), Eq. 17) is a Monte Carlo estimate of Equation* (5) *using random samples $\xi_i \sim p, i = 1 \ldots n$.*

Proof. By substituting $\xi = I - \eta$ in Equation (5), and Monte Carlo estimation of the integrals with a sum over n random samples $\xi_i \sim p, i = 1 \ldots n$, we directly arrive at the original mean shift formulation ((Comaniciu and Meer, 2002), Eq. 17). □

The autoencoder output can be interpreted as a local mean or a weighted average of images in the neighborhood of I. The weights are given by the true density $p(I)$ multiplied by the noise distribution that was used during training, which is a local Gaussian kernel $g_{\sigma_\eta^2}(\eta)$ centered at I with variance σ_η^2. Hence the parameter σ_η^2 of the autoencoder determines the size of the region around I that contributes to the local mean.

The key of our approach is the following theorem, which we prove in the appendix:

Theorem 1. *When the training noise η has a Gaussian distribution, the autoencoder error is proportional to the gradient of the log likelihood of the data density p smoothed by the Gaussian kernel $g_{\sigma_\eta^2}(\eta)$,*

$$A_{\sigma_\eta}(I) - I = \sigma_\eta^2 \nabla \log\left[g_{\sigma_\eta^2} * p\right](I), \tag{6}$$

where $*$ means convolution.

Figure 3: Local minimum of our natural image prior. Starting with a noisy image (left), we minimize the prior via gradient descent (middle: intermediate step) to reach the local minimum (right).

Hence we observe that the autoencoder error vanishes at stationary points, including local extrema, of the true density smoothed by the Gaussian kernel.

4.2 Autoencoding Prior

The above observations inspire us to use the squared magnitude of the mean shift vector as the energy (the negative log likelihood) of our prior, $L(I) = \|A_{\sigma_\eta}(I) - I\|^2$. This energy is very powerful because it tells us how close an image I is to its local mean $A_{\sigma_\eta}(I)$ in the true data density, and it vanishes at local extrema of the true density smoothed by the mean shift kernel. Figure 2(c), illustrates how small values of $L(I) = \|A_{\sigma_\eta}(I) - I\|^2$ occur close to the data manifold, as desired. Figure 3 visualizes a local minimum of our prior on natural images, which we find by iteratively minimizing the prior via gradient descent starting from a noisy input, without any help from a data term.

Including the data term, we recover latent images as

$$\begin{aligned} \operatorname*{argmin}_I \|B - D(I \otimes K)\|^2/\sigma_d^2 \\ + \gamma\|A_{\sigma_\eta}(I) - I\|^2. \end{aligned} \tag{7}$$

Our energy has two parameters that we will adjust based on the restoration problem. First, this is the mean shift kernel size σ_η, and second we introduce a parameter γ to weight the relative influence of the data term and the prior.

Optimization. Given a trained autoencoder, we minimize our loss function in Equation 7 by applying gradient descent and computing the gradient of the prior using backpropagation through the autoencoder. Algorithm 1 shows the steps to minimize Equation 7. In the first step of each iteration, we compute the gradient of the data term with respect to image I. The

Algorithm 1: Proposed gradient descent. We express convolution as a matrix-vector product.

loop $\#iterations$

- Compute data term gradients $\nabla_I L(I|B)$:
$$K^T D^T (DKI - B)/\sigma_d^2$$
- Compute prior gradients $\nabla_I L(I)$:
$$\nabla_I A_{\sigma_\eta}(I)^T \Big(A_{\sigma_\eta}(I) - I\Big) + I - A_{\sigma_\eta}(I)$$
- Update I by descending
$$\nabla_I L(I|B) + \gamma \nabla_I L(I)$$

end loop

second step is to find the gradients for our prior. The gradient of the mean shift vector $\|A_{\sigma_\eta}(I) - I\|^2$ requires the gradient of the autoencoder $A_{\sigma_\eta}(I)$, which we compute by backpropagation through the network. Finally, the image I is updated using the weighted sum of the two gradient terms.

Overcoming Training Limitations. The theory above assumes unlimited data and time to train an unlimited capacity autoencoder. In particular, to learn the true mean shift mapping, for each natural image the training data needs to include noise patterns that lead to other natural images. In practice, however, such patterns virtually never occur because of the high dimensionality. Since the DAE never observed natural images during training (produced by adding noise to other images), it overfits to noisy images. This is problematic during the gradient descent optimization, when the input to the DAE does not have noise.

As a workaround, we obtained better results by adding noise to the image before feeding it to the trained DAE during optimization. We further justify this by showing that with this workaround, we can still approximate a DAE that was trained with a desired noise variance σ_η^2. That is,

$$A_{\sigma_\eta}(I) - I \approx 2\Big(\mathbb{E}_\varepsilon\left[A_{\sigma_\varepsilon}(I - \varepsilon)\right] - I\Big), \qquad (8)$$

where $\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2)$, and A_{σ_ε} is a DAE trained with $\sigma_\varepsilon^2 = \sigma_\eta^2/2$. The key point here is that the consecutive convolution with two Gaussians is equivalent to a single Gaussian convolution with the sum of the variances (refer to supplementary material for the derivation). This is visualized in Figure 2(d). The red vectors indicate the approximated mean shift vectors using Equation 8 and the black vectors indicate the exact mean shift vectors. The approximation error decreases as we approach the true manifold.

During optimization, we approximate the expected value in Equation 8 by stochastically sampling over ε. We use momentum of 0.9 and step size 0.1 in all experiments and we found that using one noise sample per iteration performs well enough to compute meaningful gradients. This approach resulted in a PSNR gain of around $1.7dB$ for the super-resolution task (Section 5.1), compared to evaluating the left hand side of Equation 8 directly.

Bad Local Minima and Convergence. The mean shift vector field learned by the DAE could vanish in low density regions (Alain and Bengio, 2014), which corresponds to undesired local minima for our prior. In practice, however, we have not observed such degenerate solutions because our data term pulls the solution towards natural images. In all our experiments the optimization converges smoothly (Figure 1, intermediate steps), although we cannot give a theoretical guarantee.

4.3 Autoencoder Architecture and Training

Our network architecture is inspired by Zhang et al. (Zhang et al., 2016). The network consists of 20 convolutional layers with batch normalization in between except for the first and last layers, and we use ReLU activations except for the last convolutional layer. The convolution kernels are of size 3×3 and the number of channels are 3 (RGB) for input and output and 64 for the rest of the layers. Unlike typical neural autoencoders, our network does not have a bottleneck. An explicit latent space implemented as a bottleneck is not required in principle for DAE training, and we do not need it for our application. We use a fully-convolutional network that allows us to compute the gradients with respect to the image more efficiently since the neuron activations are shared between many pixels. Our network is trained on color images of the ImageNet dataset (Deng et al., 2009) by adding Gaussian noise with standard deviation $\sigma_\varepsilon = 25$ (around 10%). We perform residual learning by minimizing the L_2 distance of the output layer to the ground truth noise. We used the *Caffe* package (Jia et al., 2014) and employed an Adam solver (Kingma and Ba, 2014) with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and learning rate of 0.001, which we reduced during the iterations.

5 EXPERIMENTS AND RESULTS

We compare our approach, Denoising Autoencoder Prior (DAEP), to state of the art methods in super-resolution and non-blind deconvolution pro-

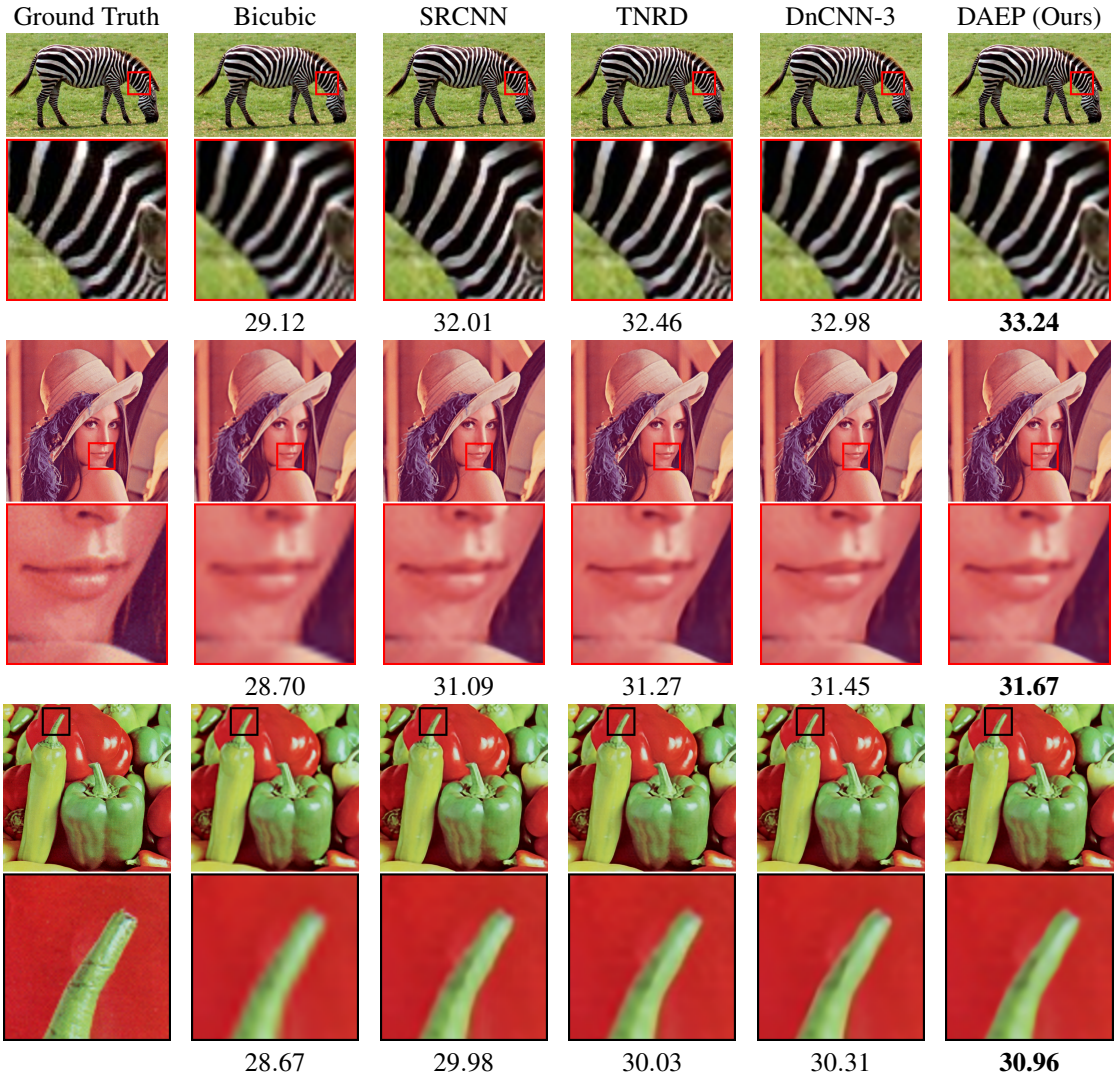

Figure 4: Comparison of super-resolution for scale factor 2 (top row), scale factor 3 (middle row), and scale factor 4 (bottom row) with the corresponding PSNR (*dB*) scores.

blems. For all our experiments, we trained the autoencoder with $\sigma_\varepsilon = 25$ ($\sigma_\eta = 25\sqrt{2}$), and the parameter of our energy (Equation 7) were set to $\gamma = 6.875/\sigma_\eta^2$. We always perform 300 gradient descent iteration steps during image restoration . The source code of the proposed method is available at https://github.com/siavashbigdeli/DAEP.

5.1 Super-Resolution

The super-resolution problem is usually defined in absence of noise ($\sigma_d = 0$), therefore we weight the prior by the inverse square root of the iteration number. This policy starts with a rough regularization and reduces the prior weight in each iteration, leading to solutions that satisfy $\sigma_d = 0$. We compare our method to recent techniques by Kim et al. (Kim et al., 2016) (SRCNN), Dong et al. (Dong et al., 2016) (VDSR), Zhang et al. (Zhang et al., 2016) (DnCNN-3), Chen and Pock (Chen and Pock, 2016) (TNRD), and IRCNN by Zhang et al. (Zhang et al., 2017). SRCNN, VDSR and DnCNN-3 train an end-to-end network by minimizing the L_2 loss between the output of the network and the high-resolution ground truth, and TNRD uses a learned reaction diffusion model. While SRCNN and TNRD were trained separately for each scale, the VDSR and DnCNN-3 models were trained jointly on $\times 2, 3$ and 4 (DnCNN-3 training included also denoising and JPEG artifact removal tasks). For $\times 5$ super-resolution we used SRCNN and TNRD models that were trained on $\times 4$, and we used VDSR and DnCNN-3 models trained jointly on $\times 2, 3$ and

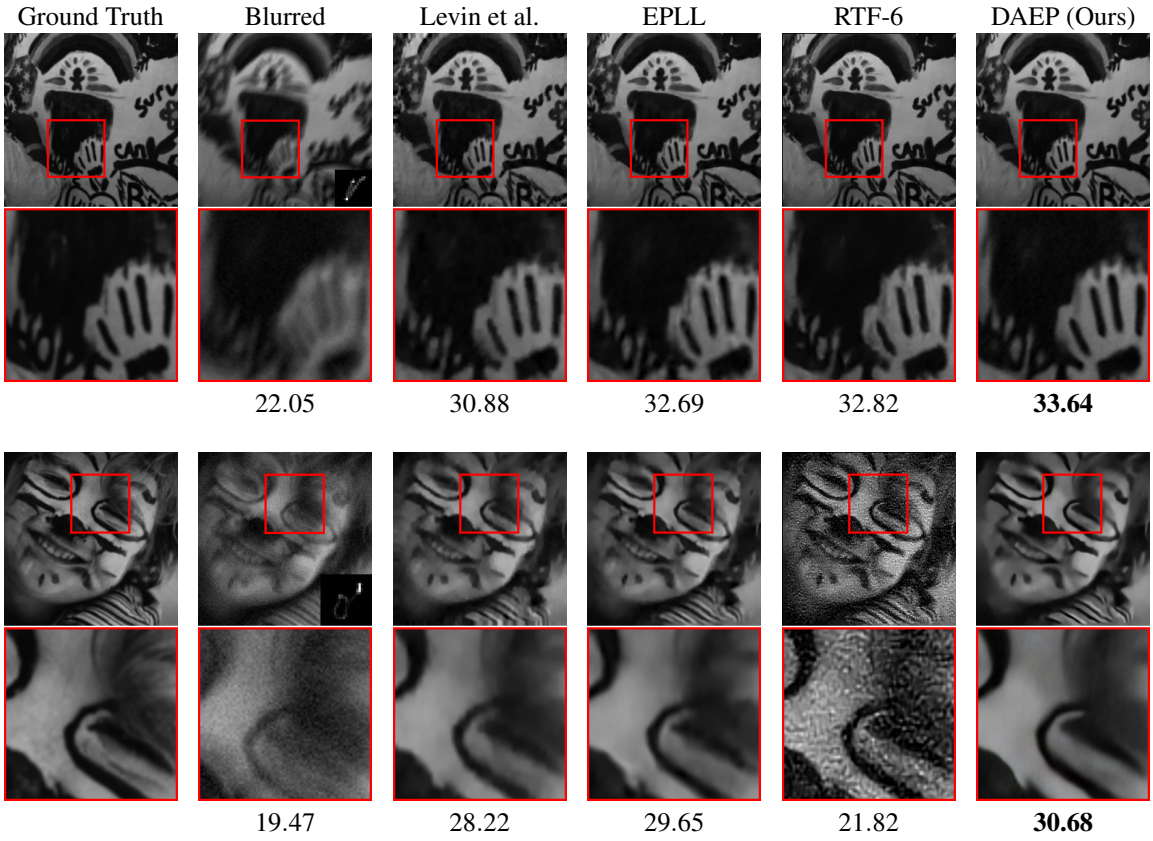

Figure 5: Comparison of non-blind deconvolution with $\sigma = 2.55$ additive noise (top row) and $\sigma = 7.65$ additive noise (bottom row) with the corresponding PSNR (*dB*) scores. The kernel is visualized in the bottom right of the blurred image.

4. Tables 1, 2 compare the average PSNR of the super-resolved images from 'Set5' and 'Set14' datasets (Bevilacqua et al., 2012; Zeyde et al., 2012) for scale factors $\times 2, 3, 4$, and 5. We compute PSNR values over cropped RGB images (where the crop size in pixels corresponds to the scale factor) for all methods. For SRCNN, however, we used a boundary of 13 pixels to provide full support for their network. While SRCNN, VDSR and DnCNN-3 solve directly for MMSE, our method solves for the MAP solution, which is not guaranteed to have better PSNR. Still, we achieve better results in average. For scale factor $\times 5$ our method performs significantly better since our prior does not need to be trained for a specific scale. Figure 4 shows visual comparisons to the super-resolution results from SRCNN (Dong et al., 2016), TNRD (Chen and Pock, 2016), and DnCNN-3 (Zhang et al., 2016) on three example images. We exclude results of VDSR due to limited space and visual similarity with DnCNN-3. Our natural image prior provides clean and sharp edges over all magnification factors.

Table 1: Average PSNR (*dB*) for super-resolution on 'Set5' (Bevilacqua et al., 2012).

Method	$\times 2$	$\times 3$	$\times 4$	$\times 5$
Bicubic	31.80	28.67	26.73	25.32
SRCNN	34.50	30.84	28.60	26.12
TNRD	34.62	31.08	28.83	26.88
VDSR	34.50	31.39	29.19	25.91
DnCNN-3	35.20	**31.58**	**29.30**	26.30
IRCNN	35.07	31.26	29.01	27.13
DAEP (Ours)	**35.23**	31.44	29.01	**27.19**

Table 2: Average PSNR (*dB*) for super-resolution on 'Set14' (Zeyde et al., 2012).

Method	$\times 2$	$\times 3$	$\times 4$	$\times 5$
Bicubic	28.53	25.92	24.44	23.46
SRCNN	30.52	27.48	25.76	24.05
TNRD	30.53	27.60	25.92	24.61
VDSR	30.72	27.81	26.16	24.01
DnCNN-3	30.99	27.93	**26.25**	24.26
IRCNN	30.79	27.68	25.96	24.73
DAEP (Ours)	**31.07**	**27.93**	26.13	**24.88**

Figure 6: Comparison of non-blind deconvolution methods on the 21st image from the Kodak image set (Kodak, 2013). For each method, we report the PSNR (*dB*) of the visualized image (left) and the average PSNR on the whole set (right). The results of other methods were reproduced from Fortunato and Oliveira (Fortunato and Oliveira, 2014) for ease of comparison.

Table 3: Average PSNR (*dB*) for non-blind deconvolution on Levin et al.'s (Levin et al., 2007) dataset for different noise levels.

Method	2.55	7.65	12.75	time(s)
Levin	31.09	27.40	25.36	3.09
EPLL	32.51	28.42	26.13	16.49
RTF-6	32.51	21.44	16.03	9.82
IRCNN	30.78	28.77	**27.41**	2.47
DAEP (Ours)	**32.69**	**28.95**	26.87	11.19

5.2 Non-Blind Deconvolution

To evaluate and compare our method for non-blind deconvolution we used the dataset from Levin et al. (Levin et al., 2007) with four grayscale images and eight blur kernels in different sizes from 13×13 to 27×27. We compare our results to Levin et al. (Levin et al., 2007) (Levin), Zoran and Weiss (Zoran and Weiss, 2011) (EPLL), Schmidt et al. (Schmidt et al., 2016b) (RTF-6), and IRCNN by Zhang et al. (Zhang et al., 2017) in Table 3, where we show the average PSNR of the deconvolution for three levels of additive noise ($\sigma \in \{2.55, 7.65, 12.75\}$). Note that RTF-6 (Schmidt et al., 2016b) is only trained for noise level $\sigma = 2.55$, therefore it does not perform well for other noise levels. Figure 5 provides visual comparisons for two deconvolution result images. Our natural image prior achieves higher PSNR and produces sharper edges and less visual artifacts compared to Levin et al. (Levin et al., 2007), Zoran and Weiss (Zoran and Weiss, 2011), and Schmidt et al. (Schmidt et al., 2016b). We report runtimes for different methods in Table 3 for image size of 128x128 on an Nvidia Titan X GPU. Our runtime is on par with popular methods such as EPLL (Zoran and Weiss, 2011).

We performed an additional comparison on color images similar to Fortunato and Oliveira (Fortunato

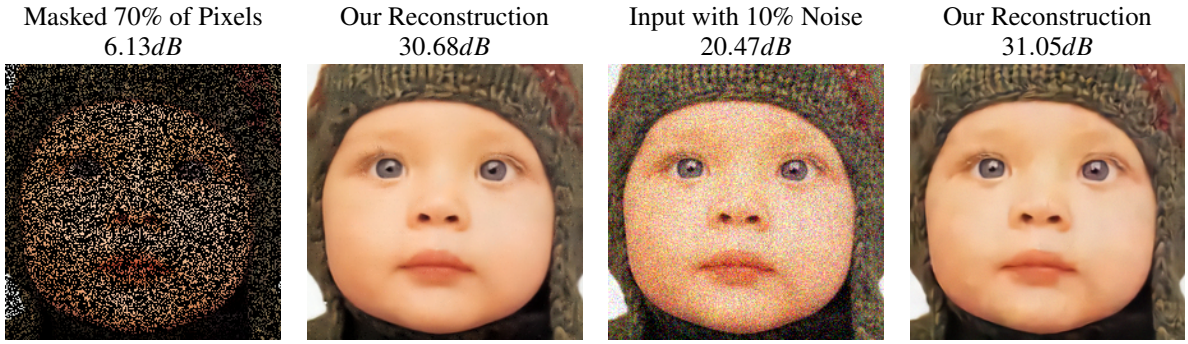

Figure 7: Restoration of images corrupted by noise and holes using the same autoencoding prior as in our other experiments.

and Oliveira, 2014) using 24 color images from the Kodak Lossless True Color Image Suite from PhotoCD PCD0992 (Kodak, 2013). The images are blurred with a 19×19 blur kernel from Krishnan and Fergus (Krishnan and Fergus, 2009) and 1% noise is added. Figure 6 shows visual comparisons and average PSNRs over the whole dataset. Our method produces much sharper results and achieves a higher PSNR in average over this dataset.

5.3 Discussion

A disadvantage of our approach is that it requires the solution of an optimization problem to restore each image. In contrast, end-to-end trained networks perform image restoration in a single feed-forward pass. For the increase in runtime computation, however, we gain much flexibility. With a single autoencoding prior, we obtain not only state of the art results for non-blind deblurring with arbitrary blur kernels and super-resolution with different magnification factors, but also successfully restore images corrupted by noise or holes as shown in Figure 7.

Our approach requires some user defined parameters (mean shift kernel size σ_η for DAE training and restoration, weight of the prior γ). While we use the same parameters for all experiments reported here, other applications may require to adjust these parameters. For example, we have experimented with image denoising (Figure 7), but so far we have not achieved state of the art results. We believe that this may require an adaptive kernel width for the DAE, and further fine-tuning of our parameters.

6 CONCLUSIONS

We introduced a natural image prior based on denoising autoencoders (DAEs). Our key observation is that optimally trained DAEs provide mean shift vectors on the true data density. Our prior minimizes the distances of restored images to their local means (the length of their mean shift vectors). This is powerful since mean shift vectors vanish at local extrema of the true density smoothed by the mean shift kernel. Our results demonstrate that a single DAE prior achieves state of the art results for non-blind image deblurring with arbitrary blur kernels and image super-resolution at different magnification factors. In the future, we plan to apply our autoencoding priors to further image restoration problems including denoising, colorization, or non-uniform and blind deblurring. While we used Gaussian noise to train our autoencoder, it is possible to use other types of data degradation for DAE training. Hence, we will investigate other DAE degradations to learn different data representations or use a mixture of DAEs for the prior.

REFERENCES

Aharon, M., Elad, M., and Bruckstein, A. (2006). K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Transactions on Signal Processing*, 54(11):4311–4322.

Alain, G. and Bengio, Y. (2014). What regularized auto-encoders learn from the data-generating distribution. *Journal of Machine Learning Research*, 15:3743–3773.

Bevilacqua, M., Roumy, A., Guillemot, C., and Alberi-Morel, M. (2012). Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In *British Machine Vision Conference, BMVC 2012, Surrey, UK, September 3-7, 2012*, pages 1–10.

Bigdeli, S. A., Jin, M., Favaro, P., and Zwicker, M. (2017). Deep mean-shift priors for image restoration. In *NIPS (to appear)*.

Burger, H. C., Schuler, C. J., and Harmeling, S. (2012). Image denoising: Can plain neural networks compete with BM3D? In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2392–2399.

Chen, Y. and Pock, T. (2016). Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Comaniciu, D. and Meer, P. (2002). Mean shift: a robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5):603–619.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE.

Dong, C., Loy, C. C., He, K., and Tang, X. (2014). *Learning a Deep Convolutional Network for Image Super-Resolution*, pages 184–199. Springer International Publishing, Cham.

Dong, C., Loy, C. C., He, K., and Tang, X. (2016). Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2):295–307.

Fattal, R. (2007). Image upsampling via imposed edge statistics. *ACM Trans. Graph.*, 26(3).

Fortunato, H. E. and Oliveira, M. M. (2014). Fast high-quality non-blind deconvolution using sparse adaptive priors. *The Visual Computer*, 30(6-8):661–671.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc.

Gu, S., Zuo, W., Xie, Q., Meng, D., Feng, X., and Zhang, L. (2015). Convolutional sparse coding for image super-resolution. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1823–1831.

Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM.

Joshi, N., Zitnick, C. L., Szeliski, R., and Kriegman, D. J. (2009). Image deblurring and denoising using color priors. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1550–1557.

Kim, J., Kwon Lee, J., and Mu Lee, K. (2016). Accurate image super-resolution using very deep convolutional networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Kingma, D. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kingma, D. P. and Welling, M. (2014). Auto-encoding variational bayes. In *ICLR 2014*.

Kodak (2013). Kodak lossless true color image suite. http://r0k.us/graphics/kodak/. Accessed: 2013-01-27.

Krishnan, D. and Fergus, R. (2009). Fast image deconvolution using hyper-laplacian priors. In *Advances in Neural Information Processing Systems*, pages 1033–1041.

Levin, A., Fergus, R., Durand, F., and Freeman, W. T. (2007). Image and depth from a conventional camera with a coded aperture. *ACM transactions on graphics (TOG)*, 26(3):70.

Levin, A., Nadler, B., Durand, F., and Freeman, W. T. (2012). *Patch Complexity, Finite Pixel Correlations and Optimal Denoising*, pages 73–86. Springer Berlin Heidelberg, Berlin, Heidelberg.

Liu, D., Wang, Z., Wen, B., Yang, J., Han, W., and Huang, T. S. (2016). Robust single image super-resolution via deep networks with sparse prior. *IEEE Transactions on Image Processing*, 25(7):3194–3207.

Mao, X.-J., Shen, C., and Yang, Y.-B. (2016). Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In *Proc. Neural Information Processing Systems*.

Nguyen, A., Yosinski, J., Bengio, Y., Dosovitskiy, A., and Clune, J. (2016). Plug & play generative networks: Conditional iterative generation of images in latent space. *arXiv preprint arXiv:1612.00005*.

Perrone, D. and Favaro, P. (2014). Total variation blind deconvolution: The devil is in the details. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2909–2916.

Schmidt, U., Jancsary, J., Nowozin, S., Roth, S., and Rother, C. (2016a). Cascades of regression tree fields for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(4):677–689.

Schmidt, U., Jancsary, J., Nowozin, S., Roth, S., and Rother, C. (2016b). Cascades of regression tree fields for image restoration. *IEEE transactions on pattern analysis and machine intelligence*, 38(4):677–689.

Schuler, C. J., Burger, H. C., Harmeling, S., and Schlkopf, B. (2013). A machine learning approach for non-blind image deconvolution. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 1067–1074.

Shan, Q., Jia, J., and Agarwala, A. (2008). High-quality motion deblurring from a single image. In *ACM Transactions on Graphics (TOG)*, volume 27, page 73. ACM.

Tappen, M. F., Russell, B. C., and Freeman, W. T. (2003). Exploiting the sparse derivative prior for super-resolution and image demosaicing. In *In IEEE Workshop on Statistical and Computational Theories of Vision*.

Venkatakrishnan, S. V., Bouman, C. A., and Wohlberg, B. (2013). Plug-and-play priors for model based reconstruction. In *GlobalSIP*, pages 945–948. IEEE.

Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. (2008). Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 1096–1103, New York, NY, USA. ACM.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., and Manzagol, P.-A. (2010). Stacked denoising autoencoders: Learning useful representations in a deep net-

work with a local denoising criterion. *J. Mach. Learn. Res.*, 11:3371–3408.

Wang, Y., Yang, J., Yin, W., and Zhang, Y. (2008). A new alternating minimization algorithm for total variation image reconstruction. *SIAM Journal on Imaging Sciences*, 1(3):248–272.

Xu, L., Ren, J. S., Liu, C., and Jia, J. (2014). Deep convolutional neural network for image deconvolution. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D., and Weinberger, K. Q., editors, *Advances in Neural Information Processing Systems 27*, pages 1790–1798. Curran Associates, Inc.

Yang, J., Wright, J., Huang, T. S., and Ma, Y. (2010). Image super-resolution via sparse representation. *IEEE Transactions on Image Processing*, 19(11):2861–2873.

Zeyde, R., Elad, M., and Protter, M. (2012). On single image scale-up using sparse-representations. In *Proceedings of the 7th International Conference on Curves and Surfaces*, pages 711–730, Berlin, Heidelberg. Springer-Verlag.

Zhang, K., Zuo, W., Chen, Y., Meng, D., and Zhang, L. (2016). Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *arXiv preprint arXiv:1608.03981*.

Zhang, K., Zuo, W., Gu, S., and Zhang, L. (2017). Learning deep cnn denoiser prior for image restoration. *arXiv preprint arXiv:1704.03264*.

Zhou, C. and Nayar, S. (2009). What are good apertures for defocus deblurring? In *Computational Photography (ICCP), 2009 IEEE International Conference on*, pages 1–8. IEEE.

Zoran, D. and Weiss, Y. (2011). From learning models of natural image patches to whole image restoration. In *2011 International Conference on Computer Vision*, pages 479–486.

APPENDIX

OPTIMAL DAE WITH GAUSSIAN NOISE, THEOREM 1

Here we provide the derivation for Equation 6 in Theorem 1.

Proof. We first rewrite the original equation for the DAE (Alain and Bengio (Alain and Bengio, 2014) and our Equation 5) as

$$\begin{aligned} A_{\sigma_\eta}(I) &= \frac{\mathbb{E}_\eta\left[p(I-\eta)(I-\eta)\right]}{\mathbb{E}_\eta\left[p(I-\eta)\right]} \\ &= I - \frac{\mathbb{E}_\eta\left[p(I-\eta)\eta\right]}{\mathbb{E}_\eta\left[p(I-\eta)\right]}, \eta \sim \mathcal{N}(0,\sigma_\eta^2). \end{aligned}$$

By expanding the numerator in the quotient we get

$$\begin{aligned} \mathbb{E}_\eta\left[p(I-\eta)\eta\right] &= \int g_{\sigma_\eta^2}(\eta) p(I-\eta)\eta d\eta \\ &= -\sigma_\eta^2 \int \nabla g_{\sigma_\eta^2}(\eta) p(I-\eta) d\eta, \end{aligned}$$

where we used the definition of the derivative of the Gaussian to remove η inside the integral. Now we can use the Leibniz rule to interchange the ∇ operator with the integral and we get

$$\mathbb{E}_\eta\left[p(I-\eta)\eta\right] = -\sigma_\eta^2 \nabla \mathbb{E}_\eta\left[p(I-\eta)\right].$$

Plugging this back into our equation for the DAE we get

$$A_{\sigma_\eta}(I) = I + \sigma_\eta^2 \frac{\nabla \mathbb{E}_\eta\left[p(I-\eta)\right]}{\mathbb{E}_\eta\left[p(I-\eta)\right]},$$

and using the derivative of the logarithm we see that this is

$$\begin{aligned} A_{\sigma_\eta}(I) &= I + \sigma_\eta^2 \nabla \log \mathbb{E}_\eta\left[p(I-\eta)\right] \\ &= I + \sigma_\eta^2 \nabla \log [g_{\sigma_\eta^2} * p](I) \end{aligned}$$

as in Equation 6. □

With this alternative formulation of the DAEs we have removed the normalization term in the denominator of the DAE definition. This result shows that the autoencoder error (that is, the mean shift vector) corresponds to the gradient of the log-likelihood of the distribution blurred with a Gaussian kernel with variance σ_η^2.

APPROXIMATION OF THE DAE

Here we would like to show that it is possible to approximate a DAE with another trained DAE by adding extra noise to its input, and computing the expectation of the output of this DAE over the added noise. Specifically, we can approximate DAE A_{σ_η} with bandwidth σ_η, by another DAE A_{σ_τ} with bandwidth and $\sigma_\tau \leq \sigma_\eta$ by computing

$$A_{\sigma_\eta}(I) - I \approx \frac{\sigma_\eta^2}{\sigma_\tau^2}\Big(\mathbb{E}_\varepsilon\left[A_{\sigma_\tau}(I-\varepsilon)\right] - I\Big),$$

where $\varepsilon \sim \mathcal{N}(0, \sigma_\eta^2 - \sigma_\tau^2)$. In our approach, we evaluate (the gradient of the squared magnitude of) this equation at run-time during image restoration by sampling the expected value on the right hand side using a single sample in each step of the gradient descent optimization.

To derive the above approximation, we start by using the alternative equation of the DAE from Equation 6 for A_{σ_τ} to write

$$A_{\sigma_\tau}(I) - I = \sigma_\tau^2 \nabla \log \mathbb{E}_\tau \left[p(I - \tau) \right],$$

and we take expectations of both sides over noise variable ε, that is

$$\mathbb{E}_\varepsilon \left[A_{\sigma_\tau}(I - \varepsilon) \right] - I = \sigma_\tau^2 \nabla \mathbb{E}_\varepsilon \left[\log \mathbb{E}_\tau \left[p(I - \tau - \varepsilon) \right] \right],$$

where we used the Leibniz rule to interchange the ∇ operator with the expectation. Now we would like to move the expectation over ε inside the log. For this we perform a first order Taylor approximation of the log around $\mathbb{E}_\varepsilon \left[\mathbb{E}_\tau \left[p(I - \tau - \varepsilon) \right] \right]$ and replace the equality sign with approximation, which gives us

$$\mathbb{E}_\varepsilon \left[A_{\sigma_\tau}(I - \varepsilon) \right] - I \approx \sigma_\tau^2 \nabla \log \mathbb{E}_\varepsilon \left[\mathbb{E}_\tau \left[p(I - \tau - \varepsilon) \right] \right].$$

Now we use the fact that consecutive convolution of the density by Gaussian kernels with bandwidths σ_ε^2 and σ_τ^2 is identical to a single convolution by a Gaussian kernel with bandwidth $\sigma_\eta^2 = \sigma_\varepsilon^2 + \sigma_\tau^2$, that is

$$\mathbb{E}_\varepsilon \left[A_{\sigma_\tau}(I - \varepsilon) \right] - I \approx \sigma_\tau^2 \nabla \log \mathbb{E}_\eta \left[p(I - \eta) \right].$$

We now use Equation 6 to rewrite this as

$$\mathbb{E}_\varepsilon \left[A_{\sigma_\tau}(I - \varepsilon) \right] - I \approx \frac{\sigma_\tau^2}{\sigma_\eta^2} \Big(A_{\sigma_\eta}(I) - I \Big),$$

which is the result we wanted. In the paper, we use the specific case where $\sigma_\tau^2 = \sigma_\varepsilon^2 = \frac{1}{2}\sigma_\eta^2$, which leads to Equation 8.

CONVERGENCE OF OUR STOCHASTIC GRADIENT DESCENT

We show the convergence of our algorithm for a single image deblurring example in Figure 8. By using a momentum in our stochastic gradient descent, we are able to avoid oscillations and our reconstruction converges smoothly to the solution.

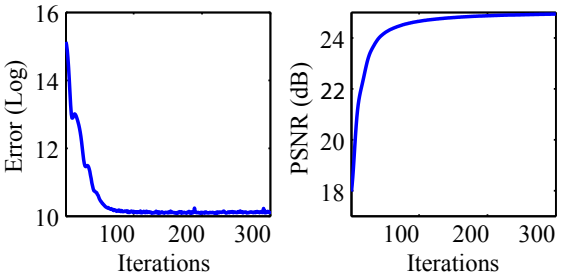

Figure 8: Convergence results of our stochastic objective error (left) and reconstruction PSNR (right) during the iterations.