Image Restoration using Plug-and-Play CNN MAP Denoisers

Siavash Bigdeli[1], David Honzátko[1], Sabine Süsstrunk[2] and L. Andrea Dunbar[1]
[1] *Centre Suisse d'Electronique et de Microtechnique (CSEM), Neuchâtel, Switzerland*
[2] *School of Computer and Communication Sciences, École Polytechnique Fédérale de Lausanne (EPFL), Switzerland*
{siavash.bigdeli, david.honzatko, andrea.dunbar}@csem.ch, sabine.susstrunk@epfl.ch

Keywords: Image Restoration, Image Denoising, MAP, Neural Networks, Deep Learning.

Abstract: Plug-and-play denoisers can be used to perform generic image restoration tasks independent of the degradation type. These methods build on the fact that the Maximum a Posteriori (MAP) optimization can be solved using smaller sub-problems, including a MAP denoising optimization. We present the first end-to-end approach to MAP estimation for image denoising using deep neural networks. We show that our method is guaranteed to minimize the MAP denoising objective, which is then used in an optimization algorithm for generic image restoration. We provide theoretical analysis of our approach and show the quantitative performance of our method in several experiments. Our experimental results show that the proposed method can achieve 70x faster performance compared to the state-of-the-art, while maintaining the theoretical perspective of MAP.

1 INTRODUCTION

Image restoration is a classical signal processing problem with application in diverse domains such as biology, physics, and entertainment. Due to the inherent ambiguity of this task, either the Maximum a Posteriori (MAP) or the Minimum Mean Squared Error (MMSE) estimators are usually used to produce consistent results. With recent advances in deep learning, however, most of the methods employ the MMSE solution due to its simple loss and straightforward training. The MMSE estimator is obtained by minimizing the euclidean distance between the results and the corresponding ground truth. In image denoising and super-resolution, Zhang et al. (Zhang et al., 2017a) showed that training deep neural networks using the MMSE objective can achieve state-of-the-art results. Often the absolute norm is used in place of euclidean norm, which achieves visually more pleasant reconstructions. However, these techniques are usually contaminated with undesired visual artifacts such as over-smoothness (Isola et al., 2017), even though they can achieve better results in terms of peak signal to noise ratio (PSNR).

Another reason to use MAP estimators is the ability to use a single prior model to perform several image restoration tasks as a generic frame work. This is usually done either by learning an explicit image prior, or by using plug-and-play denoisers that inherently learn the prior. We present a novel method to obtain MAP results efficiently using deep neural networks. In contrast to MMSE, the MAP estimator does not enforce correctness of intensity, but it optimizes for the most probable solution – conditioned to the degraded observation. In many applications such as medicine, where the detection overrules the correctness of signal, MAP is a better estimator than MMSE objective.

Performing image restoration model with MAP objective requires using an explicit image prior, which is usually very inefficient. In this paper, we propose an optimization technique with improved efficiency to solve the MAP objective with an explicit image prior model. [1] In summary, the contributions of our work is as follows:

- A novel training strategy to learn a MAP denoiser using an end-to-end model and its neural network parametrization,
- A generic image restoration algorithm based on Alternating direction method of multipliers (ADMM) that uses our neural network and is more efficient in optimizing MAP inference compared to other methods using explicit priors.

The rest of the paper is organized as follows: In Section 2 we discuss relevant work and the challenges of obtaining the MAP solution using neural

[1] The code and the trained models are available at: https://github.com/DawyD/cnn-map-denoiser

networks. Sections 3, 4 discuss the background for image restoration using MAP and MMSE estimators. In Section 5 we describe our new loss function for training neural networks to perform MAP image denoising. Finally, we discuss and demonstrate our experimental results in Section 6 and conclude our findings in Section 7.

2 RELATED WORK

Several methods have been proposed recently that use deep neural network for generic MAP image reconstruction. Among these models, some are based on hand designed features (Ulyanov et al., 2018), and some are based on the explicit density of images (Ulyanov et al., 2018; Bigdeli et al., 2017; Bigdeli and Zwicker, 2017). These techniques mostly use iterative optimization based on gradient descent that take several steps to converge. On the other hand, several plug-and-play methods have been developed for generic image restoration, but only a few methods use the explicit MAP objective (Ahmad et al., 2019). Most of the proposed techniques try to benefit from the fact that a sub-problem of the image reconstruction optimization is a denoising problem and use ad-hoc denoisers to solve this sub-problem. Some of these methods use more classical denoisers such as BM3D or Non-local means (Heide et al., 2014; Venkatakrishnan et al., 2013). More recent approaches use convolutional neural networks, trained as an MMSE model for Gaussian noise removal (Zhang et al., 2017b). Reehorst and Schniter (Reehorst and Schniter, 2018) showed, however, that the denoising sub-problem is of the form of a MAP objective and will not express the original prior if replaced with other types of denoisers like MMSE. This work focuses on developing a network model that can substitute the MAP denoising sub-problem, and use it to perform generic reconstruction of arbitrary degradation models.

Similarly, others have investigated how to provide explicit priors to replace the denoising sub-problem. Chang et al. (Chang et al., 2017) employ a classification network by assuming a discrete manifold for natural images, where all images have the same likelihoods. Sonderby et al. (Sønderby et al., 2016) makes no assumption about the image prior, but their optimization is limited to membership-based data dependencies (such as super-resolution) and cannot be generalized for other tasks after the training.

We summarize the main characteristics of related work in 1. IRCNN (Zhang et al., 2017b) and DIP (Ulyanov et al., 2018) do not have an explicit prior, therefore their solution cannot be expressed in the framework of the MAP image restoration problem. On the other hand DMSP (Bigdeli et al., 2017) uses the slow gradient descent optimization that requires many iterations to converge. We propose an algorithm that can perform up to 70x faster than DMSP and DIP by using the ADMM optimization. In contrast to IRCNN, our prior is explicit and we require only one network throughout the optimization process.

Table 1: Comparison of generic image restoration algorithms and their characteristics. Our method preserves the theoretical guarantees of the conventional MAP estimator by providing an explicit image prior, while also perform an efficient optimization.

Method	Prior	Optim.	Speed	# Nets
DIP	Implicit	GD	Slow	1
IRCNN	Implicit	HQS	Fast	25
DMSP	Explicit	GD	Slow	1
Ours	**Explicit**	**ADMM**	**Fast**	**1**

3 IMAGE RESTORATION VIA MAP OBJECTIVE

We use the standard model of degradation including the noise as

$$y = Kx + \eta, \tag{1}$$

where x is the unknown sharp image, $\eta \sim \mathcal{N}(0, \sigma^2)$ is a noise vector with standard deviation σ, and K is a Toeplitz matrix representing the blur kernel. The MAP estimator intuitively finds the most probable solution of the degradation by maximizing its posterior probability. Using our formulation of the degradation model, this leads to the following optimization

$$\underset{x}{\operatorname{argmin}} \frac{1}{2\sigma^2} ||Kx - y|| - \log p(x), \tag{2}$$

where $p(x)$ is called the image prior and it indicates the probability of the solution x. This objective can be written in the form of

$$\begin{aligned} &\underset{x}{\operatorname{argmin}} \frac{1}{2\sigma^2} ||Kx - y|| - \log p(z) \\ &\text{sbj. to } z = x, \end{aligned} \tag{3}$$

where we substitute a new variable z in place of x in the prior term. This objective has the following augmented Lagrangian (Venkatakrishnan et al., 2013):

$$\frac{1}{2\sigma^2} ||Kx - y|| - \log p(z) + \frac{\rho}{2} ||x - z + \lambda|| - \frac{\rho}{2} ||\lambda||, \tag{4}$$

where λ represents the Lagrange multipliers. Using the ADMM approximation, we can optimize the Lagrangian by iteratively solving the following objectives:

$$\hat{x} := \operatorname*{argmin}_x \frac{1}{2\sigma^2}||Kx-y|| + \frac{\rho}{2}||x-\hat{z}+\lambda|| \tag{5}$$

$$\hat{z} := \operatorname*{argmin}_z -\log p(z) + \frac{\rho}{2}||\hat{x}+\lambda-z|| \tag{6}$$

$$\lambda := \lambda + (\hat{x}-\hat{z}). \tag{7}$$

This strategy has the benefit that the first Equation (5) is quadratic and thus has a closed form solution:

$$\hat{x} := \left(K^TK+\sigma^2\rho\right)^{-1}\left(K^Ty+\sigma^2\rho(\hat{z}-\lambda)\right), \tag{8}$$

which is often solved very efficiently in the Frequency domain as

$$\hat{x} = \mathscr{F}^{-1}\left(\frac{\mathscr{F}(\bar{K})\cdot\mathscr{F}(y)+\sigma^2\rho\mathscr{F}(\hat{z}-\lambda)}{\mathscr{F}(K^TK)+\sigma^2\rho}\right), \tag{9}$$

where $\mathscr{F}$ denotes the Fourier transform, $\mathscr{F}^{-1}$ its inverse, and $\bar{K}$ a flipped and conjugate of the kernel K.

Another attractive point of using the ADMM approximation is that Equation (6) represents a MAP denoising problem, where the objective is to denoise $\hat{x}+\lambda$ given the noise variance $\sigma_R^2 = \frac{1}{\rho}$. This means that no matter the properties of the degradation model (such as blur kernel and noise variance) in Equation (5), we can guide its prior optimization using a separate denoising problem in Equation (6). Next, we show how we can learn an end-to-end MAP denoiser to optimize the objective and to preserve the explicitness of our natural image prior.

4 DENOISING AUTOENCODERS WITH MMSE OBJECTIVE

The ADMM or HQS approximations, have allowed the MAP optimization to be solved efficiently using a denoiser. Given the fact that CNN denoisers have achieved state-of-the-art results (Zhang et al., 2017a), one might use these denoisers to solve Equation (6). A denoising auto encoder (DAE) R is trained to minimize the following MMSE loss:

$$\mathcal{L}_{MSE} = \sum_x ||R(x+\eta_R)-x||, \tag{10}$$

where $\eta_R \sim \mathcal{N}(0,\sigma_R^2)$. Simonceli have showed that the minimizer of this loss is the local mean of the distribution, weighted by the noise distribution. I.e.

$$R^*(y) = y+\sigma_R^2\nabla\log\bar{p}(y), \tag{11}$$

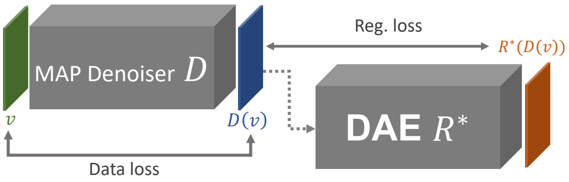


Figure 1: Our end-to-end training of MAP denoisers using trained DAEs: The DAE R is used to enforce the regularization term for optimizing D. At the same time, the output of the network D is kept close to its input using the data loss.

where $\bar{p}$ indicates the density estimate of the natural images, using a Gaussian kernel with the bandwidth σ_R^2. Therefore, we can see that the optimal MMSE denoiser does not correspond to the solution of the MAP denoiser, i.e. Equation (6). However, the difference between input and output of an optimal MMSE denoiser R^*, captures the gradient underlying distribution (Zhang et al., 2017a)

$$R^*(y)-y = \sigma_R^2\nabla\log\bar{p}(y), \tag{12}$$

where $\bar{p}$ indicates the density estimate of the natural images, using a Gaussian kernel with the bandwidth σ_R^2. This property has been used before (Bigdeli et al., 2017) to optimize the MAP objective using iterative gradient descent updates. As we show next, we can still benefit from this property to train a MAP denoiser for Equation (6).

5 LEARNING MAP DENOISERS

As shown before, the derived denoiser in Equation (6) is a MAP estimator and using a MMSE denoiser would instead implicitly change the prior from the natural image distribution to an unknown representation. To preserve the correctness of the prior, we propose to train and use a MAP denoiser instead.

We aim to train a network D that minimizes the objective in Equation (6) by approximating the true distribution with its kernel estimate as above. Inspired by the results of optimal DAEs, we formulate the following loss for our map denoisers,

$$\mathcal{L}_{MAP} = \sum_v \frac{1}{\sigma_R^2}||\bar{v}-\bar{\bar{v}}|| + \frac{\rho}{2}||\bar{v}-v||, \tag{13}$$

where $\bar{v} = D(v)$ and $\bar{\bar{v}} = R^*(\bar{v})$ is the precomputed output of the optimal MMSE DAE R^*.

Proposition 1. *Let us rewrite the cost function in Equation* (6) *as* $f(x,v) = -\log p(x) + \alpha||x-v||$. *Denoting the gradient of the MAP-DAE loss (Equation 13) with respect to the estimator's output as* $\nabla_{\bar{v}}\mathcal{L}_{MAP}$, *then*

$$f(\bar{v},v) > f(\bar{v}-\epsilon\nabla_{\bar{v}}\mathcal{L}_{MAP},v), \tag{14}$$

for small enough step size ϵ.

Proof. We write the gradient of Equation (13) as,

$$\nabla_{\bar{v}}\mathcal{L}_{MAP} = \frac{1}{\sigma_R^2}(\bar{v} - \bar{\bar{v}}) + \frac{\rho}{2}(\bar{v} - v), \quad (15)$$

$$= \frac{1}{\sigma_R^2}\big(\bar{v} - R^*(\bar{v})\big) + \frac{\rho}{2}(\bar{v} - v), \quad (16)$$

$$= -\nabla_{\bar{v}}\log\bar{p}(\bar{v}) + \frac{\rho}{2}(\bar{v} - v), \quad (17)$$

$$= \nabla_{\bar{v}}f(\bar{v}, v), \quad (18)$$

which means that the gradients of the loss in Equation (13) are equal to the gradients of the cost function in Equation (6). Therefore, by minimizing the MAP-DAE loss (Equation 13) with a small enough step size ε, we will also minimize the cost function in Equation (6). □

This means that we can train a network to optimize Equation (6) by minimizing $\mathcal{L}_{MAP}$ with respect to the parameters of the network D. Additionally, this end-to-end training is performed without requiring any paired noisy images and their corresponding ground truth images. Note that the minimization problem in Equation (6) is non-convex (due to the inherent complexity of the underlying natural image distribution) and any optimization scheme can only guarantee convergence to the local minima. However, training a neural network to optimize this objective over all images in the dataset helps in generalization (Zhang et al., 2016), which leads to getting closer to the global optimum.

For the inference time, we feed our network with value $v = \hat{x} + \lambda$, which gives the denoised results $\hat{z} = D^*(v)$ based on the MAP objective Equation (6). The resulting intermediate variable is then used in the rest of the ADMM optimization, which we describe next.

6 EXPERIMENTS

In this section we describe out implementation details and the experimental setup for two image restoration problems: image deblurring and inpainting. We compare our method with the state-of-the-art using the structural similarity image measure (SSIM) and peak signal to noise ratio (PSNR) measures.

6.1 Network Architecture and Training

We have parametrized our denoiser networks based on the DnCNN architecture (Zhang et al., 2017a). We employ this network for both the MAP and the DAE denoisers. The networks consist of 17 convolutional layers with kernel of size 3 × 3, 64 feature channels with ReLU non-linearities. The final receptive field of our network is then 35 × 35. Unlike the original DnCNN training, we did not see any benefit of using batch normalization in our networks, and omit this in our final models.

The training of the two networks are independent: to train the MAP denoiser network, we need an optimally trained DAE. The DAE training is also independent, since it requires noisy input samples generated from our clean dataset. In practice, we train the two networks at the same time, taking advantage of the parallelization to speed up the training. We make sure that the DAE is converged long enough before stopping the MAP denoiser training. In defining the loss terms of the MAP denoiser, we also make sure that the gradients of the regularization term are not propagated through the DAE.

Similar to prior work (Bigdeli and Zwicker, 2017), we observed that the DnCNN network only performs well if it is used with noisy input images, with the same noise standard deviation σ_R as in training. Consequently, we also add noise with standard deviation σ_R to network output $D(v)$ before passing it to the DAE R. This has been shown to approximate the actual objective by minimizing an upper-bound of the actual objective.

We have trained the proposed MAP denoiser on the same dataset as the original DnCNN (Zhang et al., 2017a). We used training patches of size 40 × 40 which were cropped from the Berkeley segmentation dataset (Martin et al., 2001) originally consisting of 400 gray-scale images.

6.2 MAP Denoiser Evaluation

We have evaluated the behaviour of the MAP denoiser using visual inspections. Figure 2 shows the results of the network when applied to a noise-free image. The network moves the input towards the more likely regions of the natural image distribution be removing unlikely patterns. This is done by progressively cancelling out some high frequency details, such as noise or small edges, while preserving the sharpness of the more stronger edges. This leads to the visually cleaner results, which can be controlled by the number of iterations. We can achieve visually cleaner results by controlling the number of update iterations.

6.3 Non-blind Image Deblurring

In this work we focus on application of the proposed image restoration method to non-blind image deblurring. We follow the ADMM approach for image restoration presented in Section 3. We summarize our approach in Algorithm 1, where each iteration of the

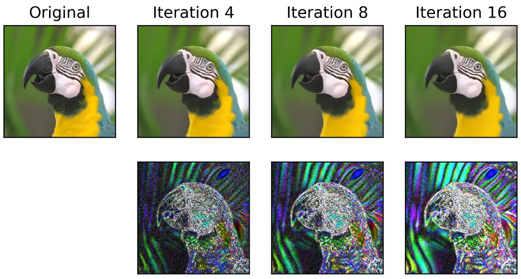


Figure 2: Iterative results of our MAP denoiser network (top row) and the residual w.r.t. the input image (bottom row).

algorithm is as follows: In the first step we optimize $\hat{x}$. For image deblurring this step can be done efficiently in frequency domain using Equation (9). Second, we compute $\hat{z}$ by a single feed-forward step using our trained MAP denoiser D^*. And finally, we update the variable λ and reiterate until convergence.

We have conducted several experiments on various datasets to find the best hyper-parameters for grayscale non-blind image deblurring. We found that using the DAE standard deviation $\sigma_R = 7$ performs best in practice. We also found that setting $\rho = 1/\sigma_R^2$, so that the Equation (5) is balanced, performs best in practice. Such setting leads to best results on Sun dataset (Sun et al., 2013), however, the optimal value for each image might be different. In the experiments we use 75 iterations of the ADMM algorithm. However, as shown in Figure 3, 35 iterations is practically enough for most experiences.

The processing speed and the convergence rate is very important in all iterative approaches. As shown in Figure 3, the proposed method converges very fast compared to DMSP (both methods are solving the same MAP optimization). A color image from BSDS300 can be processed using 35 iterations in about 0.8 seconds on an NVIDIA GTX2080 GPU including the data transfer between the host computer and the device. This makes our method very attractive and more useful in practice. We also show the convergence speed of IRCNN, where the noise level of the denoiser is exponentially decayed from 49 to 15. We note, that the convergence speed of IRCNN may be significantly increased at the cost of lower PSNR.

Algorithm 1: Optimization steps for non-blind image deblurring.

```
input  : Degraded image $y$, blur kernel matrix
         $K$, and noise standard deviation $\sigma$
while not converged do
    1. $\hat{x} = \left(K^T K + \sigma^2\rho\right)^{-1}\left(K^T y + \sigma^2\rho(\hat{z} - \lambda)\right)$
    2. $\hat{z} = D^*(\hat{x} + \lambda)$
    3. $\lambda = \lambda + (\hat{x} - \hat{z})$
output: MAP estimate output image $\hat{x}$.
```

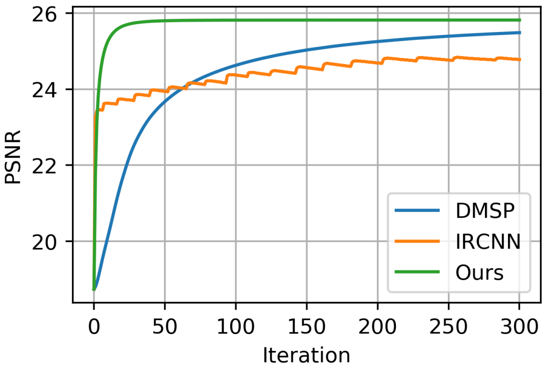


Figure 3: Convergence speed comparison on an image from BSDS300. Note that DMSP and the proposed method try to optimize the same objective. Using the ADMM approach, we can speed-up the optimization up to 70x faster.

6.3.1 Optimization

To be consistent with other methods, for all the experiments we first blur the ground-truth test image with the blur kernel. The blurring is done by convolution with a flipped version of a kernel and only the valid area is preserved. We also add a Gaussian noise with the desired standard deviation σ. After the restoration, we measure the PSNR on the valid area.

Since we optimize $\hat{x}$ in the frequency domain, the proposed method works the best when the degradation is done using circular convolution. However, in case we receive only the valid area, assuming the circular convolution leads to high error on the boarders which gets propagated towards the center in subsequent iterations. To alleviate this behaviour and maintain the benefit of processing in frequency domain, we first pad the degraded image by replicating the edges to the original size. Furthermore, in each iteration, we first estimate the non-valid area of the y by performing a cyclic convolution on the current estimate of $\hat{x}$ with the kernel. Such treatment leads to plausible results and yet it is fast enough due to the processing in the frequency domain. Figure 4 shows the iterative results of our optimization, where the image is getting sharper using the proposed denoising prior.

6.3.2 Datasets

We test the method on two different datasets and report the results in Table 2 and Table 3. The first one is the Sun dataset (Sun et al., 2013) that consists of 80 images with the longest dimension being 924 pixels. The dataset contains 8 blur kernels of various sizes to simulate the degraded images. The second dataset we use for evaluation is the BSDS300 (Martin et al., 2001). It consists of 300 images of size 321 × 481, which we convert to gray-scale. We test

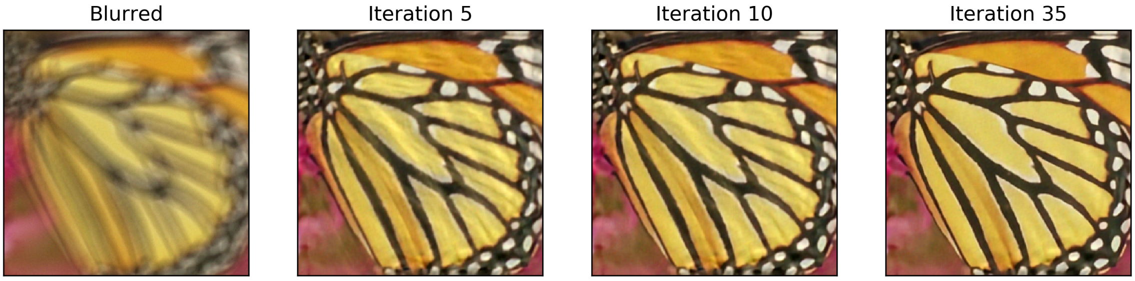


Figure 4: Deblurring results of the ADMMM iterations using the proposed algorithm. Our MAP denoiser network encourages sharp image edges and removes undesired artifacts.

Table 2: Quantitative comparisons for non-blind image deblurring using two datasets in terms of PSNR.

	Sun (Sun et al., 2013)				BSDS300 (Martin et al., 2001)			
Method $\sigma \rightarrow$	2.55	5.10	7.65	10.2	2.55	5.10	7.65	10.2
FD (Krishnan and Fergus, 2009)	30.79	28.90	27.86	27.14	24.44	23.24	22.64	22.07
EPLL (Zoran and Weiss, 2011)	**32.05**	29.60	28.25	27.34	25.38	23.53	22.54	21.91
CSF (Schmidt and Roth, 2014)	30.88	28.60	27.65	26.97	24.73	23.60	22.88	22.44
TNRD (Chen and Pock, 2016)	30.03	28.79	28.04	27.54	24.17	23.76	23.27	22.87
DAEP (Bigdeli and Zwicker, 2017)	31.76	29.31	28.01	27.16	25.42	23.67	22.78	22.21
IRCNN (Zhang et al., 2017b)	31.80	**30.13**	28.93	28.09	25.60	24.24	23.42	22.91
GradNet 7S (Jin et al., 2017)	31.75	29.31	28.04	27.54	25.57	24.23	23.46	22.94
DMSP (Bigdeli et al., 2017)	29.41	29.04	28.56	27.97	25.69	24.45	**23.60**	**22.99**
Ours	31.00	29.96	2**8.96**	**28.13**	**26.18**	**24.52**	23.51	22.79

Table 3: Comparison of SSIM scores for non-blind image deblurring using two datasets.

	Sun (Sun et al., 2013)				BSDS300 (Martin et al., 2001)			
Method $\sigma \rightarrow$	2.55	5.10	7.65	10.2	2.55	5.10	7.65	10.2
FD (Krishnan and Fergus, 2009)	0.851	0.787	0.744	0.714	0.664	0.577	0.534	0.492
EPLL (Zoran and Weiss, 2011)	**0.880**	0.807	0.758	0.721	0.712	0.590	0.521	0.476
CSF (Schmidt and Roth, 2014)	0.853	0.752	0.718	0.681	0.693	0.612	0.558	0.521
TNRD (Chen and Pock, 2016)	0.844	0.790	0.750	0.739	0.690	0.631	0.589	0.550
GradNet 7S (Jin et al., 2017)	0.873	0.798	0.750	0.733	0.731	0.653	0.595	0.552
DMSP (Bigdeli et al., 2017)	-	-	-	-	0.740	**0.671**	**0.611**	0.563
Ours	0.829	**0.817**	**0.788**	**0.758**	0.768	0.668	0.595	0.540

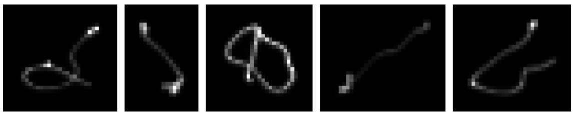


Figure 5: Blur kernels used in our experiments (Schelten et al., 2015).

this dataset with the 5 large blur kernels as in (Schelten et al., 2015). These kernels visualized in Figure 5, have complex structures and lead to challenging deblurring tasks.

Figure 6, shows a visual comparison between our method and several other state-of-the-art techniques. Our method is able to capture the sharp structures of the image, while remaining highly efficient compared to other algorithms.

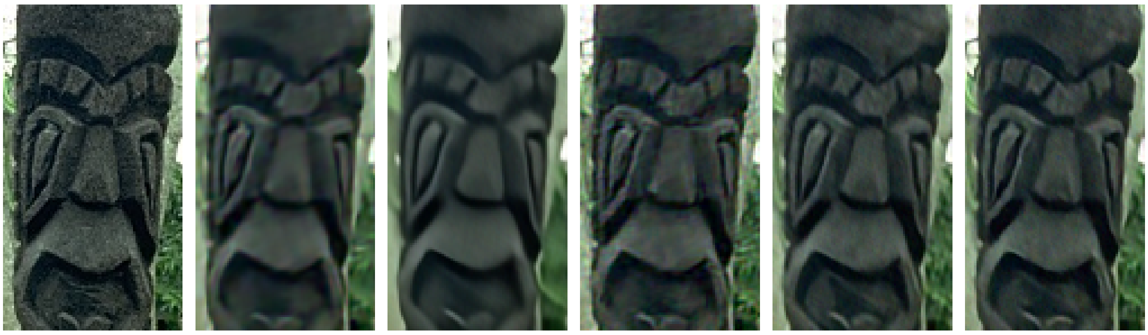
6.4 Image Inpainting

In addition to the deblurring method, we have also tested the proposed method on the inpainting problem. We follow the ADMM approach for image restoration presented in Section 3. To speed up the convergence, we initialize our solution $\hat{x}$ with image inpainted using a median filter.

6.4.1 Optimization

We conduct the test with 80% of the missing pixels where we add Gaussian noise with $\sigma = 12$ prior to the remaining pixels. The tested methods are provided both with the degraded image and with the mask specifying which pixels were dropped.

In contrast to the image deblurring, we use 300 it-



(a) Ground truth (b) EPLL (c) DAEP (d) GradNet 7S (e) DMSP (f) Ours

Figure 6: Visual Comparison of the selected methods for the task of non-blind image deblurring.

Table 4: Inpainting results (PSNR) for 80% of missing pixels and noise with std. deviation of $\sigma = 12$.

Method	cameraman	house	peppers	Lena	Barbara	boat	hill	couple
P&P-BM3D (Venkatakrishnan et al., 2013)	24.43	30.78	26.56	29.47	24.12	26.53	27.44	26.71
IRCNN (Zhang et al., 2017b)	**24.59**	30.19	26.94	29.52	**25.49**	26.58	27.55	26.62
IDBP-BM3D (Tirer and Giryes, 2018)	24.51	**31.14**	26.79	29.69	25.06	26.64	27.61	26.77
IDBP-CNN (Tirer and Giryes, 2018)	24.14	30.92	**27.17**	**29.80**	23.61	26.78	27.70	**26.80**
Ours	23.17	30.32	26.01	29.44	24.30	**26.81**	**28.40**	26.77

Table 5: Inpainting results (SSIM) for 80% of missing pixels and noise with std. deviation of $\sigma = 12$.

Method	cameraman	house	peppers	Lena	Barbara	boat	hill	couple
P&P-BM3D (Venkatakrishnan et al., 2013)	0.774	0.839	0.807	0.818	0.705	0.707	0.683	0.734
IRCNN (Zhang et al., 2017b)	0.781	0.835	0.813	0.820	**0.758**	0.723	0.706	0.736
IDBP-BM3D (Tirer and Giryes, 2018)	0.775	**0.844**	0.816	0.824	0.738	0.712	0.691	0.738
IDBP-CNN (Tirer and Giryes, 2018)	**0.786**	0.843	**0.830**	**0.836**	0.731	**0.738**	**0.714**	**0.752**
Ours	0.757	0.808	0.814	0.803	0.741	0.721	0.707	0.735

erations of the ADMM algorithm. Moreover, in each iteration, we use 200 steps of gradient descent optimization to approximate Equation (5) of the ADMM.

6.4.2 Dataset

We evaluate the methods using PSNR and SSIM scores on classical dataset consisting of images of *cameraman*, *house*, *peppers*, *Lena*, *Barbara*, *boat*, *hill*, and *couple*.

Tables 4, 5 show the PSNR and SSIM scores of our method compared to the state-of-the-art. Again, our method can achieve state-of-the-art results for the task of image inpainting. Note that we use the same MAP denoising network as in the previous experiments. We also visualize qualitative results in Figure 7, where our method can successfully recover 80% of the missing pixels from the input image.

7 CONCLUSIONS

We discussed the challenges of using the plug-and-play denoising in ADMM optimization for MAP estimation. We presented an approach for learning an end-to-end neural network to perform MAP denoising. We used this network in an ADMM framework to perform generic image restoration tasks. Our theoretical results show that we can guarantee to minimize the MAP objective using the proposed training strategy. And our experimental validation showed that our method has significant improvement in speeding up the MAP optimization, compared to other explicit approaches. Over all, other method supports the theoretical guarantees of MAP estimation, and at the same time benefits from the fast performance of other approaches without such guarantee.

ACKNOWLEDGEMENTS

Financial support from the CSEM Data Program is grateully acknowledged. Authors would also like to thank Laura Bujouves for her insightful comments.

Figure 7: Image inpainting example. **Left:** Original image, **Middle:** Degraded image with 80% of the pixels set to zero, **Right:** Inpainted image using our method.

REFERENCES

Ahmad, R., Bouman, C. A., Buzzard, G. T., Chan, S., Reehorst, E. T., and Schniter, P. (2019). Plug and play methods for magnetic resonance imaging. *arXiv preprint arXiv:1903.08616*.

Bigdeli, S. A. and Zwicker, M. (2017). Image restoration using autoencoding priors. *arXiv preprint arXiv:1703.09964*.

Bigdeli, S. A., Zwicker, M., Favaro, P., and Jin, M. (2017). Deep mean-shift priors for image restoration. In *Advances in Neural Information Processing Systems*, pages 763–772.

Chang, J.-H. R., Li, C.-L., Poczos, B., Kumar, B. V., and Sankaranarayanan, A. C. (2017). One network to solve them all-solving linear inverse problems using deep projection models. In *ICCV*, pages 5889–5898.

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*, 39(6):1256–1272.

Heide, F., Steinberger, M., Tsai, Y.-T., Rouf, M., Pająk, D., Reddy, D., Gallo, O., Liu, J., Heidrich, W., Egiazarian, K., et al. (2014). Flexisp: A flexible camera image processing framework. *ACM Transactions on Graphics (TOG)*, 33(6):231.

Isola, P., Zhu, J.-Y., Zhou, T., and Efros, A. A. (2017). Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134.

Jin, M., Roth, S., and Favaro, P. (2017). Noise-blind image deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3510–3518.

Krishnan, D. and Fergus, R. (2009). Fast image deconvolution using hyper-laplacian priors. In *Advances in neural information processing systems*, pages 1033–1041.

Martin, D., Fowlkes, C., Tal, D., and Malik, J. (2001). A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision*, volume 2, pages 416–423.

Reehorst, E. T. and Schniter, P. (2018). Regularization by denoising: Clarifications and new interpretations. *IEEE Transactions on Computational Imaging*, 5(1):52–67.

Schelten, K., Nowozin, S., Jancsary, J., Rother, C., and Roth, S. (2015). Interleaved regression tree field cascades for blind image deconvolution. In *2015 IEEE Winter Conference on Applications of Computer Vision*, pages 494–501. IEEE.

Schmidt, U. and Roth, S. (2014). Shrinkage fields for effective image restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2774–2781.

Sønderby, C. K., Caballero, J., Theis, L., Shi, W., and Huszár, F. (2016). Amortised map inference for image super-resolution. *arXiv preprint arXiv:1610.04490*.

Sun, L., Cho, S., Wang, J., and Hays, J. (2013). Edge-based blur kernel estimation using patch priors. In *IEEE International Conference on Computational Photography (ICCP)*, pages 1–8. IEEE.

Tirer, T. and Giryes, R. (2018). Image restoration by iterative denoising and backward projections. *IEEE Transactions on Image Processing*, 28(3):1220–1234.

Ulyanov, D., Vedaldi, A., and Lempitsky, V. (2018). Deep image prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9446–9454.

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

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2016). Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*.

Zhang, K., Zuo, W., Chen, Y., Meng, D., and Zhang, L. (2017a). Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155.

Zhang, K., Zuo, W., Gu, S., and Zhang, L. (2017b). Learning deep cnn denoiser prior for image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3929–3938.

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. IEEE.