

Article

# Image Restoration Based on End-to-End Unrolled Network

Xiaoping Tao<sup>1,2</sup>, Hao Zhou<sup>3</sup> and Yueting Chen<sup>3,\*</sup>

<sup>1</sup> Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China; taoxiaoping@ciomp.ac.cn

<sup>2</sup> Key Laboratory of Optical System Advanced Manufacturing Technology, Chinese Academy of Sciences, Changchun 130033, China

<sup>3</sup> State Key Laboratory of Modern Optical Instrumentation, Zhejiang University, Hangzhou 310027, China; zhopt@zju.edu.cn

\* Correspondence: chenyt@zju.edu.cn

**Abstract:** Recent studies on image restoration (IR) methods under unrolled optimization frameworks have shown that deep convolutional neural networks (DCNNs) can be implicitly used as priors to solve inverse problems. Due to the ill-conditioned nature of the inverse problem, the selection of prior knowledge is crucial for the process of IR. However, the existing methods use a fixed DCNN in each iteration, and so they cannot fully adapt to the image characteristics at each iteration stage. In this paper, we combine deep learning with traditional optimization and propose an end-to-end unrolled network based on deep priors. The entire network contains several iterations, and each iteration is composed of analytic solution updates and a small multiscale deep denoiser network. In particular, we use different denoiser networks at different stages to improve adaptability. Compared with a fixed DCNN, it greatly reduces the number of computations when the total parameters are equal and the number of iterations is the same, but the gains from a practical runtime are not as significant as indicated in the FLOP count. The experimental results of our method of three IR tasks, including denoising, deblurring, and lensless imaging, demonstrate that our proposed method achieves state-of-the-art performances in terms of both visual effects and quantitative evaluations.

**Keywords:** image restoration; deep convolutional neural networks; analytic solution; unrolled optimization



**Citation:** Tao, X.; Zhou, H.; Chen, Y. Image Restoration Based on End-to-End Unrolled Network. *Photonics* **2021**, *8*, 376. <https://doi.org/10.3390/photonics8090376>

Received: 22 June 2021

Accepted: 31 August 2021

Published: 8 September 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

Image restoration (IR) is a classical topic in the field of low-level image processing. Digital images are always degraded during the acquisition process, with issues such as electronic noise caused by the thermal vibration of atoms and blur caused by camera shake [1,2]. Therefore, image restoration is of great significance and is widely used in a variety of applications, e.g., smartphone imaging, medical imaging, and remote sensing. The purpose of image restoration is to recover an unknown latent image from a corrupted observation. In general, IR is an ill-posed inverse problem. The mathematical degradation model can be written as  $y = Ax + n$ , where  $y$  and  $x$  are the degraded measurement and the clean image, respectively.  $n$  denotes the additive noise, which is generally assumed to be additive white Gaussian noise (AWGN), and  $A$  denotes the system degradation matrix. Although IR problems have been extensively studied, they are still full of challenges due to the large amount of natural image contents [3] and the diversity of degradations.

Over the past few decades, many methods have been proposed to tackle IR problems, including denoising [4–10], deblurring [11–15], image super-resolution [16–20], and lensless imaging [21–24]. Recently, the rapid development of deep learning technology has injected new vitality into IR research. Many works based on deep convolutional neural networks (DCNNs) have achieved excellent results [10,23,25]. From the point of view of linear algebra, the reason leads to ill condition of IR is that the null space of  $A$  is nonzero. Because of this, the prior knowledge of the image is crucial, and we need to select a good estimation

of the latent image from the solution space according to this prior knowledge. A common approach is to establish a cost function by maximizing the posterior probability  $P(\mathbf{x} | \mathbf{y})$ :

$$\hat{\mathbf{x}} = \arg \max_x \log P(\mathbf{y} | \mathbf{x}) + \log P(\mathbf{x}) \quad (1)$$

where  $\log P(\mathbf{x})$  represents the prior information of the latent image, and  $\log P(\mathbf{y} | \mathbf{x})$  denotes the log-likelihood of the measurement.  $\log P(\mathbf{y} | \mathbf{x})$  is derived from statistical model of noise, such as  $\ell_2$  - norm to Gaussian noise and  $\downarrow_1$  - norm to Laplacian noise. Under the AWGN model, the cost function can be reformulated as:

$$\hat{\mathbf{x}} = \arg \min_x \frac{1}{2} \|\mathbf{y} - \mathbf{Ax}\|_2^2 + \lambda \cdot \phi(\mathbf{x}) \quad (2)$$

where  $\phi(\mathbf{x}) = -\log P(\mathbf{x})$  is a regularization term. There are two paths for obtaining the solution of Equation (2). One involves model-based methods [26–28], and the other involves learning-based methods [29–32]. The former methods reduce the value of the cost function gradually through the optimization principles, whereas the latter methods mainly rely on DCNNs and datasets. These two types of methods are described in more detail in the following four paragraphs.

Among model-based methods, a large number of models based on various priors have been proposed, including the frequently used total variation (TV) prior [33]; sparse representation prior [34] and dictionary learning [7,35,36]; nonlocal means prior [37] and nonlocal self-similarity (NLSS) [6,38]; the low rank approximation prior [39,40]; and the Markov random field (MRF) [41,42]. The characteristics of various priors are as follows.

The well-known TV prior works well with images with simple textures, but it can lead to distinct blurring in complex areas with rich details. The sparse prior model can represent local image patches as a few atoms in some domains, such as the DCT basis and the discrete wavelet transform basis [43]. Compared with the analytical dictionaries, learned dictionaries have a stronger adaptive ability to represent image patches, and they can deal with various tasks more flexibly [3]. In the past two decades, sparse models have made outstanding contributions to IR, and a large number of IR algorithms based on them have been proposed [4,6,7,16,26,34]. Using the redundant information in an image, the nonlocal means can eliminate Gaussian noise well [44]. A more efficient and robust solution is to apply the NLSS to the sparse prior model [26]. Another powerful prior is the low-rank approximation because a matrix with many nonlocal similar patches is essentially of low rank [39]. The soft threshold function can be used to solve this problem easily and quickly [45]. WNNM [40] improves the flexibility of nuclear norm and achieves good results in terms of both visual effects and quantitative evaluations. Combined with the MRF model, the Bayesian optimization framework is applied to low-level vision [41,46,47]. Although the MRF can learn a generic prior that can represent the statistics of natural scenes, its complexity is high. It is also difficult to carry out a physical interpretation. In summary, model-based methods can deal with all kinds of visual problems flexibly, but they usually incur high time and computational costs, and their effects are not as good as those of the popular DCNNs.

With the rapid development of deep learning, DCNNs have blossomed in the field of low-level vision. Many learning-based methods [8–10,29,30,48,49] have been applied to denoising tasks. Burger et al. [29] proposed using a plain multilayer perceptron for denoising. This work showed the great potential of neural networks because a simple perceptron was shown to be able to achieve the effect of a traditional well-known BM3D denoiser [4]. With batch normalization, the DnCNN [30] was established to provide an end-to-end residual learning network for predicting residuals to indirectly eliminate noise. CBD-Net [8] separated noise estimation from nonblind denoising and used two sub-networks to complete these two functions separately. RIDNet [9] exploited channel dependencies by using feature attention and built a blind real image denoising network under a modular architecture. SADNet [10] introduced the deformable convolution to implement spatial

adaptive denoising, which can achieve reconstruction with a high signal-to-noise ratio while effectively maintaining spatial texture and edges of the image. Zhang et al. [49] provides a novel and efficient RDN that achieves superiority over comparative methods for several image restoration tasks.

In addition to noise removal, DCNNs are also widely used in the fields of super-resolution and lensless imaging. The SRCNN [18] mapped low-resolution patches to high-resolution patches by three layers of convolutions. After that, many deep networks were proposed for super-resolution, such as the ESPCN [19], DRRN [20], VDSR [32], and SRGAN [25]. DCNNs have also performed well in the field of lensless imaging. Nguyen et al. [22] used a DCNN to restore lensless images and protected privacy. Khan et al. [23] first carried out model fitting and then used a DCNN to improve image quality.

Although learning-based DCNNs can quickly complete high-quality image reconstruction on GPUs after training, they usually lose the flexibility inherent in model-based methods. Additionally, the improvement in reconstruction quality is only due to the strong fitting ability of pure DCNN. Therefore, hybrid IR methods under unrolled framework were proposed [50,51]. Section 2.2 describes more details about IR methods under unrolled framework. These kinds of methods combine traditional methods with DCNNs, and the advantages of both can be exploited. When solving the inverse problem, they can incorporate the physical models of systems into the networks. First, this structure can make full use of the prior knowledge of the systems, such as the observation matrix in compressed sensing and the point spread function in deconvolution. In addition, the main function of the network is to learn a prior, rather than the whole inverse operation. Thus, unrolled structure has lower requirements on the network. Additionally, the functions are easier to achieve. Furthermore, they can improve the reconstruction quality compared with pure DCNNs. In spite of their wide applications in low-level vision tasks, there is still room for improvement in terms of optimization and the networks. For example, the existing methods usually use a fixed DCNN in each iteration, and so they cannot fully adapt to the image characteristics at each iteration stage. The gradient descent is slower for the convex problem.

In this paper, we propose a deep denoiser-based unrolled network that combines DCNNs with optimization to exploit the advantages of both. The entire end-to-end network can be unfolded into several analytic solution blocks and the subsequent small deep denoiser networks. All the parameters are learned through training. On the one hand, we solve the convex problem in the form of an analytic solution that is faster than gradient descent which usually requires multiple iterations because the solution is not accurate. On the other hand, each small deep denoiser network adopts a structure with an encoder and a decoder to capture multiscale information from the image. The small deep denoiser networks are also different at different stages so that they can better adapt to the image characteristics at each stage. Compared with a fixed DCNN, it greatly reduces the number of computations when the total parameters are the same and the number of iterations is the same. The experimental results of several IR tasks, including denoising, deblurring, and lensless imaging, demonstrate that our approach is effective and computationally efficient. The visual effects and the objective evaluations indicate that our network achieves excellent performances in high-quality image reconstruction.

The remainder of this paper is organized as follows. Section 2 reviews related works. Section 3 introduces our proposed method. Section 4 shows the numerical results of several IR tasks, and Section 5 concludes this paper.

## 2. Related Work

The unrolled network we proposed is mainly derived from two aspects: deep learning and denoiser-based IR methods under unrolled optimization. In this section, we briefly review these two aspects.

### 2.1. Deep Learning

With the rapid development of computing power, deep learning technology has led to many breakthroughs in the field of vision, including low-level denoising [10,30,48], deblurring [15,31], super-resolution [32,52] and high-level recognition [53], segmentation [54]. The image textures are richer under the generative adversarial networks (GANs) [55]. The perceptual loss takes advantage of high-level abstract features that enhance the details of super-resolution images [56]. Additionally, the training method for networks has been developed, such as batch normalization [57], gradient clipping [58], and Xavier initialization [59].

### 2.2. IR Methods under Unrolled Optimization

By decoupling the deconvolution and denoising, the original complex regularization term can be transferred. Many denoiser based IR methods have been proposed [11,50,60–62] that can integrate the strengths of model-based methods and DCNNs. Under the framework of half-quadratic splitting (HQS), a new auxiliary variable  $z$  is introduced to Equation (2). Therefore, the IR problem can be written in the following form:

$$\varphi_{\mu}(x, z) = \frac{1}{2} \|y - Ax\|_2^2 + \frac{\mu}{2} \|z - x\|_2^2 + \lambda \cdot \phi(z) \quad (3)$$

where  $\mu$  is the penalty parameter. The above equation can be solved alternatively by:

$$\begin{aligned} \hat{x} &= \arg \min_x \|y - Ax\|_2^2 + \mu \|x - \hat{z}\|_2^2 \\ \hat{z} &= \arg \min_z \frac{\mu}{2} \|\hat{x} - z\|_2^2 + \lambda \cdot \phi(z) \end{aligned} \quad (4)$$

As we can see, the former equation is a convex equation while the latter is a proximal operator with special regularization parameters. In practice, we usually treat it as a denoising problem that can avoid the explicit expression of priors and there are many successful solutions at present. The decoupling can also be achieved through the ADMM [63]; the principle is similar and will not be repeated here. IR methods under unrolled optimization frameworks can be divided into two categories: deep unfolding networks and plug-and-play. The former type is an overall end-to-end network, while the latter is not.

Plug-and-play methods can be flexibly applied to various tasks [64,65] by using one well-trained denoiser. In [11], the well-known BM3D denoiser was used for deblurring based on the generalized Nash equilibrium. The CBM3D denoiser was used for single image super-resolution (SISR) [17]. In [66], Brifman et al. realized SISR by using the NCSR denoiser [26] which combines the traditional sparse prior with the NLSS. Additionally, the results were better than those of original NCSR. The TV prior and BM3D prior were used for Fourier ptychographic microscopy [67]. In [68], the denoising-based IR method with the ADMM was used for electron microscope imaging. Additionally, the state-of-the-art DCNN denoiser prior was used in IR tasks [69]. Zhang et al. [61] trained 25 CNN denoisers at different noise levels. There are some theoretical analyses on this topic. Sreehari et al. [68] analyzed the convergence of the plug-and-play approach when the denoiser is a symmetric smoothing filter. Chan's algorithm with a bounded denoiser [70] was proven to be convergent. Ryu's work theoretically established the convergence of the PnP-ADMM algorithm when the denoiser has a certain Lipschitz condition [71]. Although the plug-and-play technique has achieved start-of-the-art results, it usually requires multiple iterations. The proposed SISR solver in [66] iterates 35 times, and the IRCNN [61] takes 30 iterations to deblur.

In response to this problem, the end-to-end deep unfolding network consisting of only a few iterations was proposed for IR [51,62]. Zhang et al. [72] proposed a deep network for compressed sensing reconstruction. Dong et al. [51] proposed an end-to-end approach named as the DPDNN. In [51], the whole iterative process was carried out six times, where the same denoising network was called each time. Despite the small number of iterations,

its effect was still outstanding. Jeon et al. [73] achieved high-spectral reconstruction through a deep unfolding network.

### 3. Proposed Algorithm for Image Restoration

In this section, we introduce the principle and process of our method in detail. The general form of the analytic solution is given. Additionally, its application and variation to the three IR tasks are discussed in detail.

#### 3.1. Our End-to-End Unrolled Network

Generally, the goal of the IR task is to obtain an output with a lower cost. In our method, the cost function is described in Equation (3). The HQS method converts our cost function into two sub-problems, as described in Equation (4). In this way, the two sub-problems are easy to solve. The first convex equation in Equation (4) can be solved by gradient descent or in the form of an analytic solution. The gradient descent algorithm is a simple and generalized method. The first-order method is often used in various inverse problems because it can obtain a good result after many iterations, and the second-order method is too computationally expensive. As we stated before, the gradient descent algorithm usually requires multiple iterations because the solution is not accurate. This leads to high time costs in traditional methods. The number of iterations is limited by time and space costs in DCNN-based IR methods. In this paper, we solve the first equation in (4) in the form of an analytic solution. By deriving  $\mathbf{x}$  and setting the derivative equal to zero, the following formula can be obtained:

$$\left(A^T A + \mu I\right) \hat{\mathbf{x}} = A^T \mathbf{y} + \mu \hat{\mathbf{z}} \tag{5}$$

It is evident that matrix inversion is a stumbling block on the road to an analytic solution because of its computational complexity. We use the singular value decomposition of  $A$  to reduce the computational complexity of matrix inversion [74] because the cost of the inversion a diagonal matrix is small. Through SVD on degenerate matrix  $A$ , the analytic solution can be written in the following form:

$$\hat{\mathbf{x}} = V_A \left(S_A^T S_A + \mu I\right)^{-1} V_A^T \left(V_A S_A^T U_A^T \mathbf{y} + \mu \hat{\mathbf{z}}\right) \tag{6}$$

where  $A = U_A S_A V_A^T$ . As we can see, the updates of the analytic solution can be calculated quickly and efficiently. The overall framework of our proposed approach is shown in Figure 1a. Since the first equation in Equation (4) is convex, we can obtain the optimal solution for each iteration. This lays the foundation for our end-to-end network consisting of only a few iterations to achieve excellent results. The solution of the latter equation in Equation (4) is a proximal operator, which is as follows:

$$prox_{\tau\phi}(\hat{\mathbf{x}}) = arg \min_z \frac{1}{2} ||\hat{\mathbf{x}} - z||_2^2 + \tau \cdot \phi(z) \tag{7}$$

where  $\tau = \lambda/\mu$ . In this step, the prior information is important. The research on IR methods under unrolled optimization shows that the DCNNs can express image priors implicitly. Combined with the DCNN’s strong fitting ability, we use the deep denoiser network to solve the latter problem in Equation (4). We use different deep prior networks at different stages to better adapt to the image characteristics of each stage. The proposed end-to-end deep analytic network based on deep priors is summarized in Algorithm 1. In our method, the number of iterations is set to six. Our deep denoiser networks at each stage are small. Benefiting from the above settings, our overall model is not too large, which allows it to avoid overfitting.



**Algorithm 1.** DCNNs based end-to-end unrolled network for IR

**Input:**  $\mathbf{A}, \mathbf{y}, \mu_0 > 0, k = 0$

**Initialization:**

(1) Initial estimation :  $\hat{x}_0 = \mathbf{A}^T \mathbf{y}$  or the least squares estimate,  $\hat{z}_0 = \hat{x}_0$

(2) SVD :  $\mathbf{A} = \mathbf{U}_A \cdot \mathbf{S}_A \cdot \mathbf{V}_A^T$

**For**  $iter = 1 : k$  **do**

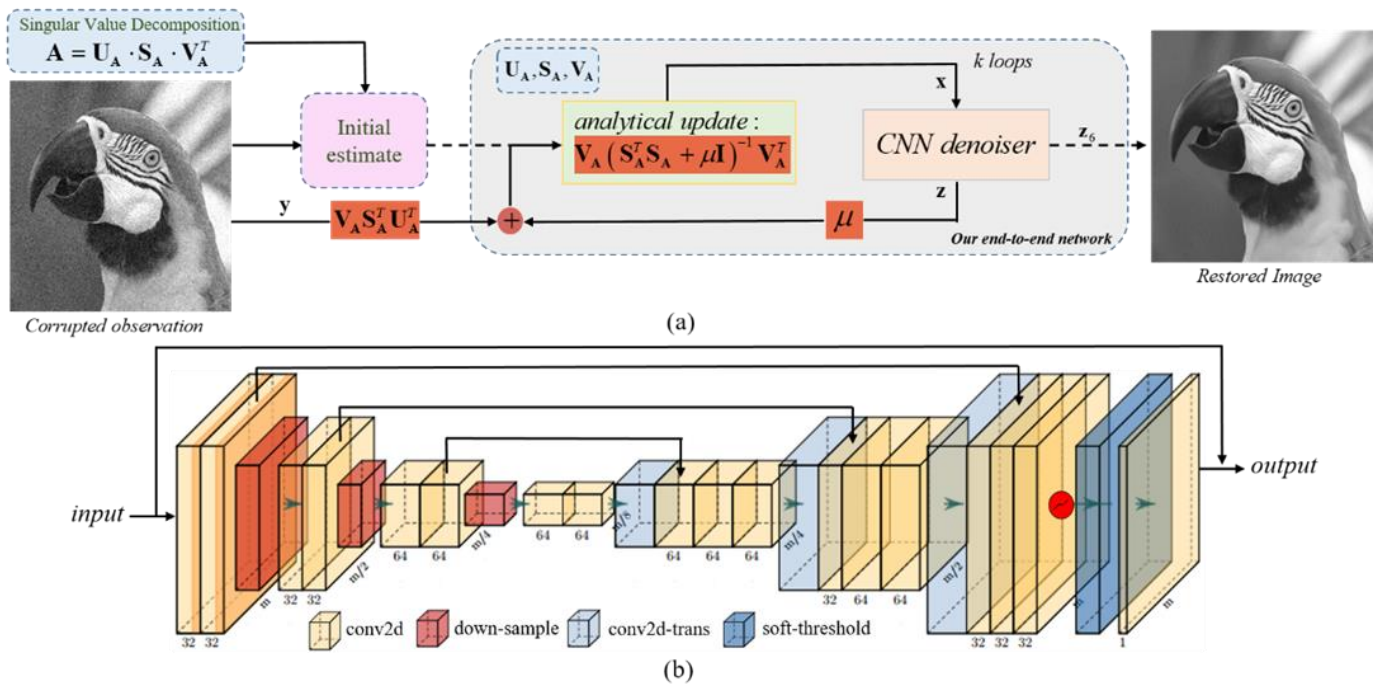
(1) Analytic updates:

$$\hat{x}_k = \mathbf{V}_A (\mathbf{S}_A^T \mathbf{S}_A + \mu_k \mathbf{I})^{-1} (\mathbf{S}_A^T \mathbf{U}_A^T \mathbf{y} + \mu_k \mathbf{V}_A^T \hat{z}_{k-1})$$

(2) Deep prior net :  $\hat{z}_k = D_k(\hat{x}_k)$

**end**

**Output**  $\hat{z}_k$



**Figure 1.** Overview of our proposed method for image restoration. (a) The architecture of the proposed deep priors-based deep analytic network. (b) The structure of the deep prior network used in our method.

**3.2. Structure of the Deep Denoiser Network**

The well-known U-Net [54] was proposed for medical image segmentation originally. Additionally, it has been widely used in visual tasks due to its excellent effects. Inspired by those works, our proposed DCNNs is a residual learning network with a U-Net structure. The architecture of our proposed deep network is illustrated in Figure 1b. The network is a four-scale U-Net with a soft-threshold function. The first half is a multiscale encoder for feature extraction, and the second half is a multiscale decoder for image reconstruction based on these features. In each scale of the encoder, we use two convolutional layers to encode spatial features and a max-pooling layer to increase the receptive field. The number of channels in the two convolutional layers in the first two scales of the encoder is 32. In the third scale, the number of channels is 64. After three feature extractions, there are two 64-channel convolutional layers at the top of the DCNNs. The kernel size of each convolutional layer is  $3 \times 3$ . In each scale of the decoder, there is a trans-convolutional layer, a skip layer, and two convolutional layers. The number of channels in the two convolutional layers in the first scale of the decoder is 32. In other two scales of the decoder, the number of channels is 64. The skip layer combines feature maps of the same size to compensate for the loss of spatial details caused by multiple extraction operations. After the decoder, a soft-threshold function is used to shrink the multichannel image. Then, a

convolutional layer restores the image to the original color space. Finally, we establish a long residual connection between the input and output, because residual learning is easier to optimize [30] and more robust [51]. Different from the original U-Net, we adopt the leaky ReLU [75] as the activation function.

### 3.3. Variation in Three Applications

We have introduced the principles and process of our method and gave the general form of analytic solutions. In this section, we discuss its specific form and variations in three visual problems (denoising, deblurring, and lensless imaging). In the denoising problem, the system degradation matrix  $A$  is the identity matrix. In this case, the analytic solution degenerates into the following form:

$$\hat{\mathbf{x}} = \frac{1}{1 + \mu}(\mathbf{y} + \mu\mathbf{z}) \tag{8}$$

As we can see, the update of the analytic solution is a basic matrix operation, which can be completed quickly and efficiently. For deblurring with a uniform kernel, the former equation in Equation (4) is usually written in a convolutional form:

$$\hat{\mathbf{x}} = \arg \min_x \|\mathbf{y} - \mathbf{psf} * \mathbf{x}\|_2^2 + \mu \|\mathbf{x} - \hat{\mathbf{z}}\|_2^2 \tag{9}$$

where  $*$  denotes a two-dimensional convolution operation. In this situation, the system degradation matrix is a large sparse blurring matrix  $A$ . It is not wise to solve the equation in matrix form. Hence, we obtain an analytic solution in the frequency domain based on energy equality, as shown below:

$$\hat{\mathbf{x}} = \mathcal{F}^{-1} \left\{ \frac{\mu_k \mathcal{F}(\hat{\mathbf{z}}) + \overline{\mathcal{F}(\mathbf{psf})} \mathcal{F}(\mathbf{y})}{\overline{\mathcal{F}(\mathbf{psf})} \mathcal{F}(\mathbf{psf}) + \mu} \right\} \tag{10}$$

where  $\mathcal{F}$  and  $\mathcal{F}^{-1}$  represent the fast Fourier transform (FFT) and the inverse FFT, respectively.  $\overline{M}$  represents the complex conjugate of matrix  $M$ . We use Equation (10) instead of analytic updates in Algorithm 1 for deblurring. The third scenario is a lensless imaging problem named FlatCam [21]. In FlatCam, the system model is:

$$\mathbf{y} = \Phi_L \mathbf{x} \Phi_R^T + \mathbf{n} \tag{11}$$

where  $\Phi_L, \Phi_R$  are system transfer matrixes.  $\mathbf{n}$  denotes noise. Therefore, the former equation in (4) becomes the following:

$$\hat{\mathbf{x}} = \arg \min_x \|\mathbf{y} - \Phi_L \mathbf{x} \Phi_R^T\|_2^2 + \mu \|\mathbf{x} - \hat{\mathbf{z}}\|_2^2 \tag{12}$$

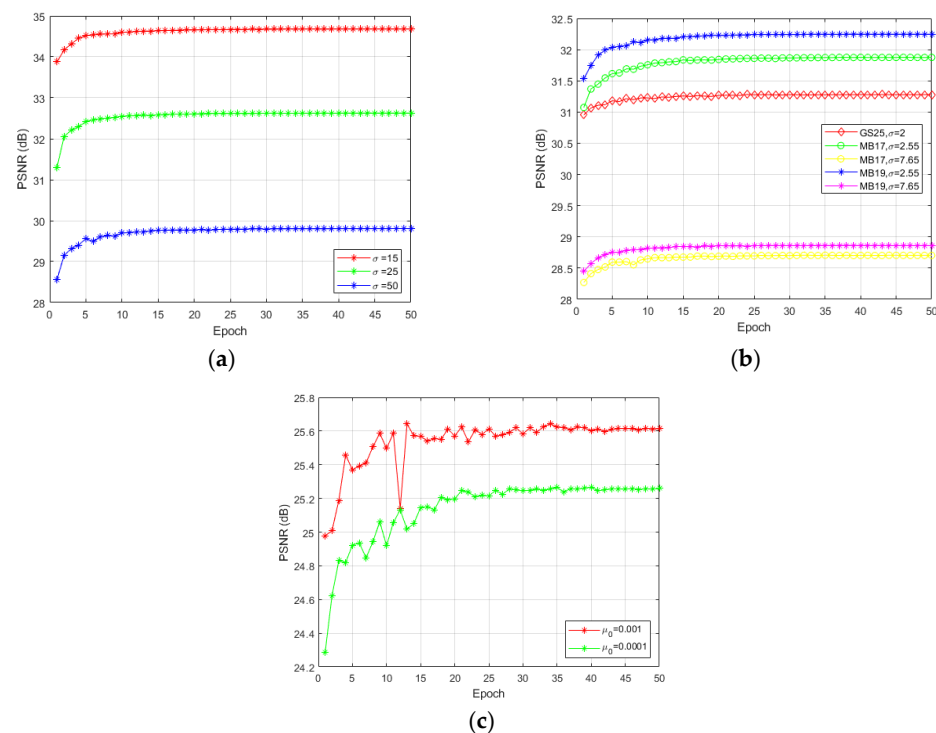
The corresponding analytic solution is as follows:

$$\hat{\mathbf{x}} = \mathbf{V}_L \left[ \frac{\mathbf{U}_L^T \mathbf{y} \mathbf{U}_R \odot (\sigma_L \sigma_R^T) + \mu \mathbf{V}_L^T \hat{\mathbf{z}} \mathbf{V}_R}{\sigma_L^2 \sigma_R^T + \mu \cdot \|\mathbf{ones}\|} \right] \mathbf{V}_R^T \tag{13}$$

where  $[\mathbf{U}_L, \mathbf{S}_L, \mathbf{V}_L^T] = \text{SVD}(\Phi_L)$ ,  $[\mathbf{U}_R, \mathbf{S}_R, \mathbf{V}_R^T] = \text{SVD}(\Phi_R)$ , and the vectors  $\sigma_L, \sigma_R$  are the diagonal entries of  $\mathbf{S}_L, \mathbf{S}_R$ .  $\|\mathbf{ones}\|$  denotes a matrix in which all elements are ones.  $\odot$  is the Hadamard product. SVD is carried out in advance, and then the above formula can be calculated efficiently. In the lensless image restoration experiment, Equation (13) is used instead of the analytic update in Algorithm 1.

As shown in [76], the analytic updates of the lensless model  $\mathbf{y} = \Phi_L \mathbf{x} \Phi_R^T$  also do not introduce singularity and gradient explosion. This lays a theoretical foundation for our network to successfully complete the training process. In the actual training process, we record the PSNR values of the test image for different epochs, as shown in Figure 2. It can

be seen that the losses converge gently to a straight line, which confirms the above analysis from the side.



**Figure 2.** PSNR (dB) during the training epochs of three applications. (a) Three Gaussian denoising experiments. (b) Five image deblurring experiments. (c) Two FlatCam imaging experiments.

#### 4. Experiments

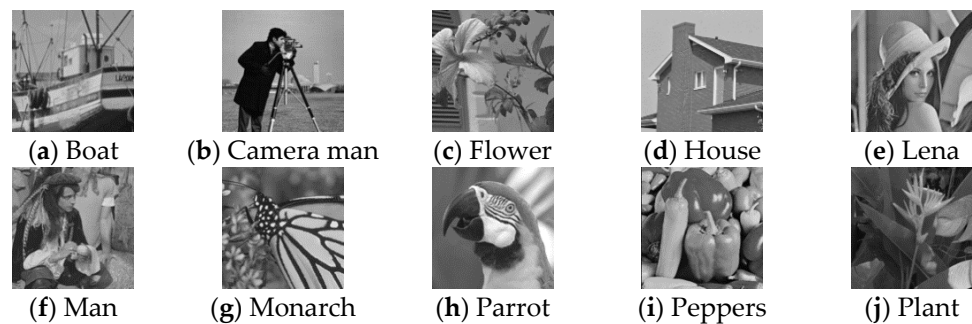
In this section, we perform experiments on three IR tasks: image denoising, image deblurring, and lensless imaging. We fix and train a model for each specific problem. All models are implemented in TensorFlow [77] and trained on a Linux server with an Intel E5-2678 CPU at 2.5 GHz with 64 GB of memory and four graphic cards (NVIDIA GTX1080Ti) with 11 GB of memory. We train our models through the ADAM optimizer [78] by setting  $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$ , and  $\varepsilon = 10^{-8}$ .  $\ell_2$  - loss is used as the loss function in all experiments, which means that  $loss = \|z_6 - GT\|_2^2$ .

We train each model for 50 epochs. In addition, the PSNR and SSIM [79] metrics are used for objective evaluation.

##### 4.1. Ablation Study

Dong et al. [51] undertook some research into the deep unfolding IR method. Additionally, he performed a comparative experiment of deep unfolding net and pure DCNNs. His results show that the performance of deep unfolding net which combining traditional with deep learning is better than pure DCNNs. On this basis, we conducted five groups of deblurring experiments to show the superiority of the analytical solution under the deep unfolding framework. The datasets and training settings in each group used were the same. More details are described in Section 4.3. The commonly used 10 images are used for deblurring tests are shown in Figure 3. The results are summarized in Table 1.





**Figure 3.** The 10 commonly used images for our deblurring testing.

**Table 1.** The average PSNR of 10 images in Figure 3 by three methods.

Kernel	$17 \times 17$ Kernel in [80]	$19 \times 19$ Kernel in [80]	$25 \times 25$ Gaussian
Noise $\sigma$	2.55	7.65	2
DPDNN [51]	31.97	28.65	31.01
DPDNN-AS	32.30	28.89	31.24
Ours	32.42	29.01	31.38

DPDNN is a deep unfolding net that uses gradient descent in the first step and a net in the second step. The DPDNN-AS method replaces the gradient descent in the original DPDNN with the analytic solution and keeps the rest unchanged. From Table 1, the average PSNR is increased over 0.2 dB by using the analytic solution, which demonstrates the advantage of the analytical solution. In addition, the increased PSNR between our network and the DPDNN-AS method shows the power of six small deep prior networks that have stronger adaptabilities at different stages.

We also performed the ablation study results on different iteration numbers of our method. Two groups of deblurring experiments on the Kodak24 dataset with different kernel size were conducted as examples. The results are summarized in Table 2. The comparisons of calculation amount are shown in Table 3. Ours-1, Ours-2, Ours-4, Ours-6 (Ours), and Ours-8 indicate the number of iterations in our method are 1, 2, 4, 6, and 8, respectively. Considering the effectiveness and computation cost, the number of iterations is set to six in our method. Compared with a fixed DCNN (DPDNN), our method greatly reduces the number of FLOPs when the total parameters are the same and the number of iterations is the same.

**Table 2.** The average PSNR and SSIM values of our method with different iteration numbers on the Kodak24 dataset.

Number of Iterations	PSNR	SSIM
$17 \times 17$ motion blur kernel of [80], $\sigma_n = 2.55$		
Ours-1	31.02	0.857
Ours-2	31.48	0.865
Ours-4	31.80	0.872
Ours-6 (Ours)	31.94	0.874
Ours-8	32.00	0.875
$19 \times 19$ motion blur kernel of [80], $\sigma_n = 2.55$		
Ours-1	31.32	0.865
Ours-2	31.81	0.873
Ours-4	32.18	0.880
Ours-6 (Ours)	32.33	0.882
Ours-8	32.41	0.884

**Table 3.** Parameters and FLOPs of DPDNN and our method with different iteration number in the deblurring tasks.

Image Size		256 × 256 × 1				
Method	DPDNN	Ours-1	Ours-2	Ours-4	Ours	Ours-8
Parameters	1249K	393K	787K	1573K	2359K	3146K
FLOPs	794G	29G	59G	118G	177G	236G
Run time (Sec)	0.0712	0.0211	0.0256	0.0453	0.0575	0.0615

#### 4.2. Image Denoising

In image denoising, the analytic updates are as in Equation (8) and  $\hat{x}_0 = y$ . To train our network, we built a training dataset from the DIV2K dataset [81]. First, we cut out 27,594 small patches from DIV2K, each of which was  $256 \times 256$  in size. Then, we added zero-mean Gaussian noise with the variance of  $\sigma_n$  to these small patches. Finally, we saved the values of these patches as integers from 0 to 255. In this way, the original patch and the patch after adding noise formed a training pair. We performed three groups of experiments for image denoising, and  $\sigma_n$  was set as 15, 25, and 50. In these three experiments, the batch size was 32, the initial value of  $\mu_0$  was 0.9, and learning rate was 0.0005. The learning rate was halved after every five epochs.

To illustrate the excellent effect of our network, we compare it with the existing model-based methods—i.e., the BM3D method [4], EPLL method [5], and WNNM [40]—and learning based methods—i.e., the TNRD [60], IRCNN [61], DnCNN-S [30], and FFDNet-cl [48]. The BSD68 dataset and the Kodak24 dataset are used for testing. The values of noisy inputs in the test are also clipped to integers between 0 and 255. The images of the above three test sets are tested in grayscale. Table 4 records the average PSNR (dB) and SSIM values of the compared methods on the BSD68 and the Kodak24 datasets. The highlighted results show that our network is better than the compared methods. Among them, the data of TNRD and IRCNN are derived from a published paper [61], and the data of other methods are obtained according to public codes.

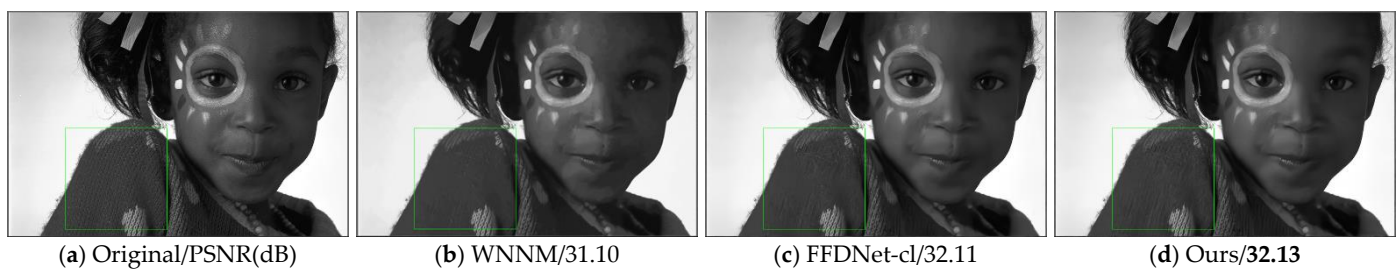
**Table 4.** The average PSNR values of test methods for denoising on the Bsd68 and Kodak24 datasets.

Method	BM3D [4]		WNNM [40]		FFDNet-cl [48]		TNRD [60]		IRCNN [61]		Ours		
BSD68	$\sigma_n$	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
	15	31.02	0.873	31.23	0.876	31.65	0.890	31.42	-	31.63	-	<b>31.70</b>	<b>0.891</b>
	25	28.34	0.797	28.52	0.803	29.21	0.829	28.92	-	29.15	-	<b>29.25</b>	<b>0.831</b>
	50	24.86	0.669	24.81	0.664	26.28	0.725	25.97	-	26.19	-	<b>26.28</b>	<b>0.726</b>
Kodak24	Method	BM3D [4]		WNNM [40]		FFDNet-cl [48]		EPLL [5]		DnCNN-S [30]		Ours	
	$\sigma_n$	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
	15	32.23	0.877	32.46	0.880	32.81	0.892	32.10	0.881	32.72	0.890	<b>32.85</b>	<b>0.893</b>
	25	29.68	0.814	29.89	0.818	30.47	0.838	29.54	0.815	30.13	0.832	<b>30.51</b>	<b>0.840</b>
	50	26.22	0.707	26.23	0.705	<b>27.61</b>	0.748	25.94	0.696	26.40	0.717	27.53	<b>0.750</b>

Figure 4 shows the denoising result of the well-known image Lena. As we can see, our result is more delicate and maintains more thin lines in the complex area than other results. When the noise level is  $\sigma_n = 25$ , our results are also better than the compared results. As shown in the green box in Figure 5, our method works well in the high-frequency region and restores more details and textures than other methods.



**Figure 4.** Image denoising results on Lena with noise level  $\sigma_n = 15$ . (a) The original image. (b) The result of BM3D. (c) The result of WNNM. (d) The result of FFDNet-cl. (e) Our result.



**Figure 5.** Image denoising results on kodim15 in Kodak24 with noise level  $\sigma_n = 25$ . (a) The original image. (b) The result of WNNM. (c) The result of FFDNet-cl. (d) Our result.

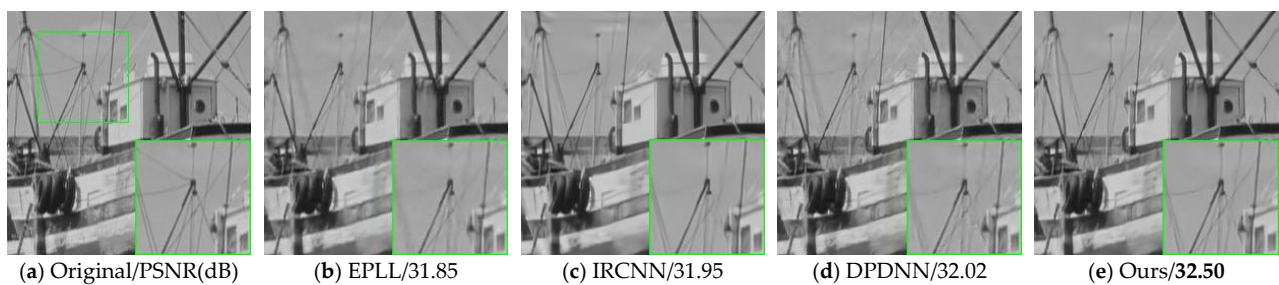
### 4.3. Image Deblurring

In order to verify the deblurring ability of our network, we performed five groups of experiments. In these experiments, we used the clear image convolution blur kernel to obtain the data set. Three blur kernels were selected, including a  $25 \times 25$  Gaussian blur kernel with a standard deviation of 1.6 and two motion blur kernels from [80], one of which was  $17 \times 17$ , and the other was  $19 \times 19$ . For image deblurring, the analytic update in Algorithm 1 is shown in Equation (10), and  $\hat{x}_0 = A^T y$ . To train our deblurring network, we first convoluted the images in the DIV2K dataset with a blur kernel and added additive Gaussian noise with a standard deviation of  $\sigma_n$ . In particular, we used zeros to fill the border of the image during the convolution. The convolution results were saved as integers between 0 and 255. Next, we cut the edge of each blurred image at half the length of the blur kernel and extracted 27,468 patches with size of  $256 \times 256$  pixels from it. We trained five models for five experiments, and the blur settings are shown in Table 4. In these deblurring experiments, the batch size was 32, the initial value of  $\mu_0$  was 0.9, and the learning rate was 0.0005. Similar to the denoising networks, our deblurring networks also reduced the learning rate by half every five epochs. During the test phase, we employed the 10 commonly used images seen in Figure 3 and the Kodak24 dataset as test images. Similarly, the above convolution was used to generate the blurred inputs for the test images. All images were operated at grayscale. With the purpose of demonstrating the excellent performance of our network, we compare it with the classical model-based methods (IDD-BM3D [11], EPLL [5], and NCSR [26]), the denoising-based IR method under framework of plug and play (IRCNN [61]), and the end-to-end deep unfolding network (DPDNN [51]). The PSNR values of the deblurring results of 10 commonly used images (see Figure 3) are shown in Table 5. The results of the IDD-BM3D, EPLL, NCSR, and IRCNN methods are obtained by restoring the test images with open codes. If there is a ringing effect in the test process, we use the egdetaper function of MATLAB to perform an edge-preserving processing on the input.

**Table 5.** The PSNR values of the deblurred 10 images in Figure 3 by the test methods.

Method	Image	Boat	C. Man	Flower	House	Lena256	Man	Monar.	Parrots	Peppers	Plant	Ave.
Gaussian blur with standard deviation 1.6, $\sigma_n = 2$												
IDD-BM3D [11]		29.97	26.65	28.40	32.49	29.58	30.43	28.37	29.62	29.43	32.25	29.72
EPLL [5]		30.55	26.66	28.81	32.83	30.03	30.63	29.37	29.80	30.02	32.88	30.16
NCSR [26]		31.19	27.62	29.28	33.33	30.30	30.93	29.86	30.52	30.24	33.56	30.68
IRCNN [61]		31.20	27.94	29.63	33.53	30.44	30.99	30.58	30.24	30.76	33.86	30.92
DPDNN [51]		31.10	28.08	29.66	33.27	30.71	31.13	30.76	30.81	30.66	33.89	31.01
Ours		<b>31.44</b>	<b>28.53</b>	<b>30.01</b>	<b>33.73</b>	<b>30.95</b>	<b>31.28</b>	<b>31.06</b>	<b>31.13</b>	<b>30.96</b>	<b>34.69</b>	<b>31.38</b>
$17 \times 17$ motion blur kernel of [81], $\sigma_n = 2.55$												
IDD-BM3D [11]		30.24	29.36	28.70	32.71	30.30	30.11	27.39	31.70	28.93	32.34	30.18
EPLL [5]		31.85	29.98	30.03	33.90	31.70	31.20	30.02	32.29	31.03	33.21	31.52
IRCNN [61]		31.95	30.84	30.51	33.49	31.90	31.31	29.20	33.15	29.80	34.09	31.62
DPDNN [51]		32.02	30.45	30.39	33.90	32.35	31.65	31.15	32.86	31.13	33.82	31.97
Ours		<b>32.50</b>	<b>30.90</b>	<b>30.77</b>	<b>34.66</b>	<b>32.72</b>	<b>31.88</b>	<b>31.67</b>	<b>33.29</b>	<b>31.37</b>	<b>34.45</b>	<b>32.42</b>
$17 \times 17$ motion blur kernel of [81], $\sigma_n = 7.65$												
IDD-BM3D [11]		27.22	25.78	25.61	30.20	27.59	27.20	25.25	27.85	26.86	29.20	27.28
EPLL [5]		26.96	24.87	25.07	28.93	27.33	27.24	23.73	26.14	27.04	28.65	26.60
IRCNN [61]		28.56	<b>27.69</b>	26.92	31.40	28.81	28.41	27.25	29.55	27.75	30.52	28.69
DPDNN [51]		28.60	27.28	26.82	31.08	28.85	28.51	27.47	29.45	28.18	30.30	28.65
Ours		<b>29.03</b>	27.63	<b>27.20</b>	<b>31.75</b>	<b>29.17</b>	<b>28.70</b>	<b>27.84</b>	<b>29.70</b>	<b>28.38</b>	<b>30.70</b>	<b>29.01</b>
$19 \times 19$ motion blur kernel of [81], $\sigma_n = 2.55$												
IDD-BM3D [11]		30.29	29.42	29.38	31.82	30.49	30.52	28.93	31.21	28.97	32.72	30.38
EPLL [5]		32.13	30.57	30.47	33.19	32.31	31.58	30.91	32.62	31.41	33.74	31.89
IRCNN [61]		31.59	30.57	30.93	32.00	31.74	31.30	30.52	32.48	29.88	34.19	31.52
DPDNN [51]		32.59	31.06	31.36	33.63	32.94	32.05	32.02	33.31	31.66	34.67	32.53
Ours		<b>33.13</b>	31.42	<b>31.81</b>	34.27	<b>33.21</b>	<b>32.25</b>	32.43	33.61	<b>31.85</b>	<b>35.15</b>	<b>32.91</b>
$19 \times 19$ motion blur kernel of [81], $\sigma_n = 7.65$												
IDD-BM3D [11]		27.54	26.32	25.62	30.17	27.89	27.38	25.58	28.00	27.29	29.42	27.52
EPLL [5]		27.01	25.61	24.85	29.04	27.51	27.54	24.48	26.63	27.43	28.99	26.91
IRCNN [61]		28.80	27.77	27.43	30.92	29.25	28.52	28.04	29.68	28.47	31.12	29.00
DPDNN [51]		28.84	27.61	27.23	30.78	29.27	28.72	28.20	29.76	28.64	31.02	29.01
Ours		<b>29.17</b>	<b>27.77</b>	<b>27.53</b>	31.24	29.39	<b>28.86</b>	<b>28.43</b>	<b>29.91</b>	<b>28.77</b>	<b>31.44</b>	<b>29.25</b>

We retrain the DPDNN method with our training set according to open codes. It can be seen from Table 5 that our method is obviously superior to the compared methods. Our results are 0.36 dB higher than those of the DPDNN, on average. We selected three groups of images to visually show the effect of the deblurring, as shown in Figures 6–8. Among them, Figure 6 shows that our result recovers the most object information in the region close to the background. Figure 7 shows the effect of processing high-frequency areas, and the enlarged green boxes show that we have restored more hairs and details.

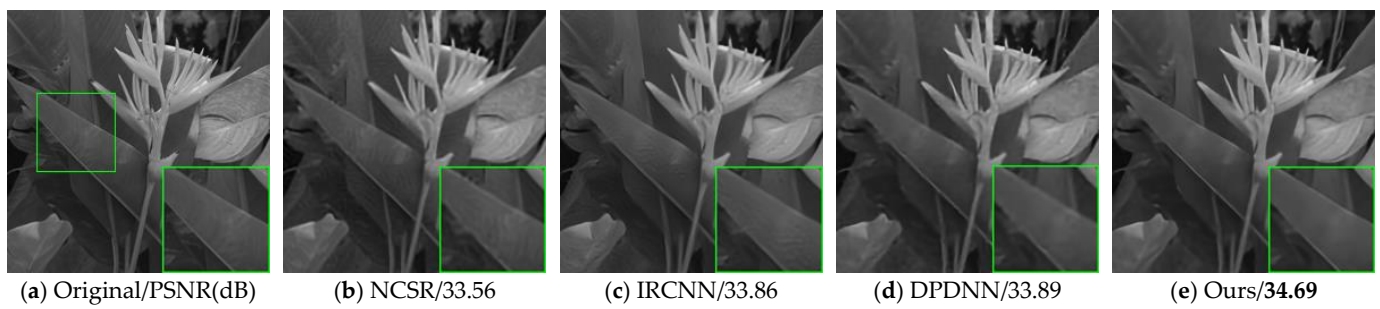


**Figure 6.** Image deblurring results on Boat in Figure 3 with  $17 \times 17$  motion blur kernel and  $\sigma_n = 2.55$ . (a) The original image. (b) The result of EPLL. (c) The result of IRCNN. (d) The result of DPDNN. (e) Our result.



**Figure 7.** Image deblurring results on Man in Figure 3 with  $19 \times 19$  motion blur kernel and  $\sigma_n = 2.55$ . (a) The original image. (b) The result of IDD-BM3D. (c) The result of IRCNN. (d) The result of DPDNN. (e) Our result.





**Figure 8.** Image deblurring results on Plant in Figure 3 with  $25 \times 25$  Gaussian kernel and  $\sigma_n = 2$ . (a) The original image. (b) The result of NCSR. (c) The result of IRCNN. (d) The result of DPDNN. (e) Our result.

In addition to the good performance of our method on the motion blur kernel, our method also works well on the Gaussian kernel. As shown in Figure 8, our result has high contrast and sharp edges. We also test our network on the Kodak24 dataset. The PSNR and SSIM values are summarized in Table 6. Our results are superior to the other results.

**Table 6.** The Average PSNR Values of The Test Methods For Deblurring on The Kodak24 Dataset.

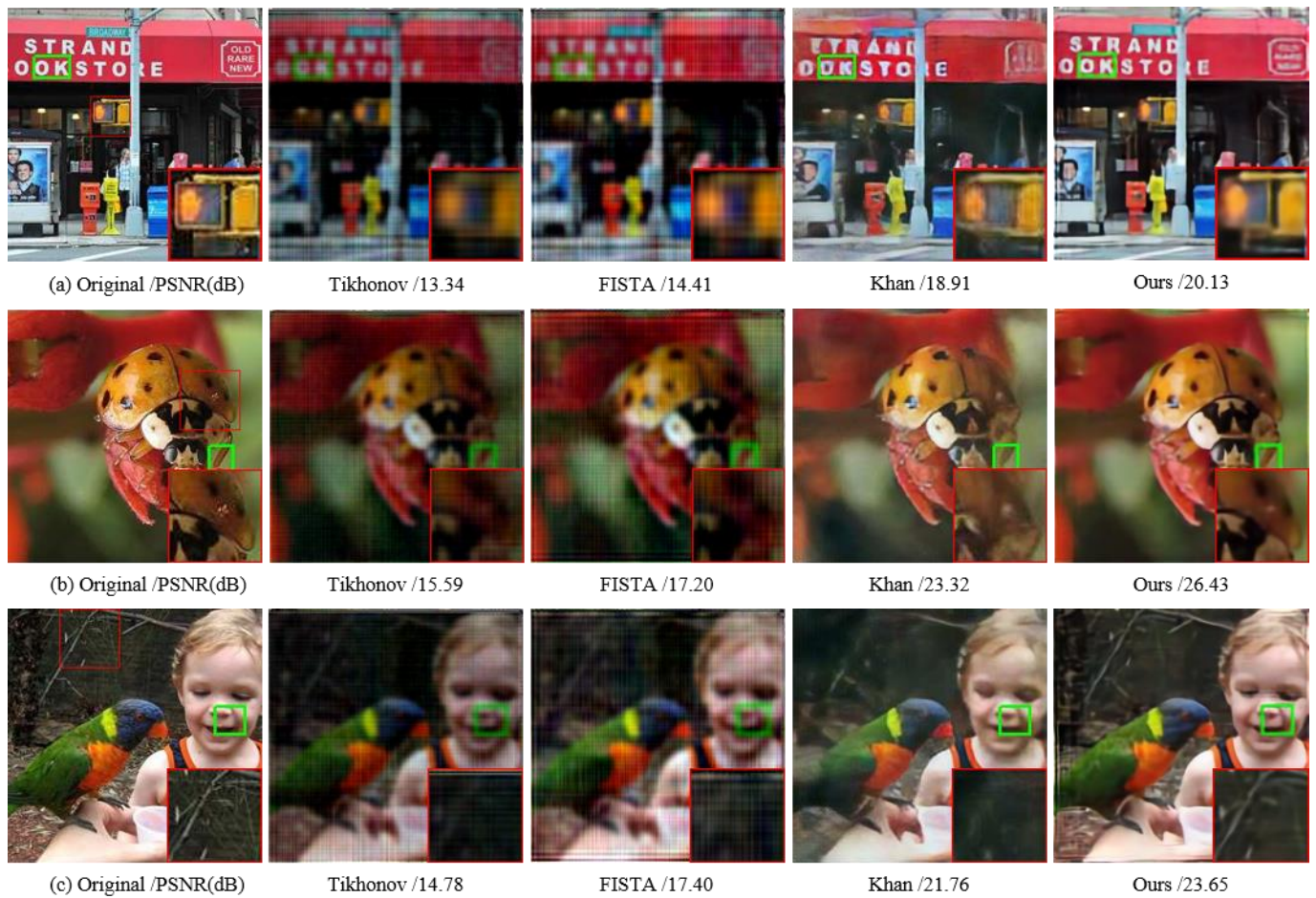
Method	Gaussian Blur with Standard Deviation $1.6, \sigma_n = 2$		$17 \times 17$ Motion Blur Kernel of [80], $\sigma_n = 2.55$		$17 \times 17$ Motion Blur Kernel of [80], $\sigma_n = 7.65$		$19 \times 19$ Motion Blur Kernel of [80], $\sigma_n = 2.55$		$19 \times 19$ Motion Blur Kernel of [80], $\sigma_n = 7.65$	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
IDD-BM3D [11]	29.54	0.828	30.36	0.821	26.74	0.706	30.30	0.827	26.88	0.713
EPLL [5]	29.48	0.823	31.13	0.842	26.82	0.706	31.49	0.852	27.09	0.715
NCSR [26]	29.96	0.833	-	-	-	-	-	-	-	-
IRCNN [61]	29.99	0.831	31.33	0.849	28.28	0.765	30.88	0.841	28.52	0.773
DPDNN [51]	30.15	0.842	31.57	0.860	28.38	0.765	32.00	0.871	28.72	0.780
DPDNN-AS	30.29	0.847	31.84	0.870	28.58	0.775	32.28	0.879	28.89	0.786
Ours	<b>30.43</b>	<b>0.850</b>	<b>31.94</b>	<b>0.874</b>	<b>28.68</b>	<b>0.779</b>	<b>32.33</b>	<b>0.882</b>	<b>28.99</b>	<b>0.790</b>

#### 4.4. Lensless Imaging

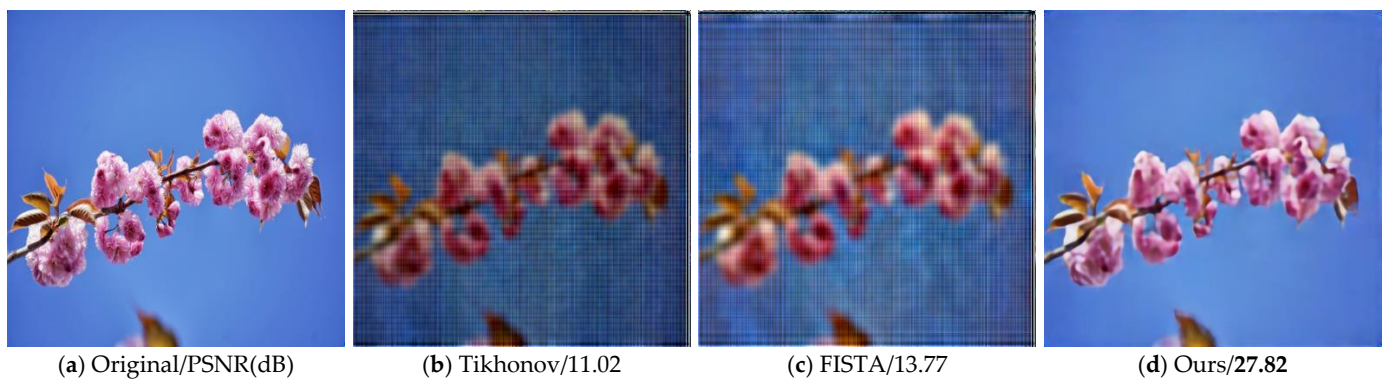
In this section, we apply our network to the lensless FlatCam. The imaging model is as in Equation (11). The corresponding analytic solution is shown in Equation (13).  $\hat{x}_0$  is the least squares estimation with the Tikhonov regularization term. We use the training set from the MLS (maximum length sequence) mask in [76]. There are 10,000 pairs of data in this training set. The degraded measurement is  $2048 \times 2048$  in size. The ground truth is  $512 \times 512$  in size. More experimental details are described in [76]. The test images come from two places: ImageNet [82] and the valid set of DIV2K [81]. As with the training set, we displayed the test images on the screen and then record the values on the CMOS. For the restoration of FlatCam images, the batch size was 8, the initial value of  $\mu_0$  was 0.001, and the learning rate was 0.0005. Similarly, the learning rate was halved every five epochs.

After the model was trained, we compared it with the existing methods, such as Tikhonov [21], FISTA [83], and Khan’s method FlatNet [23]. Figure 9 shows the results on three images of ImageNet. In Figure 9, Khan’s results are derived from [23], and other results are obtained by our programming. Since our original images are from [23], we discarded the pixels in the lower right corner when calculating the PSNR. As shown in Figure 9, our method outperforms the other three methods. Our results are closest to the ground truth without color distortion. To test the image in DIV2K’s valid set, we resized the original image to 512. Figure 10 shows the results on 0898.png of DIV2K’s valid set. As we can see, our overall recovery rate is good, but there is still a blur in the detail area due to the ill condition of FlatCam. The average PSNR and SSIM values for 100 valid images in the DIV2K valid set are summarized in Table 7.





**Figure 9.** FlatCam imaging results on three images of ImageNet. (a) The results of image ‘Store’. (b) The results of image ‘Insect’. (c) The results of image ‘Boy’.



**Figure 10.** FlatCam imaging results on 0898.png of DIV2K dataset. (a) The original image. (b) The result of Tikhonov. (c) The result of FISTA. (d) Our result.

**Table 7.** The average PSNR and SSIM values of 100 images in DIV2K’s valid set.

Method	PSNR	SSIM
Tikhonov [21]	12.93	0.39
FISTA [83]	14.91	0.42
Ours	24.63	0.78

## 5. Conclusions

In this paper, we propose a DCNN denoiser based unrolled network for image restoration. We unfold the tedious iterative process in the model-based method into an end-to-end network consisting of several iterations, each of which has an analytic solution update step and a small multiscale deep denoiser network. Every DCNN serves as a denoiser rather than as the whole inverse process, which makes the network function easier to realize. In this way, our method can take advantage of optimization and the DCNNs. Specifically, we solve the convex problem in the form of an analytic solution that is faster than gradient descent under this framework. In addition, we use different multiscale prior networks in different iterations to better accommodate the image features. Compared with a fixed DCNN, it greatly reduces the number of computations when the total parameters are the same and the number of iterations is the same. Under an unrolled optimization framework, our method incorporates the physical model into the overall network, which provides a guarantee of high-quality image restoration. Visual effects and quantitative evaluation of the method for three IR tasks, including denoising, deblurring, and lensless imaging, indicate that our method achieves excellent performance in high quality image reconstruction.

**Author Contributions:** Conceptualization, methodology and writing, X.T.; formal analysis and software, H.Z.; validation and supervision, Y.C. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research is supported by Advanced Science Key Research Project, Chinese Academy of Science (QYZDJ-SSW-JSC038); Key Research Project of International Cooperation, Chinese Academy of Science (181722KYSB20180015); National Natural Science Foundation of China (11903036, 61805243).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** The data that support the findings of this study are available from the corresponding author upon reasonable request.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Boyat, A.K.; Joshi, B.K. A review paper: Noise models in digital image processing. *arXiv* **2015**, arXiv:1505.03489. [[CrossRef](#)]
2. Yang, C.; Feng, H.; Xu, Z.; Chen, Y.; Li, Q. Image Deblurring Utilizing Inertial Sensors and a Short-Long-Short Exposure Strategy. *IEEE Trans. Image Process.* **2020**, *29*, 4614–4626. [[CrossRef](#)]
3. Zhang, L.; Zuo, W. Image Restoration: From Sparse and Low-rank Priors to Deep Priors. *IEEE Signal. Process. Mag.* **2017**, *34*, 172–179. [[CrossRef](#)]
4. Dabov, K.; Foi, A.; Katkovnik, V.; Egiazarian, K. Image denoising by sparse 3-d transform-domain collaborative filtering. *IEEE Trans. Image Process.* **2007**, *16*, 2080–2095. [[CrossRef](#)] [[PubMed](#)]
5. Zoran, D.; Weiss, Y. From learning models of natural image patches to whole image restoration. In Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 468–479.
6. Mairal, J.; Bach, F.; Ponce, J.; Sapiro, G.; Zisserman, A. Non-local sparse models for image restoration. In Proceedings of the IEEE International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 2272–2279.
7. Elad, M.; Aharon, M. Image denoising via Sparse and Redundant Representations over Learned Dictionaries. *IEEE Trans. Image Process.* **2006**, *15*, 3736–3745. [[CrossRef](#)] [[PubMed](#)]
8. Guo, S.; Yan, Z.; Zhang, K.; Zuo, W.; Zhang, L. Toward convolutional blind denoising of real photographs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 1712–1722.
9. Anwar, S.; Barnes, N. Real image denoising with feature attention. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 3155–3164.
10. Chang, M.; Li, Q.; Feng, H.; Xu, Z. Spatial-Adaptive Network for Single Image Denoising. *arXiv* **2020**, arXiv:2001.10291.
11. Danielyan, A.; Katkovnik, V.; Egiazarian, K. BM3D Frames and Variational Image Deblurring. *IEEE Trans. Image Process.* **2012**, *21*, 1715–1728. [[CrossRef](#)] [[PubMed](#)]
12. Ji, H.; Wang, K. Robust Image Deblurring With an Inaccurate Blur Kernel. *IEEE Trans. Image Process.* **2012**, *21*, 1624–1634. [[CrossRef](#)] [[PubMed](#)]
13. Schmidt, U.; Rother, C.; Nowozin, S.; Jancsary, J.; Roth, S. Discriminative Non-blind Deblurring. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 604–611.

14. Pan, J.; Sun, D.; Pfister, H.; Yang, M. Blind image deblurring using dark channel prior. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1628–1636.
15. Kupyn, O.; Budzan, V.; Mykhailych, M.; Mishkin, D.; Matas, J. DeblurGAN: Blind Motion Deblurring Using Conditional Adversarial Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 8183–8192.
16. Yang, J.; Wright, J.; Huang, T.S.; Ma, Y. Image Super-resolution via Sparse Representation. *IEEE Trans. Image Process.* **2010**, *19*, 2861–2873. [[CrossRef](#)] [[PubMed](#)]
17. Egiazarian, K.; Katkovnik, V. Single image super-resolution via BM3D sparse coding. In Proceedings of the European Signal Processing Conference, Nice, France, 31 August–4 September 2015; pp. 2849–2853.
18. Dong, C.; Loy, C.C.; He, K.; Tang, X. Image Super-Resolution Using Deep Convolutional Networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *38*, 295–307. [[CrossRef](#)]
19. Shi, W.; Caballero, J.; Huszar, F.; Totz, J.; Aitken, A.P.; Bishop, R.; Rueckert, D.; Wang, Z. Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1874–1883.
20. Tai, Y.; Yang, J.; Liu, X. Image Super-resolution via Deep Recursive Residual Network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3147–3155.
21. Asif, M.S.; Ayremlou, A.; Sankaranarayanan, A.; Veeraraghavan, A.; Baraniuk, R.G. FlatCam: Thin, Lensless Cameras Using Coded Aperture and Computation. *IEEE Trans. Comput. Imaging* **2017**, *3*, 384–397. [[CrossRef](#)]
22. Canh, T.N.; Nagahara, H. Deep Compressive Sensing for Visual Privacy Protection in FlatCam Imaging. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Seoul, Korea, 27–28 October 2019; pp. 3978–3986.
23. Khan, S.S.; Adarsh, V.R.; Boominathan, V.; Tan, J.; Veeraraghavan, A.; Mitra, K. Towards photorealistic reconstruction of highly multiplexed lensless images. In Proceedings of the IEEE International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 7860–7869.
24. Monakhova, K.; Yurtsever, J.; Kuo, G.; Antipa, N.; Yanny, K.; Waller, L. Learned reconstructions for practical mask-based lensless imaging. *Opt. Express* **2019**, *27*, 28075–29090. [[CrossRef](#)] [[PubMed](#)]
25. Ledig, C.; Theis, L.; Huszar, F.; Caballero, J.; Cunningham, A.; Acosta, A.; Aitken, A.; Tejani, A.; Totz, J.; Wang, Z.; 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, Honolulu, HI, USA, 21–26 July 2017; pp. 4681–4690.
26. Dong, W.; Zhang, L.; Shi, G.; Li, X. Nonlocally centralized sparse representation for image restoration. *IEEE Trans. Image Process.* **2013**, *22*, 1620–1630. [[CrossRef](#)] [[PubMed](#)]
27. Krishnan, D.; Fergus, R. Fast image deconvolution using hyper-Laplacian priors. In *Advances In Neural Information Processing Systems*; Curran Associates Inc.: Red Hook, NY, USA, 2009; pp. 1033–1041.
28. Bioucas-Dias, J.M.; Ario, M.; Figueiredo, A.T. A new TwIST: Two-step iterative shrinkage/thresholding algorithms for image restoration. *IEEE Trans. Image Process.* **2007**, *16*, 2992–3004. [[CrossRef](#)]
29. Burger, H.C.; Schuler, C.J.; Harmeling, S. Image denoising: Can plain neural networks compete with BM3D? In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 2392–2399.
30. Zhang, K.; Zuo, W.; Chen, Y.; Meng, D.; Zhang, L. Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Trans. Image Process.* **2017**, *26*, 3142–3155. [[CrossRef](#)]
31. Xu, L.; Ren, J.S.; Liu, C.; Jia, J. Deep convolutional neural network for image deconvolution. In *Advances in Neural Information Processing Systems*; MIT Press: Cambridge, MA, USA, 2014; pp. 1790–1798.
32. Kim, J.; Lee, J.K.; Lee, K.M. Accurate image super-resolution using very deep convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 1646–1654.
33. Osher, S.; Burger, M.; Goldfarb, D.; Xu, J.; Yin, W. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling Simul.* **2005**, *4*, 460–489. [[CrossRef](#)]
34. Mairal, J.; Elad, M.; Sapiro, G. Sparse representation for color image restoration. *IEEE Trans. Image Process.* **2008**, *17*, 53–69. [[CrossRef](#)]
35. Aharon, M.; Elad, M.; Bruckstein, A. K-SVD, An algorithm for designing overcomplete dictionaries for sparse representation. *IEEE Trans. Signal Process.* **2006**, *54*, 4311–4322. [[CrossRef](#)]
36. Dong, W.; Li, X.; Zhang, L.; Shi, G. Sparsity-based image denoising via dictionary learning and structural clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA, 20–25 June 2011; pp. 457–464.
37. Buades, A.; Coll, B.; Morel, J.M. A non-local algorithm for image denoising. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA, 20–25 June 2005; Volume 2, pp. 60–65.
38. Xu, J.; Zhang, L.; Zuo, W.; Zhang, D.; Feng, X. Patch group based nonlocal self-similarity prior learning for image denoising. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 244–252.
39. Dong, W.; Shi, G.; Li, X. Nonlocal image restoration with bilateral variance estimation: A low-rank approach. *IEEE Trans. Image Process.* **2013**, *22*, 700–711. [[CrossRef](#)]
40. Gu, S.; Zhang, L.; Zuo, W.; Feng, X. Weighted nuclear norm minimization with application to image denoising. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 2862–2869.



41. Barbu, A. Training an active random field for real-time image denoising. *IEEE Trans. Image Process.* **2009**, *18*, 2451–2462. [[CrossRef](#)]
42. Roth, S.; Black, M.J. Field of experts. *Int. J. Comput. Vis.* **2009**, *82*, 205–229. [[CrossRef](#)]
43. Donoho, D.L. De-noising by soft-thresholding. *IEEE Trans. Inf. Theory* **1995**, *41*, 613–627. [[CrossRef](#)]
44. Buades, A.; Coll, B.; Morel, J.M. Image denoising methods. A new nonlocal principle. *SIAM Rev.* **2010**, *52*, 113–147. [[CrossRef](#)]
45. Cai, J.; Candes, E.J.; Shen, Z. A singular value thresholding algorithm for matrix completion. *SIAM J. Optim.* **2010**, *20*, 1956–1982. [[CrossRef](#)]
46. Sun, J.; Tappen, M.F. Separable Markov random field model and its application in low level vision. *IEEE Trans. Image Process.* **2013**, *22*, 402–407. [[CrossRef](#)] [[PubMed](#)]
47. Schmidt, U.; Roth, S. Shrinkage fields for effective image restoration. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 2774–2781.
48. Zhang, K.; Zuo, W.; Zhang, L. FFDNet: Toward a fast and flexible solution for CNN-based image denoising. *IEEE Trans. Image Process.* **2018**, *27*, 4608–4622. [[CrossRef](#)] [[PubMed](#)]
49. Zhang, Y.; Tian, Y.; Kong, Y.; Fu, B.Z.Y. Residual Dense Network for Image Restoration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 2480–2495. [[CrossRef](#)] [[PubMed](#)]
50. Venkatakrisnan, S.V.; Bouman, C.A.; Wohlberg, B. Plug-and-play priors for model based reconstruction. In Proceedings of the IEEE Global Conference on Signal and Information Processing, Austin, TX, USA, 3–5 December 2013; pp. 945–948.
51. Dong, W.; Wang, P.; Yin, W.; Shi, G.; Wu, F.; Lu, X. Denoising prior driven deep neural network for image restoration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2019**, *41*, 2305–2318. [[CrossRef](#)]
52. Tai, Y.; Yang, J.; Liu, X.; Xu, C. MemNet: A persistent memory network for image restoration. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4539–4547.
53. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
54. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241.
55. Goodfellow, I.J.; Abadie, J.P.; Mirza, M.; Xu, B.; Farley, D.W.; Ozair, S.; Courville, A. Generative Adversarial Networks. *arXiv* **2014**, arXiv:1406.2661.
56. Johnson, J.; Alahi, A.; Li, F. Perceptual loss for real-time style transfer and super-resolution. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 694–711.
57. Lofte, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv* **2015**, arXiv:1502.03167.
58. Zhang, J.; He, T.; Sra, S.; Jadbabaie, A. Why gradient clipping accelerates training: Theoretical justification for adaptivity. *arXiv* **2019**, arXiv:1905.11881.
59. Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. In Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, Sardinia, Italy, 13–15 May 2010; pp. 249–256.
60. Chen, Y.; Pock, T. Trainable nonlinear reaction diffusion: A flexible framework for fast and effective image restoration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 1256–1272. [[CrossRef](#)] [[PubMed](#)]
61. Zhang, K.; Zuo, W.; Gu, S.; Zhang, L. Learning deep CNN denoiser prior for image restoration. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 3929–3938.
62. Bertocchi, C.; Chouzenoux, E.; Corbineau, M.C.; Pesquet, J.C.; Prato, M. Deep unfolding of a proximal interior point method for image restoration. *Inverse Probl.* **2020**, *36*, 34005. [[CrossRef](#)]
63. Teodoro, A.M.; Bioucas-Dias, J.M.; Figueiredo, M.A.T. Image restoration and reconstruction using variable splitting and class-adapted image priors. In Proceedings of the IEEE International Conference on Image Processing, Phoenix, AZ, USA, 25–28 September 2016; pp. 3518–3522.
64. Kamilov, U.S.; Mansour, H.; Wohlberg, B. A Plug-and-play priors approach for solving nonlinear imaging inverse problems. *IEEE Signal. Process. Lett.* **2007**, *24*, 1872–1876. [[CrossRef](#)]
65. Tirer, T.; Giryes, R. Image restoration by iterative denoising and backward projections. *IEEE Trans. Image Process.* **2019**, *28*, 1220–1234. [[CrossRef](#)]
66. Brifman, A.; Romano, Y.; Elad, M. Turning a denoiser into a super-resolver using plug and play priors. In Proceedings of the IEEE International Conference on Image Processing, Phoenix, AZ, USA, 25–28 September 2016; pp. 1404–1408.
67. Sun, Y.; Xu, S.; Li, Y.; Tian, L.; Wohlberg, B.; Kamilov, U.S. Regularized fourier ptychography using an online plug-and-play algorithm. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, Brighton, UK, 12–17 May 2019; pp. 7665–7669.
68. Sreehari, S.; Venkatakrisnan, S.V.; Wohlberg, B.; Buzzard, G.T.; Drummy, L.F.; Simmons, J.P. Plug-and-play priors for bright field electron tomography and sparse interpolation. *IEEE Trans. Comput. Imaging* **2016**, *2*, 408–423. [[CrossRef](#)]
69. Bigdeli, S.; Honzatko, D.; Susstrunk, S.; Dunbar, L.A. Image restoration using plug-and-play cnn map denoisers. *arXiv* **2019**, arXiv:1912.09299.
70. Chan, S.H.; Wang, X.; Elgendy, O.A. Plug-and-play ADMM for image restoration: Fixed-point convergence and applications. *IEEE Trans. Comput. Imaging* **2017**, *3*, 84–98. [[CrossRef](#)]

71. Ryu, E.K.; Liu, J.; Wang, S.; Chen, X.; Wang, Z.; Yin, W. Plug-and-play methods provably converge with properly trained denoisers. *arXiv* **2019**, arXiv:1905.05406.
72. Zhang, J.; Ghanem, B. ISTA-Net: Interpretable Optimization-inspired deep network for image compressive sensing. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 1828–1837.
73. Jeon, D.S.; Baek, S.H.; Yi, S.; Fu, Q.; Dun, X.; Heidrich, W.; Kim, M.H. Compact snapshot hyperspectral imaging with diffracted rotation. *ACM Trans. Graph.* **2019**, *38*, 1–13. [[CrossRef](#)]
74. Zhou, H.; Feng, H.; Xu, W.; Xu, Z.; Li, Q.; Chen, Y. Deep denoiser prior based deep analytic network for lensless image restoration. *Opt. Express* **2021**, *29*, 27237–27253. [[CrossRef](#)]
75. Maas, A.L.; Hannun, A.Y.; Ng, A.Y. Rectifier nonlinearities improve neural network acoustic models. In Proceedings of the 30th International Conference on International Conference on Machine Learning, Atlanta, GA, USA, 16–21 June 2013; Volume 28.
76. Zhou, H.; Feng, H.; Hu, Z.; Xu, Z.; Li, Q.; Chen, Y. Lensless cameras using a mask based on almost perfect sequence through deep learning. *Opt. Express* **2020**, *28*, 30248–30262. [[CrossRef](#)]
77. Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G.S.; Davis, A.; Dean, J.; Devin, M.; et al. TensorFlow: Large-scale machine learning on Heterogeneous distributed systems. *arXiv* **2016**, arXiv:1603.04467.
78. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
79. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [[CrossRef](#)] [[PubMed](#)]
80. Levin, A.; Weiss, Y.; Durand, F.; Freeman, W.T. Understanding and evaluating blind deconvolution algorithms. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 1964–1971.
81. Agustsson, E.; Timofte, R. Ntire 2017 challenge on single image super-resolution: Dataset and study. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, HI, USA, 21–26 July 2017; pp. 126–135.
82. Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; et al. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252. [[CrossRef](#)]
83. Beck, A.; Teboulle, M. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM J. Imaging Sci.* **2019**, *2*, 183–202. [[CrossRef](#)]