This ICCV Workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

DeepRED: Deep Image Prior Powered by RED

Gary Mataev
The Computer-Science Department
The Technion - Israel Institute of Technology
garym24@campus.technion.ac.il

Peyman Milanfar
Google Research
milanfar@google.com

Michael Elad
Google Research
melad@google.com

Abstract

Inverse problems in imaging are extensively studied, with a variety of strategies, tools, and theory that have been accumulated over the years. Recently, this field has been immensely influenced by the emergence of deep-learning techniques. One such contribution, which is the focus of this paper, is the Deep Image Prior (DIP) work by Ulyanov, Vedaldi, and Lempitsky (2018). DIP offers a new approach towards the regularization of inverse problems, obtained by forcing the recovered image to be synthesized from a given deep architecture. While DIP has been shown to be quite an effective unsupervised approach, its results still fall short when compared to state-of-the-art alternatives.

In this work, we aim to boost DIP by adding an explicit prior, which enriches the overall regularization effect in order to lead to better-recovered images. More specifically, we propose to bring-in the concept of Regularization by Denoising (RED), which leverages existing denoisers for regularizing inverse problems. Our work shows how the two (DIP and RED) can be merged into a highly effective unsupervised recovery process while avoiding the need to differentiate the chosen denoiser, and leading to very effective results, demonstrated for several tested problems.

1. Introduction

Inverse problems in imaging center around the recovery of an unknown image $\mathbf{x}$ based on given corrupted measurement $\mathbf{y}$. These problems are typically posed as energy minimization tasks, drawing their mathematical formulation from a statistical (Bayesian) modeling of the posterior distribution, $P(\mathbf{x}|\mathbf{y})$. As inverse problems tend to be ill-posed, a key in the success of the recovery process is the choice of the regularization, which serves as the image prior that stabilizes the degradation inversion, and directs the outcome towards a more plausible image.

The broad field of inverse problems in imaging has been extensively explored in the past several decades. This vast work has covered various aspects, ranging from the formulation of such problems, through the introduction of diverse ways to pose and use the regularization, and all the way to optimization techniques for minimizing the obtained energy function. This massive research effort has led to one of the most prominent fields in the broad arena of imaging sciences, and to many success stories in applications, treating problems such as denoising, deblurring, inpainting, super-resolution, tomographic reconstruction, and more.

The emergence of deep-learning a decade ago brought a revolution to the way machine learning is practiced. At first, this feat mostly focused on supervised classification tasks, leading to state-of-the-art results in challenging recognition applications. However, this revolution found its way quite rapidly to inverse problems in imaging, due to the ability to consider these as specific regression problems. The practiced rationale in such schemes is as follows: Given many examples of pairs of an original image and its corrupted version, one could learn a deep network to match the degraded image to its source. This became a commonly suggested and very effective path to the above-described classical Bayesian alternative, see e.g., [34, 12, 32, 20, 19, 13, 18, 35, 30, 33, 26, 16].

The recent work by Ulyanov et. al. [27, 28] is an exceptional contribution in the intersection between inverse problems and deep-learning. This work presents the *Deep Image Prior* (DIP) method, a new strategy for handling the regularization task in inverse problems. Rather than taking the supervised avenue, as most earlier methods do, DIP suggests to use the deep network itself as the regularizer to the inverse problem. More specifically, DIP removes the explicit regularization, and replaces it by the assumption that the unknown image $\mathbf{x}$ should be a generated image from a learned network. DIP fits the network's parameters for the corrupted image, this way adapting it per each image to be treated. Our special interest in this work stems from the brilliant idea of implicitly using the structure of a network[1] to obtain a regularization effect in recovering $\mathbf{x}$.

While DIP has been shown to be quite effective, and demonstrated successfully on several inverse problems (de-

[1] ... and possibly the optimization strategy as well.

noising, JPEG artifact removal, inpainting, and super-resolution), its results still fall short when compared to unsupervised state-of-the-art alternatives. This brings up the idea to offer an extra boost to DIP by returning the *explicit* regularization, so as to enrich the implicit one, and this way lead to better recovered images. The natural question is, of-course, which regularization to use, as there are so many options available. Interestingly, the need to bring back an extra regularization came up quite recently in the work reported in [14], where Total-Variation [24] has been used and shown to lead to improved recovery results. Another relevant work along these lines is [15], in which Stein's Unbiased Risk Estimator (SURE) is leveraged to yield an effective regularization expression for boosting DIP.

Contributions: In this work we propose to bring in the recently introduced concept of Regularization by Denoising (RED) [21] and merge it with DIP. The special appeal of RED is threefold: (i) RED produces a wide family of regularization options, each with its own strengths; (ii) RED can use any denoiser[2]; and (iii) RED is superior to many other regularization schemes. In this work we use NLM [3] and BM3D [5] as the two denoisers within RED. Both bring an extra force that does not exist in DIP, due to their reliance on the self-similarity property. This adds a non-locality flavor to our overall recovery algorithm, which complements the DIP architecture regularization effect.

A special challenge in our work is finding a way to train the new compound objective, DIP+RED, while avoiding an explicit differentiation of the denoising function. This is achieved using the Alternating Directions Methods of Multipliers (ADMM) [2], which enjoys an extra side-benefit: a stable recovery with respect to the stopping rule employed. The proposed scheme, termed DeepRED, is tested on image denoising, single image super-resolution, and image deblurring, showing the clear benefit that RED provides. The obtained results exhibit marked improvements, both with respect to the native RED as reported in [21], and DIP itself. Indeed, DeepRED shows state-of-the-art results among unsupervised methods for the image deblurring task (when compared to [31]).

This paper is organized as follows: The next section presents background material on the inverse problems we target in this work, as well as describing DIP and RED, the two pillars of this work. In Section 3 we present the combined DeepRED scheme, and develop the ADMM algorithm for its training. Section 4 presents our experimental results, validating the benefits of the additional explicit regularization on a series of inverse problems. We conclude the paper in Section 5 by summarizing its message and results and proposing potential future research directions.

[2]And this includes a TV-based denoising, implying that the work reported in [14] can be considered as a special case of our approach. Deep-learning based denoisers, e.g. [34, 12, 32, 20] can be used as well.

2. Background

In this section, we give more details on the inverse problems we target, and briefly present both the Deep Image Prior (DIP) approach and the concept of Regularization by Denoising (RED).

2.1. Inverse Problems of Interest

Within the broad field of inverse problems, our work considers the case where the measurement $\mathbf{y}$ is given by $\mathbf{y} = \mathbf{H}\mathbf{x} + \mathbf{v}$, where $\mathbf{H}$ is any known linear degradation matrix, and $\mathbf{v}$ is an Additive White Gaussian Noise (AWGN). The recovery of $\mathbf{x}$ from $\mathbf{y}$ could be obtained by solving

$$\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{H}\mathbf{x} - \mathbf{y}\|_2^2 + \lambda\rho(\mathbf{x}), \tag{1}$$

where $\rho(\mathbf{x})$ serves as the chosen regularization term. By modifying the operator $\mathbf{H}$, we can switch between several popular problems in image recovery:

- Denoising is obtained for $\mathbf{H} = \mathbf{I}$,
- Deblurring (deconvolution) assumes that $\mathbf{H}$ is a convolution filter,
- Inpainting is obtained when $\mathbf{H}$ is built as the identity matrix with missing rows referring to missing samples,
- Super-resolution refers to a matrix $\mathbf{H}$ that represents both a blur followed by a sub-sampling, and
- Tomographic reconstruction assumes that $\mathbf{H}$ applies the Radon projections or portion thereof.

We stress that the paradigm presented in this paper could be easily extended to handle other types of noise (e.g., Laplace, Poisson, Gamma, or other models). This should be done by replacing the expression $\|\mathbf{H}\mathbf{x} - \mathbf{y}\|_2^2$ by the minus log-likelihood of the appropriate distribution, as done in [25, 23, 8] in the context of the Poisson noise. Note that our view on this matter is somewhat different from the view of the authors of [27], who suggest to handle other types of noise while still using the L_2 penalty.

2.2. Deep Image Prior (DIP)

DIP embarks from the formulation posed in Equation (1), and starts by removing the regularization term $\rho(\mathbf{x})$. The idea is to find the minimizer of the first term, $\|\mathbf{H}\mathbf{x} - \mathbf{y}\|_2^2$. However, this amounts to the Maximum-Likelihood Estimate (MLE), which is known to perform very poorly for the inverse problem cases considered in this paper. DIP overcomes this weakness by assuming that the unknown, $\mathbf{x}$, should be the output of a deep network, $\mathbf{x} = \mathbf{T}_\Theta(\mathbf{z})$, where $\mathbf{z}$ is a fixed random vector, and Θ stands for the network's parameters to be learned. Thus, DIP suggests to solve

$$\min_{\Theta} ||\mathbf{H}\mathbf{T}_\Theta(\mathbf{z}) - \mathbf{y}||_2^2, \tag{2}$$

and presents $\mathbf{T}_\Theta(\mathbf{z})$ as the recovered image.

Observe that the training of Θ itself serves also as the inference, i.e., this raining is the recovery process, and this should be done for each input image separately and independently. This procedure is "unsupervised" in the sense that no ideal outcome (label) is presented to the learning. Rather, the training is guided by the attempt to best match the output of the network to the measured and corrupted image. Over-fitting in this case amounts to a recovery of $\mathbf{x}$ that minimizes the above L_2 expression while being of poor visual quality. This is avoided due to the implicit regularization imposed by the architecture of the network $\mathbf{T}_\Theta(\mathbf{z})$ and the early stopping.[3] Indeed, the fact that DIP operates well and recovers high quality images could be perceived as a manifestation of the "correctness" of the chosen architecture to represent image synthesis.

In practice, DIP performs very well. The work in [27] reports several sets of experiments on (i) image denoising – leading to performance that is little bit weaker than CBM3D [5] and better than NLM [3]; (ii) Single Image Super-Resolution – leading to substantially better results than bicubic interpolation and TV-based restoration, but inferior to the learning based methods [10, 11]; and (iii) Inpainting – in which the results are shown to be much better than CSC-based ones [17].

2.3. Regularization by Denoising (RED)

The quest for an effective regularization for inverse problems in imaging has played a central role in the vast progress of this field. Various ideas were brought to serve the construction of $\rho(\mathbf{x})$ in Equation (1), all aiming to identify sources of inner structure in visual data. These may rely on piecewise spatial smoothness (e.g., [24]), self-similarity across different positions and scales (e.g., [3, 22]), sparsity with respect to a properly chosen transform or representation (e.g. [5]), and more.

Among the various inverse problems mentioned above, denoising has gained a unique position due to its relative simplicity. This problem has become the de-facto testbed for exploring new regularization ideas. As a consequence, many highly effective and trustworthy denoising algorithms were developed in the past two decades. This brought a surprising twist in the evolution of regularizers, turning the table and seeking a way to construct a regularization by using denoising algorithms. The plug-and-play-prior [29] and the Regularization by Denoising (RED) [21] are two prime such techniques for turning a denoiser into a regularization. RED suggests to use the following as the regularization function:

$$\rho(\mathbf{x}) = \frac{1}{2}\mathbf{x}^T(\mathbf{x} - f(\mathbf{x})), \tag{3}$$

where $f(\cdot)$ is a denoiser of choice. We will not dwell on the rationale of this expression, beyond stating its close resemblance to a spatial smoothness term. Amazingly, under mild conditions[4] on $f(\cdot)$, two key and highly beneficial properties are obtained: (i) The gradient of $\rho(\cdot)$ w.r.t. $\mathbf{x}$ is simple and given by $\nabla\rho(\mathbf{x}) = \mathbf{x} - f(\mathbf{x})$, which avoids differentiating the denoiser function; and (ii) $\rho(\cdot)$ is a convex functional. The work reported in [21] introduced the concept of RED and showed how to leverage these two properties in order to obtain an effective regularization for various inverse problems. Our goal in this work is to bring this method to DIP, with the hope to boost its performance.

3. The Proposed DeepRED Scheme

3.1. Algorithm Derivation

Merging DIP[5] and RED, our objective function becomes

$$\min_{\mathbf{x},\Theta} \frac{1}{2}\|\mathbf{H}\mathbf{T}_\Theta(\mathbf{z}) - \mathbf{y}\|_2^2 + \frac{\lambda}{2}\mathbf{x}^T(\mathbf{x} - f(\mathbf{x})) \quad s.t. \quad \mathbf{x} = \mathbf{T}_\Theta(\mathbf{z}). \tag{4}$$

Note that a simple strategy is to avoid the use of $\mathbf{x}$ and define the whole optimization w.r.t. the unknowns Θ. This calls for solving

$$\min_{\Theta} \quad \frac{1}{2}\|\mathbf{H}\mathbf{T}_\Theta(\mathbf{z}) - \mathbf{y}\|_2^2 + \frac{\lambda}{2}\left[\mathbf{T}_\Theta(\mathbf{z})\right]^T\left(\left[\mathbf{T}_\Theta(\mathbf{z})\right] - f\left(\left[\mathbf{T}_\Theta(\mathbf{z})\right]\right)\right).$$

While this may seem simpler, it is in fact leading to a near dead-end, since back-propagating over $\mathbf{T}$ calls for the differentiation of the denoising function $f(\cdot)$. For most denoisers this would be a daunting task that must be avoided. As we have explained above, under mild conditions, RED enjoys the benefit of avoiding such a direct differentiation, and we would like to leverage this property here.

The remedy to this problem comes in the form of the Alternating Directions Method of Multipliers (ADMM) [2]. Starting with Equation (4), we turn the constraint into a penalty using the Augmented Lagrangian (AL) [1]:

$$\min_{\mathbf{x},\Theta} \quad \frac{1}{2}\|\mathbf{H}\mathbf{T}_\Theta(\mathbf{z}) - \mathbf{y}\|_2^2 + \frac{\lambda}{2}\mathbf{x}^T(\mathbf{x} - f(\mathbf{x})) + \frac{\mu}{2}\|\mathbf{x} - \mathbf{T}_\Theta(\mathbf{z})\|_2^2 - \mu\mathbf{u}^T(\mathbf{x} - \mathbf{T}_\Theta(\mathbf{z})). \tag{5}$$

In this expression $\mathbf{u}$ stands for the Lagrange multipliers vector for the set of equality constraints, and μ is a free parameter to be chosen. Merging the last two terms, we get the

[3] The number of iterations is bounded so as to avoid overfitting.

[4] The function $f(\cdot)$ should be differentiable, have a symmetric Jacobian, satisfy a local homogeneity condition, and be passive.

[5] Note that all the derivations and algorithms proposed in this paper are applicable just as well to *Deep-Decoder* [9], an appealing followup work to DIP that promotes a simpler architecture for $\mathbf{T}_\Theta(\mathbf{z})$.

scaled form of the AL [1],

$$\min_{\mathbf{x},\Theta} \quad \frac{1}{2}\|\mathbf{H}\mathbf{T}_\Theta(\mathbf{z})-\mathbf{y}\|_2^2+\frac{\lambda}{2}\mathbf{x}^T\left(\mathbf{x}-f(\mathbf{x})\right) + \frac{\mu}{2}\|\mathbf{x}-\mathbf{T}_\Theta(\mathbf{z})-\mathbf{u}\|_2^2. \tag{6}$$

The ADMM algorithm amounts to a sequential update of the three unknowns in this expression: Θ, $\mathbf{x}$, and $\mathbf{u}$. Fixing $\mathbf{x}$ and $\mathbf{u}$, the update of Θ is done by solving

$$\min_{\Theta} \frac{1}{2}\|\mathbf{H}\mathbf{T}_\Theta(\mathbf{z})-\mathbf{y}\|_2^2+\frac{\mu}{2}\|\mathbf{x}-\mathbf{T}_\Theta(\mathbf{z})-\mathbf{u}\|_2^2, \tag{7}$$

which is very close in spirit to the optimization done in DIP (using back-propagation), modified by a proximity regularization that forces $\mathbf{T}_\Theta(\mathbf{z})$ to be close to $\mathbf{x}-\mathbf{u}$. This proximity term provides as an additional stabilizing and robustifying effect to the DIP minimization.

Fixing Θ and $\mathbf{u}$, $\mathbf{x}$ should be updated by solving

$$\min_{\mathbf{x}} \frac{\lambda}{2}\mathbf{x}^T\left(\mathbf{x}-f(\mathbf{x})\right)+\frac{\mu}{2}\|\mathbf{x}-\mathbf{T}_\Theta(\mathbf{z})-\mathbf{u}\|_2^2. \tag{8}$$

This is a classic RED objective [21], representing a denoising of the image $\mathbf{T}_\Theta(\mathbf{z})+\mathbf{u}$, and we suggest solving it in one of two ways: The first option is using the fixed-point strategy by zeroing the derivative of the above w.r.t. $\mathbf{x}$, and exploiting the fact that $\nabla\rho(\mathbf{x})=\mathbf{x}-f(\mathbf{x})$. This leads to

$$\lambda\left(\mathbf{x}-f(\mathbf{x})\right)+\mu\left(\mathbf{x}-\mathbf{T}_\Theta(\mathbf{z})-\mathbf{u}\right)=0. \tag{9}$$

Assigning indices to the above equation,

$$\lambda\left(\mathbf{x}_{j+1}-f(\mathbf{x}_j)\right)+\mu\left(\mathbf{x}_{j+1}-\mathbf{T}_\Theta(\mathbf{z})-\mathbf{u}\right)=0 \tag{10}$$

leads to the update formula

$$\mathbf{x}_{j+1}=\frac{1}{\lambda+\mu}\left(\lambda f(\mathbf{x}_j)+\mu(\mathbf{T}_\Theta(\mathbf{z})+\mathbf{u})\right). \tag{11}$$

Applying this iterative update several times provides the needed update for $\mathbf{x}$. An alternative approach for updating $\mathbf{x}$ is a simpler steepest-descent, using the above described gradient. Thus, the update equation would be

$$\mathbf{x}_{j+1}=\mathbf{x}_j-c\left[\lambda\left(\mathbf{x}_j-f(\mathbf{x}_j)\right)+\mu\left(\mathbf{x}_j-\mathbf{T}_\Theta(\mathbf{z})-\mathbf{u}\right)\right], \tag{12}$$

and c should be chosen so as to guarantee a descent.

As for the Lagrange multipliers vector $\mathbf{u}$, its update is much easier, given by $\mathbf{u}_{k+1}=\mathbf{u}_k-\mathbf{x}+\mathbf{T}_\Theta(\mathbf{z})$, as emerging from the AL method [2, 1]. Algorithm 1 summarizes the steps to be taken to apply this overall algorithm for handling the DeepRED objective minimization.

Algorithm 1: ADMM Minimization of the DeepRED objective (Equation (4)).

Result: Obtain the restored image $\mathbf{x}$
Parameters:
- λ - the RED regularization strength
- μ - the ADMM free parameter
- Steepest-descent parameters for updating Θ
- c - Step-size in the SD update of $\mathbf{x}$
- J - number of inner iterations for the update of $\mathbf{x}$

Init: Set $k=0$, $\mathbf{u}_0=0$, $\mathbf{x}_0=\mathbf{y}$, and set Θ_0 randomly
while *not converged* **do**
  Update Θ_{k+1}**:** Solve Equation (7) using steepest descent and back-propagation
  Update $\mathbf{x}_{k+1}$**:** Apply either the fixed point (Eq.(11)) or the SD (Eq.(12)) for J iterations
  Update $\mathbf{u}_{k+1}$**:** $\mathbf{u}_{k+1}=\mathbf{u}_k-\mathbf{x}_{k+1}+\mathbf{T}_{\Theta_{k+1}}(\mathbf{z})$
  k=k+1
end

3.2. Implementation Details

The original DIP algorithm [27] offers three features that influence the output quality of the restored images. The first is an early stopping, which prevents the network from overfitting to the measurements. The second is a smoothing applied on the outcome of the last iterations, and the third is an averaging over separate runs with a different random vector $\mathbf{z}$. Our tests implement all these as well, but we emphasize that the early stopping is relevant in our DeepRED scheme only for saving computations, as the explicit regularization robustifies the recovery from the risk of overfitting.

Due to the involvement of a highly non-linear system $\mathbf{T}_\Theta(\mathbf{z})$ in our overall optimization, no convergence guarantees can be provided. In addition, when using denoisers that violate the conditions posed in [21], the denoising residual $\mathbf{x}-f(\mathbf{x})$ is no longer the exact derivative of the RED prior. Nevertheless, as we show in the experimental results, tendency for a consistent descent and a convergence are obtained empirically.

In our tests we have chosen $J=1$, which means that the denoiser $f(\cdot)$ is applied once in each ADMM round of updates. The heaviest loads in our algorithm are both the update of Θ and the activation of the denoiser. Fortunately, we can speed the overall run of the algorithm by adopting the following two measures: (i) The denoiser and the update of Θ can be run in parallel, as shown in Figure 1; and (ii) We apply the denoiser once every few outer iterations of the ADMM in order to save run-time.

4. Experimental Results

We now present a series of experiments in which we test the proposed DeepRED scheme. We consider three applica-

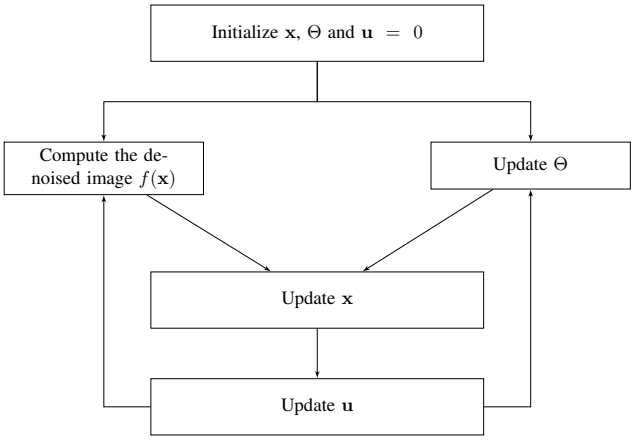


Figure 1: The denoiser can be applied in parallel to the update of Θ in order to speed-up the overall algorithm.

tions: image denoising and Single Image Super-Resolution (SISR), which were also studied in [27], and image deblurring, following the experiments reported in [21] and [31]. Our aim in all these experiments is to show that (i) DeepRED behaves well numerically; (ii) it is better than both DIP and RED; (iii) it performs better than DIP+TV [12]; and (iv) DeepRED is the among the best unsupervised restoration algorithms, taking the lead in image deblurring.

In all the reported tests the same network as in [27] is used with an i.i.d. uniform ($\sim [0, 0.1]$) random input tensor of size $32 \times W \times H$, where $W \times H$ is the size of the output image to synthesize. Table 1 summarizes the various parameters used for each application. These include the additional noise perturbation standard-deviation (σ_{noise}), the learning rate (LR), the employed denoiser and the noise level fed to it σ_f, the values of λ and μ (see 1), and the number of iterations. All the reported results for DIP are obtained by directly running the released code. We note that there are slight changes between the values we get and the ones reported in [27].

When using DeepRED, we employ the Fixed-Point Strategy as described in 1, and apply the denoiser once ($J = 1$) every 10 iterations. Following [27] and [7], in the deblurring and super-resolution experiments, the results are compared on the luminance channel, whereas the denoising results are evaluated with all three channels.

	Parameters						
	σ_{noise}	LR	denoiser	σ_f	λ	μ	iter.
Denoising	0.033	0.008	NLM	3	0.5	0.5	6000
SISR x4	0.02	0.001	BM3D	5	0.05	0.06	2000
SISR x8	0.02	0.001	BM3D	5	0.05	0.06	4000
Deblurring	0.01	0.004	NLM	3	0.02	0.04	30000

Table 1: Parameters used in the experiments.

4.1. Image Denoising

In this experiment, which follows the one in [27], the goal is to remove a white additive Gaussian noise with $\sigma = 25$ from the given images. We evaluate our results on 9 color images[6]. The regularization denoiser we use is Python's scikit-image fast version of Non-Local-Means [3]. The average PSNR (Peak Signal-to-Noise Ratio) of this NLM filter stands on $29.13dB$. When plugged into RED, the performance improves to $29.3dB$. Turning to DIP and its boosted version, DIP's best result is obtained using both averaging strategies (sliding window and average over two runs) getting to $30.53dB$, whereas DeepRED obtains $31.24dB$ – a $0.71dB$ improvement.

Comparing our results to the ones in [14] poses some difficulties, since their performance is given in SNR and not PSNR. Also, we suspect that DIP is poorly functioning in their tests due to the excessive number of iterations used. Disregarding these reservations, we may state that [14] reports of an $0.24dB$ improvement over DIP in image denoising with $\sigma = 25$, whereas our gain stands on $0.71dB$. We should mention [4] – another recent improvement over DIP that relies on stochastic gradient Langevin. They report an average of $30.81dB$, a $0.43dB$ behind our result. This again shows the effectiveness and need of RED.

We use this experiment to briefly discuss run-time of the involved algorithms. Both DIP and DeepRED are quite demanding optimization processes. When used with the same number of iterations (1800), DeepRED is clearly slower due to the additional denoising computations. In this case, the average run-time[7] of DIP on the 9 test images is 6.6 minutes per image, whereas DeepRED requires 9.5 minutes.

4.2. Single Image Super-Resolution (SISR)

This experiment follows [27] as well. Given a low-resolution image, the goal is to recover it's scaled-up version. We test scaling factors of 4 and 8 and compare our results to both DIP [27] and RED [21] on two datasets. These results are summarized in Tables 2 and 3. As can be seen, RED+DIP is consistently better than both DIP or RED alone. Figure 2 presents two visual results taken from these experiments to illustrate the recovery obtained.

We use this experiment to have a closer look at the numerical behavior of the proposed algorithm. For the image head from Set5, we present in Figure 3 the loss of DeepRED as given in Equation (4) as a function of the iteration number. As can be seen, there is a consistent descent. However, notice in the zoomed-in version of this graph the small fluctuations around this general descent behavior, which are due to the additional noise injected in each

[6] http://www.cs.tut.fi/~foi/GCF-BM3D/.

[7] All the reported simulations are run on Intel (R) Xeon (R) CPU E5-2620 v4 @ 2.10Ghz with a GeForce RTX 2080 Ti GPU.

[8] We use the 12 color images from this data-set.

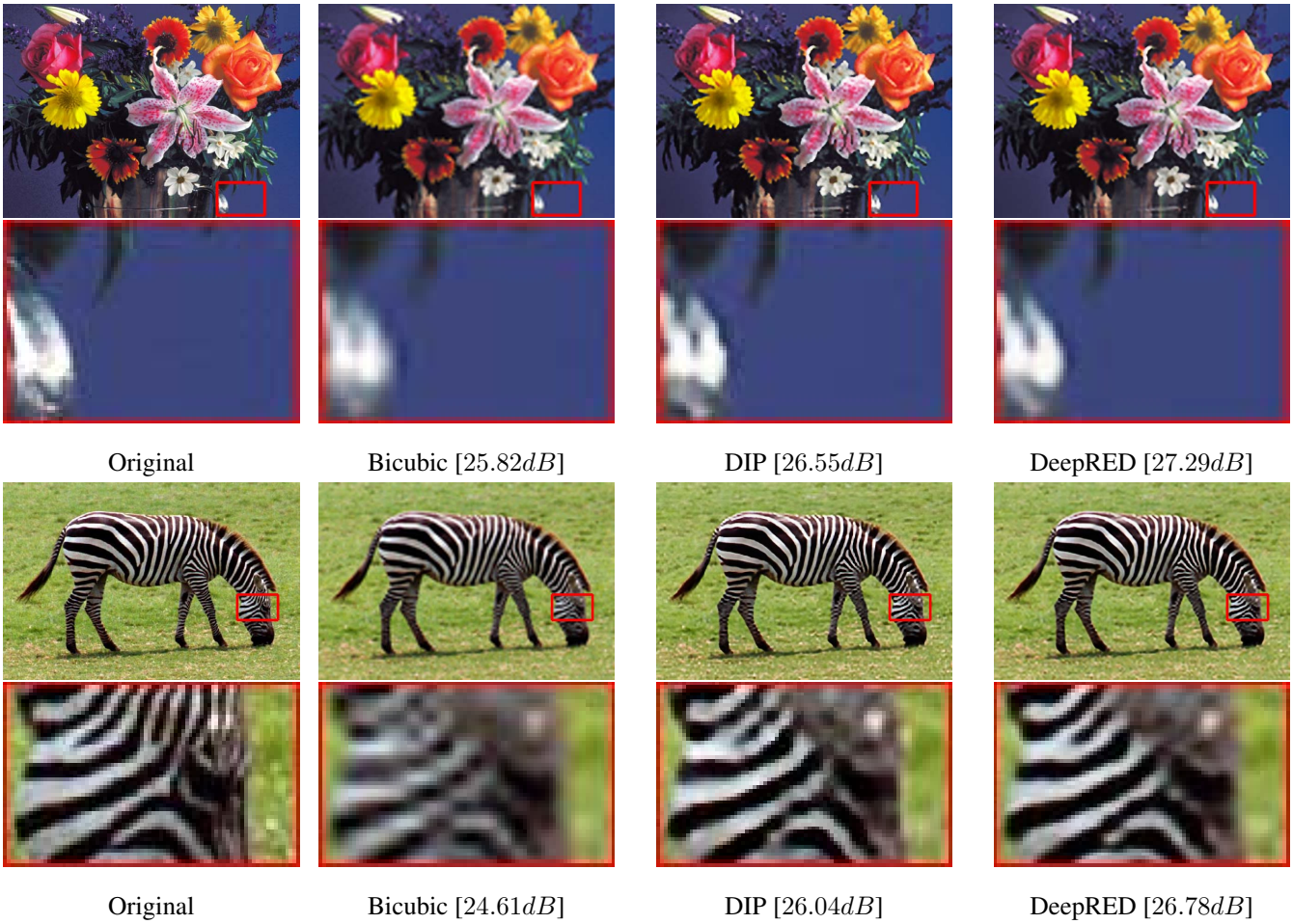


Figure 2: Super resolution results. Top: Flowers (Set14) with scale-factor 4. Bottom: Zebra (Set14) with scale-factor 4.

Set5 Super-Resolution Results (4:1)						
Algorithm	baby	bird	btrfly	head	woman	average
DeepRED	33.08	32.62	**26.33**	32.46	**29.11**	**30.72**
RED [FP-BM3D]	**33.38**	**32.66**	24.03	**32.62**	28.46	30.23
DIP [Our Run]	31.65	31.90	26.01	31.53	28.65	29.95
Set5 Super-Resolution Results (8:1)						
DeepRED	**28.93**	**27.05**	20.04	**30.06**	**24.09**	**26.04**
RED [FP-BM3D]	28.44	26.74	18.96	30.00	23.68	25.56
DIP [Our Run]	28.36	27.01	**20.10**	29.85	23.89	25.84

Table 2: Super-resolution results for Set5.

iteration. The same figure also shows the ADMM equality constraint gap (again, see Equation (4)). Clearly, this gap is narrowing, getting very close to the satisfaction of the constraint $\mathbf{x} = T_\Theta(\mathbf{z})$. The last graph shows the PSNR of the output image over the iterations. RED's regularization tends to robustify the overall recovery algorithm against overfitting, which stands in contrast to the behavior of DIP alone. Similar qualitative graphs are obtained for various other images and applications, showing the same tendencies, and thus are omitted.

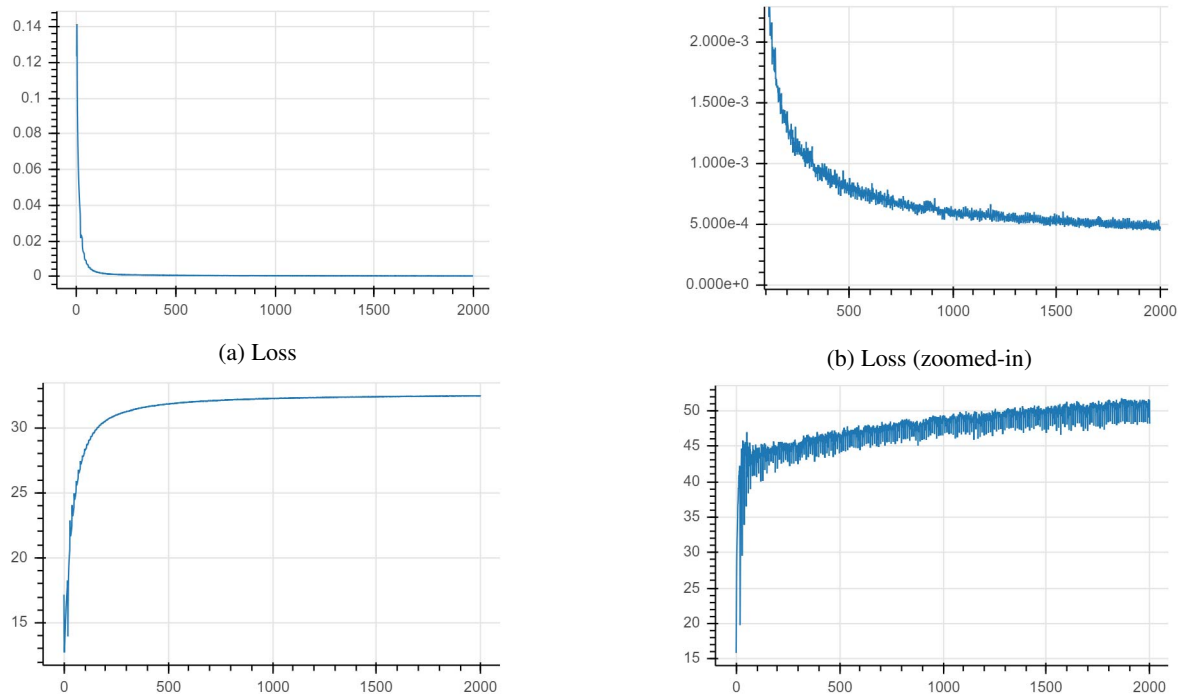
4.3. Image Deblurring

The next experiments follow similar ones in [21] and [31], in which we are given a blurred and noisy image with a known degradation operator $\mathbf{H}$, and the goal is to restore the original image. We consider two cases: (i) A 9×9 uniform blur, and (ii) A 25×25 Gaussian blur of width $\sigma = 1.6$. In both cases, the blurry image is further contaminated by white additive Gaussian noise with $\sigma_n = \sqrt{2}$. We present two comparisons, one using color images (Table 4) and the other with gray-scale ones (Table 5). In the first, 4 color images[9] are used, and DeepRED is compared with with DIP [27], RED [21] and NCSR Deblur [7]. In the second experiment 5 gray-scale images from Set5 are tested, and the comparison is with MSWNN [31], IRCNN [33], RED [21], NCSR [7], IDD-BM3D [6], and EPLL [36]. Figures 4 and 5 present two sets of inputs and results from the color experi-

[9] http://www4.comp.polyu.edu.hk/~cslzhang/NCSR.htm

Set14[8] Super-Resolution Results (4:1)													
Algorithm	baboon	barbara	coastgrd	comic	face	flowers	foreman	lenna	monarch	pepper	ppt3	zebra	average
DeepRED	22.51	25.76	**26.00**	**22.74**	32.37	**27.29**	**29.70**	**31.62**	**30.76**	**31.10**	**24.97**	**26.78**	**27.63**
RED [FP-BM3D]	**22.55**	25.76	25.88	22.57	**32.60**	26.96	29.38	31.56	29.33	31.05	24.50	26.17	27.36
DIP [Our Run]	22.21	25.53	25.82	22.46	31.48	26.55	29.38	30.86	30.27	30.52	24.75	26.04	27.16
Set14 Super-Resolution Results (8:1)													
DeepRED	**21.33**	**24.02**	**23.98**	**20.05**	**29.95**	**23.51**	**25.38**	**28.12**	**25.34**	27.91	**20.69**	**21.03**	**24.28**
RED [FP-BM3D]	21.29	23.94	23.51	19.84	29.90	23.19	24.62	27.69	24.39	27.45	20.23	20.61	23.89
DIP [Our Run]	21.18	24.01	23.74	19.95	29.65	23.32	25.00	27.92	24.85	**27.99**	20.59	20.98	24.10

Table 3: Super-resolution results for Set14.



(a) Loss

(b) Loss (zoomed-in)

(c) PSNR

(d) ADMM equality constraint

Figure 3: The numerical behavior obtained in the SISR test on head (Set5): (a) and (b) show the loss as a function of the iteration; (c) presents the output PSNR; and (d) shows the ADMM constraint gap.

ment, showing clearly the benefit of the RED regularization effect. Looking at Tables 4 and 5, DeepRED performs very well, outperforming all the other alternative methods.

5. Conclusions

DIP is a deep-learning-based unsupervised restoration algorithm of great appeal. This work offers a way to further boost its performance. Our solution relies on RED - the concept of regularizing inverse problems using an existing denoising algorithm. As demonstrated in this paper, DeepRED is a very effective machine for handling various inverse problems.

Further work is required in order to better understand and improve this scheme: (i) Both DIP and DeepRED

[19.07dB] [18.28dB]

Figure 4: The blurred images Butterfly and Leaves.

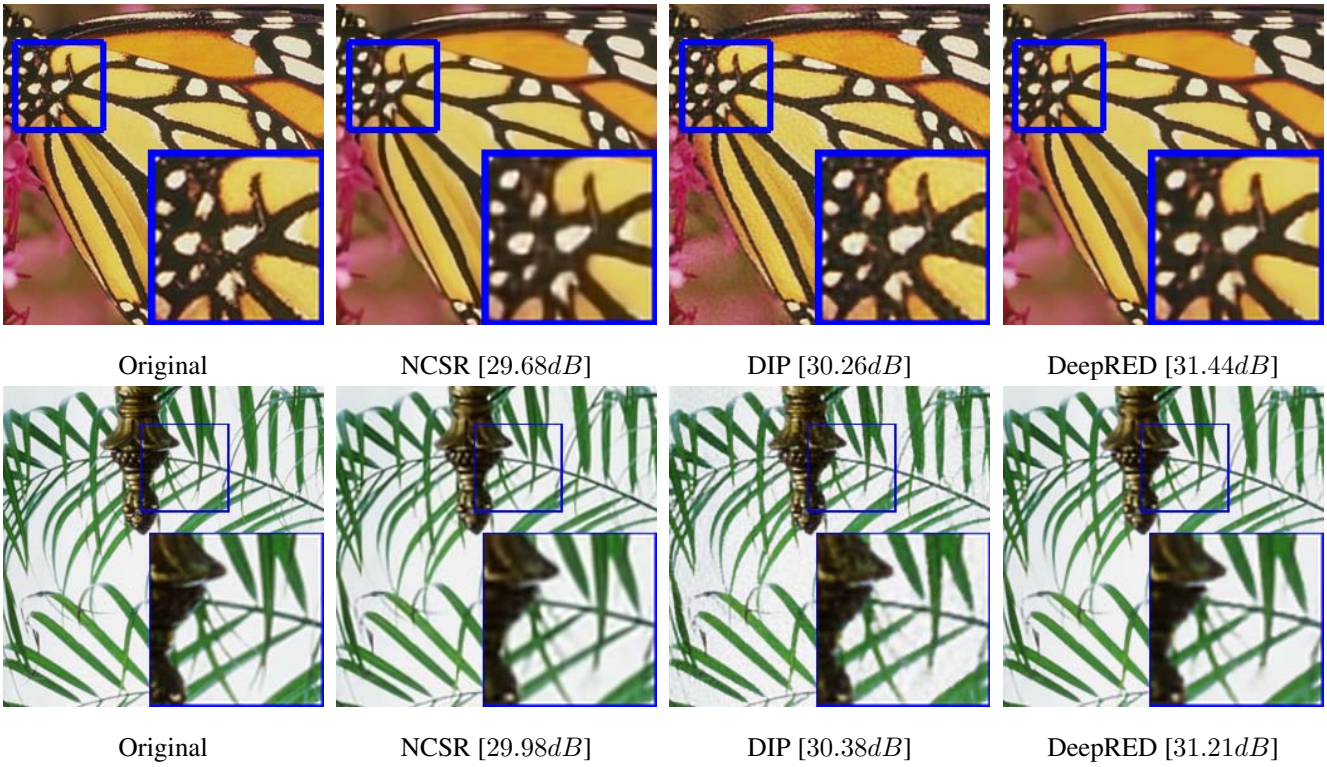


Figure 5: Uniform Deblurring Results: Top – Butterfly, Bottom – Leaves.

Uniform Deblurring Results					
Algorithm	Butterfly	Leaves	Parrots	Starfish	Average
DeepRED	**31.44**	**31.21**	**32.03**	**31.06**	**31.43**
DIP	30.26	30.38	31.00	30.42	30.51
RED FP-TNRD	30.41	30.13	31.83	30.57	30.74
NCSR Deblur	29.68	29.98	31.95	30.28	30.47
Blurred	19.07	18.28	23.87	22.56	20.94
Gaussian Deblurring Results					
Algorithm	Butterfly	Leaves	Parrots	Starfish	Average
DeepRED	**32.19**	**32.27**	32.84	**32.74**	**32.51**
DIP	31.21	31.51	31.91	31.83	31.62
RED FP-TNRD	31.66	31.93	33.33	32.49	32.35
NCSR Deblur	30.84	31.57	**33.39**	32.27	32.02
Blurred	22.81	22.12	26.96	25.83	24.43

Table 4: Color image deblurring results.

should be sped-up in order to make them more practical and appealing. This may be within reach with alternative optimization strategies; (ii) Incorporating better denoisers within the RED scheme (perhaps deep-learning based ones) may lead to further boost in performance; (iii) A more thorough study of the regularization effect that DIP introduces may help in devising a complementary explicit regularization to add via RED, thereby getting a stronger effect and better performance; and (iv) The DIP approach (with or without RED) has an important advantage over supervised regression methods: Whereas the latter aims for a Minimum-Mean-Squared-Error estimation, DIP(+RED) is a Maximum-A'posteriori Probability estimate by definition, a fact that implies a better expected perceptual quality at the cost of a reduced PSNR. A more in-depth study of this matter is central to the understanding of both these restoration strategies.

Uniform Deblurring Results						
Algorithm	Baby	Bird	Butterfly	Head	Woman	Average
DeepRED	33.11	**34.28**	**29.94**	**31.87**	31.14	**32.07**
MSWNN	**33.14**	34.14	28.83	31.81	**31.23**	31.83
IRCNN	32.85	33.90	28.93	31.74	31.08	31.70
RED+TNRD	32.91	33.70	28.60	31.74	30.49	31.49
NCSR	32.81	33.32	27.90	31.55	30.68	31.25
IDD-BM3D	32.98	33.56	27.77	31.65	30.49	31.29
EPLL	32.76	32.49	26.03	31.37	29.05	30.34
Blurred	26.35	24.67	17.75	26.20	22.15	23.42
Gaussian Deblurring Results						
Algorithm	Baby	Bird	Butterfly	Head	Woman	Average
DeepRED	**35.30**	**37.09**	**30.59**	**33.22**	**32.84**	**33.81**
MSWNN	35.21	36.56	30.20	33.01	32.71	33.54
IRCNN	34.83	36.64	29.96	32.68	32.36	33.30
RED+TNRD	34.73	35.88	29.63	32.76	32.13	33.03
NCSR	34.47	35.44	28.77	32.64	31.94	32.65
IDD-BM3D	35.01	36.75	29.28	32.94	32.40	33.27
EPLL	35.06	36.20	28.46	32.88	31.85	32.89
Blurred	30.19	28.87	21.49	29.00	25.91	27.09

Table 5: Gray-scale image deblurring results (Set5).

References

[1] M. V. Afonso, J. M. Bioucas-Dias, and M. A. Figueiredo. Fast image recovery using variable splitting and constrained optimization. *IEEE Transactions on Image Processing*, 19(9):2345–2356, 2010. 3, 4

[2] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011. 2, 3, 4

[3] A. Buades, B. Coll, and J.-M. Morel. A non-local algorithm for image denoising. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 60–65. IEEE, 2005. 2, 3, 5

[4] Z. Cheng, M. Gadelha, S. Maji, and D. Sheldon. A bayesian perspective on the deep image prior. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5443–5451, 2019. 5

[5] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16(8), 2007. 2, 3

[6] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image restoration by sparse 3d transform-domain collaborative filtering. In *Image Processing: Algorithms and Systems VI*, volume 6812, page 681207. International Society for Optics and Photonics, 2008. 6

[7] W. Dong, L. Zhang, G. Shi, and X. Li. Nonlocally centralized sparse representation for image restoration. *IEEE Transactions on Image Processing*, 22(4):1620–1630, 2013. 5, 6

[8] R. Giryes and M. Elad. Sparsity-based poisson denoising with dictionary learning. *IEEE Transactions on Image Processing*, 23(12):5057–5069, 2014. 2

[9] R. Heckel and P. Hand. Deep decoder: Concise image representations from untrained non-convolutional networks. *International conference on learning representations (ICLR)*, 2019. 3

[10] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 3

[11] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017. 3

[12] S. Lefkimmiatis. Universal denoising networks: A novel cnn architecture for image denoising. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3204–3213, 2018. 1, 2

[13] D. Liu, B. Wen, Y. Fan, C. C. Loy, and T. S. Huang. Non-local recurrent network for image restoration. In *Advances in Neural Information Processing Systems*, pages 1680–1689, 2018. 1

[14] J. Liu, Y. Sun, X. Xu, and U. S. Kamilov. Image restoration using total variation regularized deep image prior. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 7715–7719. IEEE, 2019. 2, 5

[15] C. A. Metzler, A. Mousavi, R. Heckel, and R. G. Baraniuk. Unsupervised learning with stein's unbiased risk estimator. *arXiv preprint arXiv:1805.10531*, 2018. 2

[16] S. Nah, T. Hyun Kim, and K. Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3883–3891, 2017. 1

[17] V. Papyan, Y. Romano, J. Sulam, and M. Elad. Convolutional dictionary learning via local processing. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5306–5314, 2017. 3

[18] S.-J. Park, H. Son, S. Cho, K.-S. Hong, and S. Lee. Srfeat: Single image super-resolution with feature discrimination. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 439–455, 2018. 1

[19] T. Plötz and S. Roth. Neural nearest neighbors networks. In *Advances in Neural Information Processing Systems*, pages 1095–1106, 2018. 1

[20] T. Remez, O. Litany, R. Giryes, and A. M. Bronstein. Deep class-aware image denoising. In *International conference on sampling theory and applications (SampTA)*, pages 138–142. IEEE, 2017. 1, 2

[21] Y. Romano, M. Elad, and P. Milanfar. The little engine that could: Regularization by denoising (red). *SIAM Journal on Imaging Sciences*, 10:1804–1844, 2017. 2, 3, 4, 5, 6

[22] Y. Romano, M. Protter, and M. Elad. Single image interpolation via adaptive nonlocal sparsity-based modeling. *IEEE Transactions on Image Processing*, 23(7):3085–3098, 2014. 3

[23] A. Rond, R. Giryes, and M. Elad. Poisson inverse problems by the plug-and-play scheme. *Journal of Visual Communication and Image Representation*, 41:96–108, 2016. 2

[24] L. I. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: nonlinear phenomena*, 60(1-4):259–268, 1992. 2, 3

[25] J. Salmon, Z. Harmany, C. Deledalle, and R. Willett. Poisson noise reduction with non-local pca. *Journal of Mathematical Imaging and Vision*, 48:279–294, 2014. 2

[26] X. Tao, H. Gao, X. Shen, J. Wang, and J. Jia. Scale-recurrent network for deep image deblurring. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8174–8182, 2018. 1

[27] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Deep image prior. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1, 2, 3, 4, 5, 6

[28] D. Ulyanov, A. Vedaldi, and V. Lempitsky. Deep image prior. *Submitted to IJCV*, 2018. 1

[29] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg. Plug-and-play priors for model based reconstruction. In *IEEE Global Conference on Signal and Information Processing*, pages 945–948. IEEE, 2013. 3

[30] Y. Wang, F. Perazzi, B. McWilliams, A. Sorkine-Hornung, O. Sorkine-Hornung, and C. Schroers. A fully progressive approach to single-image super-resolution. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 864–873, 2018. 1

[31] N. Yair and T. Michaeli. Multi-scale weighted nuclear norm image restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3165–3174, 2018. 2, 5, 6

[32] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017. 1, 2

[33] K. Zhang, W. Zuo, S. Gu, and L. Zhang. Learning deep cnn denoiser prior for image restoration. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3929–3938, 2017. 1, 6

[34] K. Zhang, W. Zuo, and L. Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing*, 27(9):4608–4622, 2018. 1, 2

[35] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 286–301, 2018. 1

[36] D. Zoran and Y. Weiss. From learning models of natural image patches to whole image restoration. In *2011 International Conference on Computer Vision*, pages 479–486. IEEE, 2011. 6