Deep Generalized Unfolding Networks for Image Restoration

Chong Mou†, Qian Wang†, Jian Zhang†,‡
†Peking University Shenzhen Graduate School, Shenzhen, China
‡Peng Cheng Laboratory, Shenzhen, China
eechongm@gmail.com; lucywq1028@gmail.com; zhangjian.sz@pku.edu.cn

Abstract

Deep neural networks (DNN) have achieved great success in image restoration. However, most DNN methods are designed as a black box, lacking transparency and interpretability. Although some methods are proposed to combine traditional optimization algorithms with DNN, they usually demand pre-defined degradation processes or hand-crafted assumptions, making it difficult to deal with complex and real-world applications. In this paper, we propose a ***D****eep* ***G****eneralized* ***U****nfolding* ***Net****work (DGUNet) for image restoration. Concretely, without loss of interpretability, we integrate a gradient estimation strategy into the gradient descent step of the Proximal Gradient Descent (PGD) algorithm, driving it to deal with complex and real-world image degradation. In addition, we design inter-stage information pathways across proximal mapping in different PGD iterations to rectify the intrinsic information loss in most deep unfolding networks (DUN) through a multi-scale and spatial-adaptive way. By integrating the flexible gradient descent and informative proximal mapping, we unfold the iterative PGD algorithm into a trainable DNN. Extensive experiments on various image restoration tasks demonstrate the superiority of our method in terms of state-of-the-art performance, interpretability, and generalizability. The source code is available at github.com/MC-E/DGUNet.*

1. Introduction

Image restoration (IR) aims to recover the high-quality image $\mathbf{x}$ from its degraded measurement $\mathbf{y}$. The degradation process is generally defined as:

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{n}, \tag{1}$$

where $\mathbf{A}$ is the degradation matrix, and $\mathbf{n}$ represents the additive noise. It is typically an ill-posed problem. According to $\mathbf{A}$, IR can be categorized into many subtasks, *e.g.*, image denoising, deblurring, deraining, compressive sensing. In the past few decades, IR has been extensively studied, leading to three main active research topics, *i.e.*, model-based methods , deep learning methods , and hybrid methods.

This work was supported in part by Shenzhen Fundamental Research Program (No.GXWD20201231165807007-20200807164903001) and National Natural Science Foundation of China (61902009). (*Corresponding author: Jian Zhang.*)

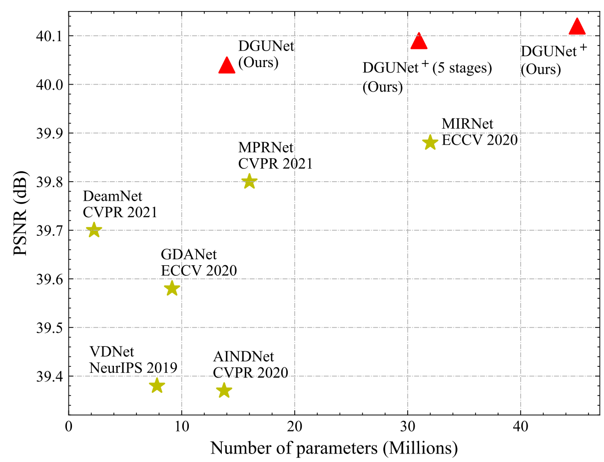

Figure 1. Real image denoising performance (y-axis) of our DGUNet and some recent denoisers (VDNet [75], GDANet [76], AINDNet [33], MIRNet [78], DeamNet [53], MPRNet [79]) under different parameter capacities (x-axis) on DND [52] dataset.

Model-based methods (*e.g.*, [7, 12, 17, 25, 32, 56]) usually formulate IR as a Bayesian problem, solving Eq. (1) under a unified MAP (maximizing a posterior) framework:

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\texttt{argmax}} \log P(\mathbf{x}|\mathbf{y}) = \underset{\mathbf{x}}{\texttt{argmax}} \log P(\mathbf{y}|\mathbf{x}) + \log P(\mathbf{x}), \tag{2}$$

where $\log P(\mathbf{y}|\mathbf{x})$ and $\log P(\mathbf{x})$ represent the data fidelity and regularization terms, respectively. The data fidelity term is usually defined as an ℓ_2 norm, expressing Eq. (2) as the following energy function:

$$\hat{\mathbf{x}} = \underset{\mathbf{x}}{\texttt{argmin}} \frac{1}{2}||\mathbf{y} - \mathbf{A}\mathbf{x}||_2^2 + \lambda J(\mathbf{x}), \tag{3}$$

where λ is a hyper-parameter to weight the regularization term $J(\mathbf{x})$. The data fidelity term guarantees the solution accords with degradation. The regulation term alleviates the ill-posed problem by enforcing desired property, which involves sophisticated priors, *e.g.*, total variation [49], sparse representation [15, 18, 42], low-rank [23], and self-similarity [7, 12]. However, the representation ability of handcrafted design is limited, leading to unstable results, and they are usually time-consuming in inference.

Recently, deep-learning IR [13, 91–93] has achieved impressive success, as they can learn strong priors from large-scale datasets. Up to now, numerous function units have been proposed. [62] proposed a memory strategy to broadcast useful information in different layers. [6, 24, 33, 75, 79] utilized hourglass-shaped architectures to explore multi-scale features. Some non-local methods [40, 45, 46] were proposed to enlarge the receptive field. Although the promising performance and fast inference, the black box design makes it hard to analyze the role of different components, and performance gains are often attributed to stacking new modules at the price of increased model complexity.

To combine interpretability and adaptivity, some hybrid methods proposed integrating deep networks into classic optimization algorithms. For instance, deep plug-and-play (PNP) methods [44, 74, 89, 92, 94] integrated pre-trained CNN denoiser, as the prior, into iterative optimization frameworks for different IR tasks. Unfortunately, they usually suffer from time-consuming inference. Recently, deep unfolding networks (DUN) [16, 53, 83, 88] proposed optimizing all parameters end-to-end, delivering better performance and faster inference. However, due to the interpretable design, most of them require known degradation processes to derive the solution. Nevertheless, the degradation processes in real-world applications are complicated and unassured with signal-dependent and spatially variant distribution. Thus, most DUN methods make handcrafted degradation assumptions [16] or explicitly provide the network with degradation factors [88] to deal with pre-defined image degradation problems. In addition, since traditional model-based methods output an image in each iteration, the corresponding DUN has to adopt the input and output of each stage as an image. This inherent design inevitably results in feature-to-image information distortion. Such information loss in DUN has little notice in existing works.

To rectify the above issues and bridge the gap between model-based methods and deep learning methods, we propose a deep generalized unfolding network (DGUNet). On the one hand, our method has good interpretability as model-based methods by formulating the model design via a Proximal Gradient Descent (PGD) algorithm. On the other hand, similar to deep learning methods, our method is trained end-to-end with an unhindered feature pathway and can be easily applied to complex and real-world applications. To achieve this, we first integrate a gradient estimation strategy to the gradient descent step of the PGD algorithm to predict the gradient in degradation-unknown cases. We then design inter-stage information pathways to compensate for the intrinsic information loss in DUN. To summarize, this work has the following contributions:

- The iterative optimization step of PGD algorithm is used to guide the mode design, leading to an end-to-end trainable and also interpretable model (DGUNet).
- Our DGUNet presents a general CNN-based implementation of DUN by combining a gradient estimation strategy into the PGD algorithm, enabling PGD to be easily applied to complex and real-world IR tasks.
- We design inter-stage information pathways in the DUN framework to broadcast multi-scale features in a spatial-adaptive normalization way, which rectifies the intrinsic information loss in most DUN methods.
- Extensive experiments demonstrate that our method can solve general IR tasks with state-of-the-art performance (including **twelve** synthetic and real-world test sets) and attractive complexity (see Fig. 1).

2. Related Works

2.1. Model-based Image Restoration Methods

As mentioned previously, model-based methods [85, 87, 98, 99] usually solve IR in a Bayesian perspective, which is formulated into a MAP optimization problem as Eq. (3), containing a data fidelity term and a regularization term. HQS [27], ADMM [5] and PGD [4] are commonly used optimization algorithms. These methods usually decoupled the data fidelity term and regularization term of the objective function, resulting in an iterative scheme consisting of alternately solving a data subproblem and a prior subproblem. For instance, [100] integrated Gaussian Mixture prior to HQS. In [28], Heide *et al.* used an alternative to ADMM and HQS to decouple the data term and prior term. [64] plugged class-specific Gaussian mixture denoiser into ADMM to solve image deblurring and compressive sensing.

2.2. Deep Learning Image Restoration Methods

Motivated by the great success of deep neural networks (DNN), DNN-based methods have been widely used in low-level image processing tasks. [13, 14] are early attempts at applying convolutional neural networks (CNN) for IR. Subsequently, Zhang *et al.* proposed DnCNN [91], significantly improving the restoration performance by residual learning. [62] proposed dense-connected memory block to collect useful information from preceding layers. [95, 96] proposed stacking residual blocks (RB) without Batch Normalization [30] to extend the network depth. In addition to estimating the clean image with a fixed scale, some hourglass-shaped networks [6, 24, 33, 75, 79] were proposed to explore multi-scale feature maps for IR.

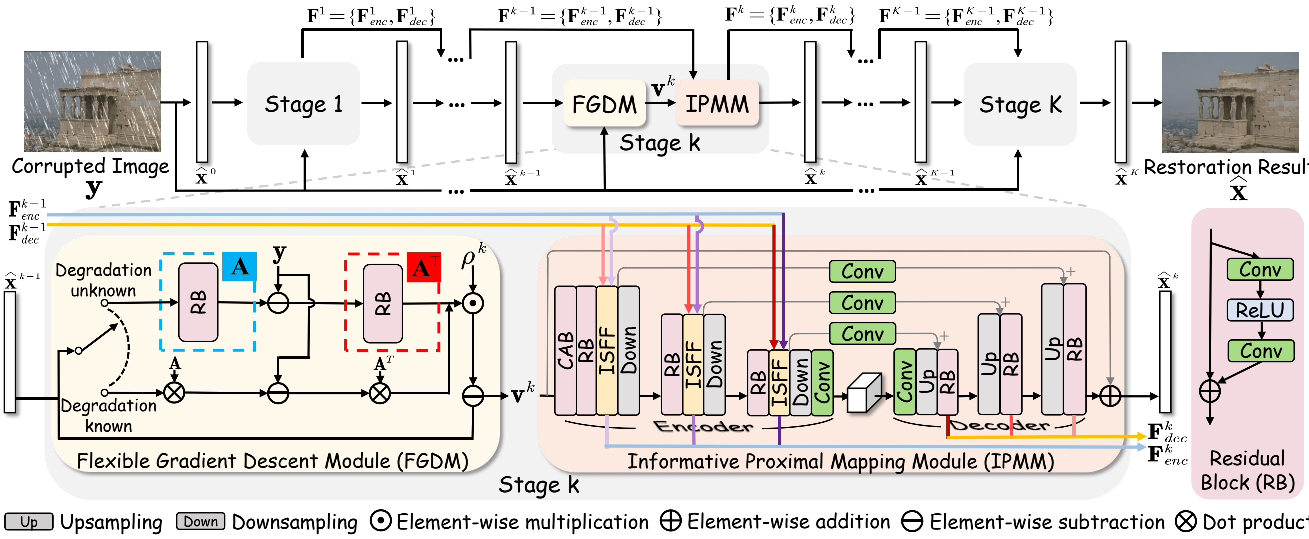

Figure 2. Illustration of our proposed deep generalized unfolding network (DGUNet). We present the overall architecture in the first row, mainly composed of several stages. Each stage corresponds to an iteration in the PGD algorithm. The second row presents the detailed design of each stage, containing a flexible gradient descent module (FGDM) and an informative proximal mapping module (IPMM).

2.3. Deep Unfolding Networks

The main idea of deep unfolding networks (DUN) is that conventional iterative optimization algorithms can be implemented equivalently by a stack of recurrent DNN blocks. Such correspondence was originally applied in deep plug-and-play (PNP) methods [44, 57, 74, 92, 94], which utilize trained denoiser to implicitly express the regularization term $J(\mathbf{x})$ as a denoising problem. Inspired by PNP, DUN methods are trained in an end-to-end manner by jointly optimizing trainable denoisers in specific tasks. For instance, [16] jointly optimized an UNet as the proximal mapping in the ADMM [5] algorithm. Nevertheless, its network structure is closely related to the handcrafted degradation assumptions to deal with pre-defined image degradation. [88] used a ResUNet to replace the proximal mapping in the HQS [27] algorithm. However, the degradation process is also manually designed, and its network requires scale factor, blur kernel, and noise level as additional inputs, causing the performance to depend largely on the accuracy of provided degradation factors. [60, 73, 83] solved compressive sensing by PGD algorithm [4] with known degradation process. Moreover, most DUN methods are beset by information loss due to the feature-to-image transformation at the end of each stage. Though the skip connections in [48] benefit the information transfer, its implementation remains primitive, *e.g.*, feature fusion is performed by concatenation on a single decoder layer of the proximal mapping module.

3. Methodology

In this section, we first briefly review the traditional Proximal Gradient Descent (PGD) algorithm and then elaborate on our proposed DGUNet.

3.1. Traditional Proximal Gradient Descent

Technically, the PGD algorithm approximatively expresses Eq. (3) as an iterative convergence problem through the following iterative function:

$$\hat{\mathbf{x}}^k = \underset{\mathbf{x}}{\texttt{argmin}}\, \frac{1}{2}||\mathbf{x}-(\hat{\mathbf{x}}^{k-1} - \rho\nabla g(\hat{\mathbf{x}}^{k-1}))||_2^2 + \lambda J(\mathbf{x}), \tag{4}$$

where $\hat{\mathbf{x}}^k$ refers to the output of the k-th iteration, and $g(\cdot)$ represents the data fidelity term in Eq. (3). ∇ is the differential operator, weighted by the step size ρ. Mathematically, the red part of the above function is a gradient descent operation, and the blue part can be solved by the proximal operator $\texttt{prox}_{\lambda,J}$. Thus, it leads to two subproblems, *i.e.*, gradient descent (Eq. (5a)) and proximal mapping (Eq. (5b)):

$$\mathbf{v}^k = \hat{\mathbf{x}}^{k-1} - \rho\mathbf{A}^\top(\mathbf{A}\hat{\mathbf{x}}^{k-1} - \mathbf{y}), \tag{5a}$$

$$\hat{\mathbf{x}}^k = \texttt{prox}_{\lambda,J}(\mathbf{v}^k). \tag{5b}$$

The PGD algorithm iteratively updates $\mathbf{v}^k$ and $\hat{\mathbf{x}}^k$ until convergence. ISTA [4] is a typical PGD-based algorithm in which the regulation term is defined as an ℓ_1 norm, *i.e.*, $J(\mathbf{x}) = ||\mathbf{x}||_1$. Thus, the proximal mapping in ISTA is derived as a soft thresholding function: $\texttt{prox}_{\lambda,J}(\mathbf{v}^k) = \texttt{sign}(\mathbf{v}^k)\texttt{max}(0, |\mathbf{v}^k| - \lambda)$. However, the handcrafted ℓ_1 regulation has limited representation abilities, and its application is restricted to a few degradation-known tasks (*e.g.*, compressive sensing). Focusing on improving the traditional PGD algorithm, in this paper, we unfold it by deep neural networks with robust and generalized design.

3.2. Proposed Deep Generalized Unfolding Network

The whole network architecture of our proposed DGUNet is presented in Fig. 2, which is an unfolding framework of the Proximal Gradient Descent (PGD) algorithm based on deep neural networks (DNN). Our DGUNet is composed of several repeated stages. Each stage contains a flexible gradient descent module (FGDM) and an informative proximal mapping module (IPMM), corresponding to the gradient descent (Eq. (5a)) and proximal mapping (Eq. (5b)) in an iteration step of the PGD algorithm, respectively. The number of stages is set as seven by default, and they share the same parameters except for the first and last stages. To further improve the model performance, we also present a plus version, dubbed as DGUNet^{+}, in which all stages are parameter-independent.

Flexible Gradient Descent Module. As illustrated in Eq. (5a), the gradient descent step is trivial when the degradation matrix $\mathbf{A}$ is known. However, $\mathbf{A}$ is unknown in some degradation problems, making the gradient calculation (*i.e.*, $\mathbf{A}^{\top}(\mathbf{A}\hat{\mathbf{x}}^{k-1}-\mathbf{y})$) intractable. In this context, we propose a flexible gradient descent module (FGDM), shown in the second row of Fig. 2. It has two model settings to deal with the degradation known and unknown cases reasonably.

In the case of $\mathbf{A}$ is known, we directly use the accurate $\mathbf{A}$ to calculate the gradient. To improve the robustness, we set the step size ρ as a trainable parameter in each stage, leading to the following gradient descent operation:

$$\mathbf{v}^{k}=\hat{\mathbf{x}}^{k-1}-\rho^{k}\mathbf{A}^{\top}(\mathbf{A}\hat{\mathbf{x}}^{k-1}-\mathbf{y}). \tag{6}$$

If $\mathbf{A}$ is unknown, instead of making task-specific assumptions for different degradation problems, we adopt a data-driven strategy to predict the gradient. Technically, we utilize two independent residual blocks, dubbed as $\mathcal{F}_{\mathbf{A}}^{k}$ and $\mathcal{F}_{\mathbf{A}^{\top}}^{k}$, to simulate $\mathbf{A}$ and its transpose $\mathbf{A}^{\top}$ in the k-th stage. The gradient is calculated as $\mathcal{F}_{\mathbf{A}^{\top}}^{k}(\mathcal{F}_{\mathbf{A}}^{k}(\hat{\mathbf{x}}^{k-1})-\mathbf{y})$. Thus without loss of interpretability, the gradient descent in our proposed DGUNet can be defined as the following function in the degradation-unknown cases.

$$\mathbf{v}^{k}=\hat{\mathbf{x}}^{k-1}-\rho^{k}\mathcal{F}_{\mathbf{A}^{\top}}^{k}(\mathcal{F}_{\mathbf{A}}^{k}(\hat{\mathbf{x}}^{k-1})-\mathbf{y}). \tag{7}$$

Informative Proximal Mapping Module. For the solution of Eq. (5b), it is known that, from a Bayesian perspective, it actually corresponds to a denoising problem [9, 74]. In this context, we design an informative proximal mapping module (IPMM), shown in the second row of Fig. 2. Our IPMM is an hourglass-shaped architecture, consisting of an encoder and a decoder to utilize the multi-scale feature maps. Specifically, our IPMM begins with a channel attention block (CAB) to extract shallow features. We employ the residual block (RB) without Batch Normalization [30] to extract features at three scales. Here, we utilize $\mathbf{F}_{enc}^{k}=\{\mathbf{F}_{enc\circledast n}^{k}\}_{n=1}^{3}$ and $\mathbf{F}_{dec}^{k}=\{\mathbf{F}_{dec\circledast n}^{k}\}_{n=1}^{3}$ to represent the encoder and decoder features extracted from the

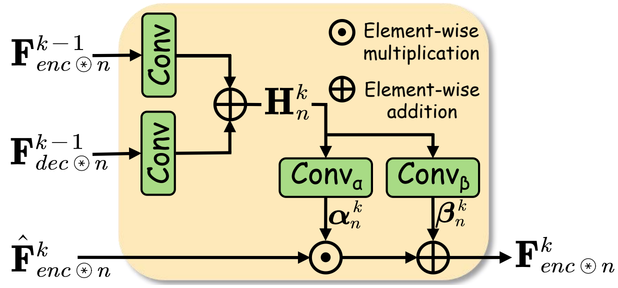

Figure 3. Illustration of our inter-stage feature fusion module (ISFF) at the n-th scale in the k-th stage. The encoder and decoder features from the previous stage are fused to the current stage in a spatial-adaptive normalization manner.

k-th stage at the n-th scale. In order to switch scales in our IPMM, we use 2×2 max-pooling with stride 2 for downsampling, and we use bilinear upsampling followed by a convolutional layer for upsampling. Similar to many competitive denoisers, we add a global pathway from the input to the output, which encourages the network to bypass low-frequency information. At the end of IPMM, we utilize the supervised attention module (SAM) in [79] to extract clean features and then inject them into the next stage through subspace projection [11].

Considering the intrinsic information loss in most DUN methods, we design inter-stage information pathways at each scale to broadcast useful information from encoder and decoder in different stages. For the illustration purpose, we use different colored lines to distinguish encoder and decoder information with different scales in Fig. 2. To fuse the inter-stage information, we design an inter-stage feature fusion submodule (ISFF) at each scale in the encoder. Note that the inter-stage information can also be naturally propagated to the decoder due to the skip-connections between encoder and decoder. The detailed architecture of our ISFF is presented in Fig. 3, which is inspired by [51, 66]. Concretely, at each scale, we transmit encoder and decoder features from the previous stage to the current stage. They are first embedded by two independent 1×1 convolutional layers and merged by the element-wise addition. In the k-th stage, the fusion result at the n-th scale is represented as $\mathbf{H}_{n}^{k}$. It is used to compute two affine parameters $\{\boldsymbol{\alpha}_{n}^{k},\boldsymbol{\beta}_{n}^{k}\}\in\mathbb{R}^{C\times H\times W}$ to transfer the intermediate output $\hat{\mathbf{F}}_{enc\circledast n}^{k}\in\mathbb{R}^{C\times H\times W}$ to an informative one $\mathbf{F}_{enc\circledast n}^{k}\in\mathbb{R}^{C\times H\times W}$, where C, H, and W refer to the size of channel, height, and width, respectively. Mathematically, our proposed inter-stage feature fusion is defined as the following feature representation:

$$\begin{cases}\mathbf{H}_{n}^{k-1}=\mathtt{Conv}(\mathbf{F}_{enc\circledast n}^{k-1})+\mathtt{Conv}(\mathbf{F}_{dec\circledast n}^{k-1})\\ \boldsymbol{\alpha}_{n}^{k},\boldsymbol{\beta}_{n}^{k}=\mathtt{Conv}_{\alpha}(\mathbf{H}_{n}^{k-1}),\mathtt{Conv}_{\beta}(\mathbf{H}_{n}^{k-1})\\ \mathbf{F}_{enc\circledast n}^{k}=\hat{\mathbf{F}}_{enc\circledast n}^{k}\odot\boldsymbol{\alpha}_{n}^{k}+\boldsymbol{\beta}_{n}^{k}.\end{cases} \tag{8}$$

Table 1. Quantitative results (PSNR and SSIM) of image deraining. The best and second-best scores are **highlighted** and <u>underlined</u>.

Method	Test100 [79]		Rain100H [71]		Rain100L [71]		Test2800 [79]		Test1200 [79]		Average	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑
DerainNet [19]	22.77	0.810	14.92	0.592	27.03	0.884	24.31	0.861	23.38	0.835	22.48	0.796
SEMI [68]	22.35	0.788	16.56	0.486	25.03	0.842	24.43	0.782	26.05	0.822	22.88	0.744
DIDMDN [81]	22.56	0.818	17.35	0.524	25.23	0.741	28.13	0.867	29.65	0.901	24.58	0.770
UMRL [72]	24.41	0.829	26.01	0.832	29.18	0.923	29.97	0.905	30.55	0.910	28.02	0.880
RESCAN [38]	25.00	0.835	26.36	0.786	29.80	0.881	31.29	0.904	30.51	0.882	28.59	0.857
PreNet [54]	24.81	0.851	26.77	0.858	32.44	0.950	31.75	0.916	31.36	0.911	29.42	0.897
MSPFN [31]	27.50	0.876	28.66	0.860	32.40	0.933	32.82	0.930	32.39	0.916	30.75	0.903
MPRNet [79]	30.27	0.897	30.41	0.890	36.40	0.965	33.64	0.938	32.91	<u>0.916</u>	32.73	0.921
DGUNet (Ours)	<u>30.32</u>	<u>0.899</u>	<u>30.66</u>	<u>0.891</u>	<u>37.42</u>	<u>0.969</u>	<u>33.68</u>	<u>0.938</u>	**33.23**	**0.920**	<u>33.06</u>	<u>0.923</u>
DGUNet$^+$ (Ours)	**30.86**	**0.907**	**31.06**	**0.897**	**38.25**	**0.974**	**34.01**	**0.942**	<u>33.08</u>	<u>0.916</u>	**33.46**	**0.927**

The above feature fusion process is a standard spatial-adaptive normalization [51]. Unlike conditional normalization methods [30,65], $\boldsymbol{\alpha}_n^k$ and $\boldsymbol{\beta}_n^k$ are not vectors but tensors with spatial dimensions. In this way, while the encoder and decoder obtain multi-scale features, feature maps at each scale can also have the refined memory of previous stages with the well-preserved spatial information, leading to an informative proximal mapping. For the illustration purpose, we utilize $\mathbf{F}^k$ to represent the set of multi-scale encoder and decoder features, *i.e.*, $\mathbf{F}^k = \{\mathbf{F}_{enc}^k, \ \mathbf{F}_{dec}^k\}$. Finally, our IPMM expresses Eq. (5b) as:

$$\hat{\mathbf{x}}^k, \mathbf{F}^k = \texttt{prox}_{\boldsymbol{\theta}^k}(\mathbf{v}^k, \mathbf{F}^{k-1}), \tag{9}$$

where $\boldsymbol{\theta}^k$ denotes the parameters of IPMM in the k-the stage. In the light of the above discussion, we finally define the convergence process of our DGUNet in **Algorithm 1**.

Algorithm 1: Proposed DGUNet

Initialization:
(1) Initialize the iteration depth k=0 and ceiling K;
(2) Initialize the input $\hat{\mathbf{x}}^0 = \mathbf{y}$;
(3) Initialize the inter-stage feature $\mathbf{F}^0 = None$;
while $k < K$ **do**
  if A *is unknown* **then**
    Update $\mathbf{v}^{k+1}$ by Eq. (6);
  else
    Update $\mathbf{v}^{k+1}$ by Eq. (7);
  end
  Update $\hat{\mathbf{x}}^{k+1}$ and $\mathbf{F}^{k+1}$ by Eq. (9);
  $k = k + 1$;
end
Output: $[\hat{\mathbf{x}}^1, \hat{\mathbf{x}}^2, \cdots, \hat{\mathbf{x}}^K]$

3.3. Loss Function Design

Without bells and whistles, we optimize our DGUNet and DGUNet$^+$ with the commonly used ℓ_2 loss function, involving the output from all stages. Specifically, given the degraded measurement $\mathbf{y}$ and the ground-truth image $\mathbf{x}$, the goal of the training is defined as:

$$\mathcal{L}(\boldsymbol{\Omega}) = \sum_{k=1}^{K} \left\| \mathbf{x} - \hat{\mathbf{x}}^k \right\|_2^2, \tag{10}$$

where K refers to the total number of stages, and $\hat{\mathbf{x}}^k$ represents the restoration result from the k-th stage. $\boldsymbol{\Omega} = \{\boldsymbol{\rho}^k, \mathcal{F}_{\mathbf{A}}^k(\cdot), \mathcal{F}_{\mathbf{A}^T}^k(\cdot), \boldsymbol{\theta}^k\}_{k=1}^K$ is the set of trainable parameters of our proposed DGUNet.

4. Experiments

We apply our DGUNet to image deraining, deblurring, denoising, and compressive sensing. For each application, we train and evaluate our DGUNet with standard benchmarks and commonly used settings. The comparison is conducted with several recent methods.

4.1. Training Details

Our DGUNet is trained in an end-to-end manner. For image denoising, image deraining, and image deblurring tasks, we apply the same training strategy as MPRNet [79]. Specifically, we use Adam optimizer [34], with the initial learning rate being 2×10^{-4}. Considering the model depth, we utilize the warming up strategy [26] to gradually improve the learning rate. The network is trained on 256×256 image patches, randomly cropped from training images. The batch size is set as 16 for 4×10^5 iterations. In image compressive sensing, the network is trained on 32×32 image patches, with the learning rate being 1×10^{-4}. The batch size is set as 128 for 200 epochs.

The model training is performed on 2 Nvidia Tesla V100 GPUs and can be completed within three days. For evaluation, we report standard metrics (PSNR and SSIM [67]).

4.2. Image Deraining Results

For image deraining, the training data is the same as MSPFN [31] and MPRNet [79]. Specifically, we use 11,200 clean-rain image pairs in Rain14000 [20], 1,800 image pairs in Rain1800 [71], 700 image pairs in Rain800 [82] and 12

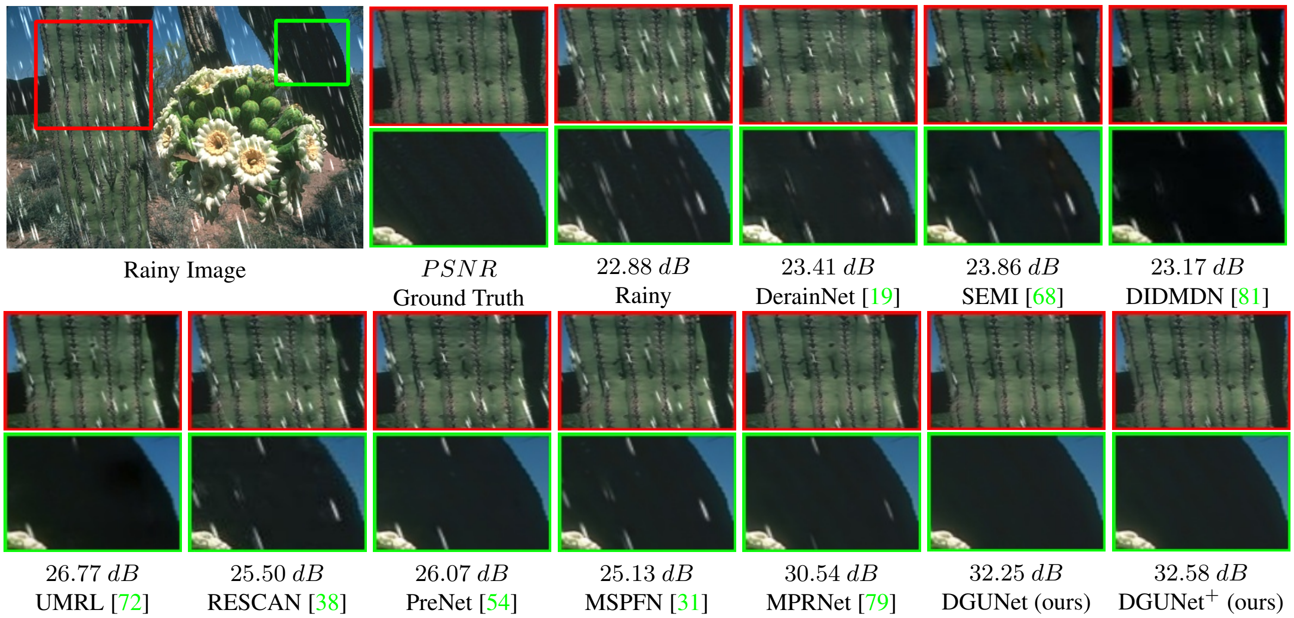

Figure 4. Visual comparison of image deraining. Our DGUNet and DGUNet$^+$ generate more natural details while removing raindrops.

image pairs in Rain12 [39] to train our model. For evaluation, five datasets, including Test2800 [20], Test1200 [81], Test100 [82], Rain100H [71] and Rain100L [71] are utilized as the test sets. We compare our proposed DGUNet with eight competitive methods [19,31,38,54,68,72,79,81]. The quantitative comparison results are presented in Tab. 1. One can see that our DGUNet and DGUNet$^+$ can outperform other methods on all test sets. Concretely, there are $1.85\ dB$, $0.65\ dB$, and $0.73\ dB$ gains compared with the recent MPRNet on Rain100L, Rain100H, and average of these five test sets, respectively. The visual comparison is presented in Fig. 4, showing the better restoration result of our DGUNet and DGUNet$^+$. Especially compared with MSPFN and MPRNet, our method can remove inconspicuous rain lines better and produce more visually satisfying results with vivid details.

4.3. Image Deblurring Results

For image deblurring, similar to [36, 55, 63, 79, 80], we train our model with $2,103$ image pairs from GoPro [47] dataset and evaluate each method on the test sets from GoPro and HIDE [58], which contain $1,111$ and $2,025$ samples, respectively. Unlike pre-defined blur kernels, these two datasets are generated in real scenes, involving real-world degradation factors such as camera response function and human-aware motion blur. We compare our method with several competitive works [50, 61, 80, 90] and the recent best algorithm MPRNet [79]. The quantitative evaluation is summarized in Tab. 2, presenting that our DGUNet and DGUNet$^+$ outperform other methods on these two test sets. Specifically, our DGUNet$^+$ outperforms MPRNet with $0.51\ dB$ and $0.44\ dB$ on GoPro and HIDE test sets.

Table 2. Image deblurring results on GoPro [47] and HIDE [58]. The best and second-best scores are **highlighted** and <u>underlined</u>.

Method	GoPro [47]		HIDE [58]	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑
Xu et al. [70]	21.00	0.741	-	-
Hyun et al. [29]	23.64	0.824	-	-
Whyte et al. [69]	24.60	0.846	-	-
Gong et al. [22]	26.40	0.863	-	-
DeblurGAN [36]	28.70	0.858	24.51	0.871
Nah et al. [47]	29.08	0.914	25.73	0.874
Zhang et al. [84]	29.19	0.931	-	-
DeblurGAN-v2 [37]	29.55	0.934	26.61	0.875
SRN [63]	30.26	0.934	28.36	0.915
Shen et al. [58]	-	-	28.89	0.930
Gao et al. [21]	30.90	0.935	29.11	0.913
DBGAN [90]	31.10	0.942	28.94	0.915
MT-RNN [50]	31.15	0.945	29.15	0.918
DMPHN [80]	31.20	0.940	29.09	0.924
Suin et al. [61]	31.85	0.948	29.98	0.930
MPRNet [79]	32.66	0.959	30.96	0.939
DGUNet (Ours)	<u>32.71</u>	<u>0.960</u>	<u>30.96</u>	<u>0.940</u>
DGUNet$^+$ (Ours)	**33.17**	**0.963**	**31.40**	**0.944**

The visual comparison is shown in Fig. 5. Clearly, the results of our DGUNet and DGUNet$^+$ have higher visual quality, especially in recovering complex textures.

4.4. Image Denoising Results

For this application, we train our DGUNet on the commonly used SIDD dataset [2], which contains 320 degraded-clean image pairs corrupted by realistic noise with unknown distribution and noise levels. We evaluate each method on SIDD and DND [52] test sets. We compare our DGUNet with several recent methods [10, 53, 76, 77, 79] and report the evaluation results (PSNR and SSIM) in Tab. 3. One

Figure 5. Visual comparison of image deblurring. Our method produces sharper results that are visually closer to the ground truth.

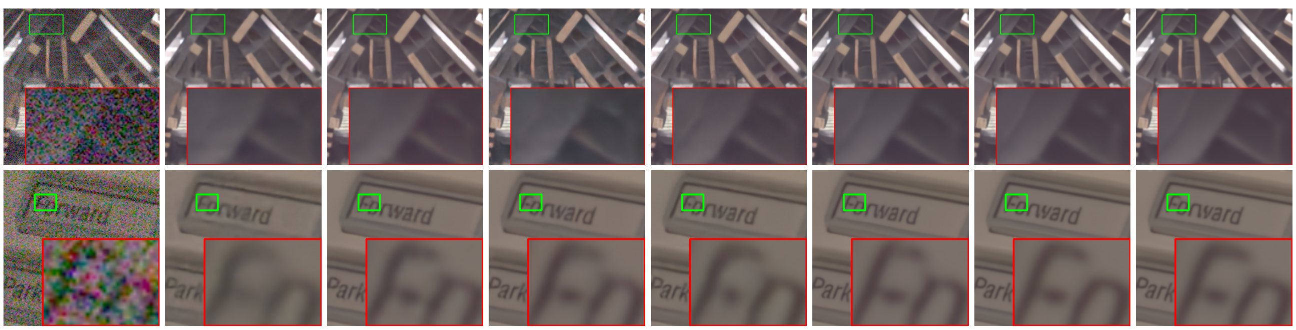

Figure 6. Visual comparison of real image denoising. Our method can remove unknown noise better while retaining satisfying details.

Table 3. Image denoising results on SIDD [2] and DND [52]. The best and second-best scores are **highlighted** and <u>underlined</u>.

Method	SIDD [2]		DND [52]	
	PSNR↑	SSIM↑	PSNR↑	SSIM↑
DnCNN [91]	23.66	0.583	32.43	0.790
MLP [8]	24.71	0.641	34.23	0.833
BM3D [12]	25.65	0.685	34.51	0.851
CBDNet [24]	30.78	0.801	38.06	0.942
RIDNet [3]	38.71	0.951	39.26	0.953
AINDNet [33]	38.95	0.952	39.37	0.951
VDN [75]	39.28	0.956	39.38	0.952
SADNet [10]	39.46	0.957	39.59	0.952
DANet+ [76]	39.47	0.957	39.58	0.955
MIRNet [78]	39.72	0.959	39.88	0.956
CycleISP [77]	39.52	0.957	39.56	0.956
DeamNet [53]	39.43	0.956	39.70	0.953
MPRNet [79]	39.71	0.958	39.80	0.954
DGUNet (Ours)	<u>39.88</u>	<u>0.959</u>	<u>40.04</u>	<u>0.956</u>
DGUNet$^+$ (Ours)	**39.91**	**0.960**	**40.12**	**0.957**

can see that our method achieves the best performance on both SIDD and DND test sets. Specifically, our DGUNet$^+$ outperforms MPRNet with 0.32 dB and 0.20 dB on DND and SIDD test sets, respectively. The visual comparison is presented in Fig. 6, including two samples from DND (the first row) and SIDD (the second row) test sets. Clearly, our method has good robustness to both high-intensity and low-intensity noise to recover the actual texture and structures, *e.g.*, the pattern of wood and the edge of letters.

4.5. Compressive Sensing Results

For this application, we choose the widely used BSD400 dataset [43] as the training data and evaluate each method on Set11 [35] and BSD68 [43] test sets. Same as [41,86,97], for a given set of CS ratios $\{1\%, 4\%, 10\%, 25\%, 50\%\}$, we jointly optimize the sampling matrix with the whole network. Note that in the task of compressive sensing, the degradation matrix $\mathbf{A}$ is exactly known, *i.e.*, the sampling matrix $\mathbf{\Phi}$. Thus, we directly use $\mathbf{\Phi}$ to calculate the gradient. The quantitative comparison is presented in Tab. 4. One can see that our DGUNet and DGUNet$^+$ have obvious advantages over either classic methods [59, 83] and recent top-performing methods [41, 86, 97], and the margin becomes more obvious at low CS ratios. For instance, there are 2 dB gains compared with OPINENet$^+$ [86] on the Set11 test set, with the CS ratio being 1%. An interesting finding is that DGUNet performs better than DGUNet$^+$ in some cases. This is mainly because the training set is small and the large model can not be fully optimized. The visual comparison is presented in Fig. 7, showing that our method can recover more details and sharper edges than other methods.

Table 4. Quantitative results of image compressive sensing. The best and second-best scores are **highlighted** and underlined.

Dataset	Ratio	ISTANet$^+$ [83]	CSNet [59]	AdapRecon [41]	OPINENet$^+$ [86]	AMPNet [97]	DGUNet	DGUNet$^+$
Set11	1%	17.42/0.4029	19.87/0.4977	19.63/0.4848	20.15/0.5340	20.04/0.5132	22.09/0.6096	**22.15/0.6114**
	4%	21.32/0.6037	23.93/0.7338	23.87/0.7279	25.69/0.7920	24.64/0.7527	**26.84/0.8249**	26.83/0.8230
	10%	26.64/0.8087	27.59/0.8575	27.39/0.8521	29.81/0.8884	28.84/0.8765	**31.07/0.9123**	30.93/0.9088
	25%	32.59/0.9254	31.70/0.9274	31.75/0.9257	34.86/0.9509	34.42/0.9513	36.11/0.9611	**36.18/0.9616**
	50%	38.11/0.9707	37.19/0.9700	35.87/0.9625	40.17/0.9797	40.12/0.9818	41.22/0.9836	**41.24/0.9837**
BSD68	1%	19.14/0.4158	21.91/0.4958	21.50/0.4825	22.11/0.5140	21.97/0.5086	22.65/0.5396	**22.70/0.5406**
	4%	22.17/0.5486	24.63/0.6564	24.30/0.6491	25.00/0.6825	25.40/0.6985	**25.55/0.7008**	25.45/0.6987
	10%	25.32/0.7022	27.02/0.7864	26.72/0.7821	27.82/0.8045	27.41/0.8036	**28.26/0.8193**	28.14/0.8165
	25%	29.36/0.8525	30.22/0.8918	30.10/0.8901	31.51/0.9061	31.56/0.9121	31.90/0.9155	**31.98/0.9158**
	50%	34.04/0.9424	34.82/0.9590	33.60/0.9479	36.35/0.9660	36.64/0.9707	37.01/0.9714	**37.04/0.9718**

PSNR	23.51 dB	23.06 dB	24.41 dB	24.73 dB	24.56 dB	26.13 dB	26.41 dB
Ground Truth	ISTANet$^+$ [83]	AdapRecon [41]	CSNet [59]	OPINENet$^+$ [86]	AMPNet [97]	DGUNet	DGUNet$^+$

Figure 7. Visual comparison of compressive sensing with the CS ratio being 10%. Our method can produce results with higher quality.

Table 5. Ablation study of the number of stages and number of feature fusion scales in our method on Rain100H test set.

Stages	9	7			5	3
Scales	3	3	2	1	3	3
PSNR	**31.27**	31.06	30.87	30.68	30.43	29.67

Table 6. Ablation study of different components in our method.

Mode	DGUNet$^+$	w/o FGDM	w/o SP	w/o ISFF
PSNR	**31.06**	30.51	30.92	30.09

4.6. Ablation Study

We present the ablation study in Tab. 5 and Tab. 6 to investigate the number of stages, number of feature fusion scales, and different components of our method. Experiments are conducted on image deraining and evaluated on Rain100H [71]. In order to highlight the performance changes, all modifications are made on our DGUNet$^+$.

Number of stages. In this part, we explore the gains brought by the number of stages, including 9, 7, 5, and 3 stages. From Tab. 5, we can find that the performance increases with the number of stages, demonstrating the effectiveness of the iterative network design. By making a trade-off between performance and computational complexity, we employ seven stages in our DGUNet and DGUNet$^+$.

Inter-stage feature fusion. As mentioned previously, to rectify the weakness of transforming a multi-channel feature map back to an image at the end of each stage, we introduce an inter-stage feature fusion module (ISFF). To demonstrate the effectiveness of ISFF, we remove it from our DGUNet$^+$, represented as "w/o ISFF" in Tab. 6, and we study the performance gains of multi-scale feature fusion in Tab. 5. We can find that our ISFF has obvious gains, and the performance increases with the number of feature fusion scales. Additionally, we study the effectiveness of spatial-adaptive fusion by replacing it with the direct addition, represented as "w/o SP" in Tab. 6. The performance degradation demonstrates the positive effect of this design.

Gradient descent module. Although our proposed DGUNet has good interpretability, the effect of such design also needs to be discussed carefully. In the experiment, we remove the Flexible Gradient Descent Module (FGDM) from our DGUNet$^+$, leading to an UNet-cascading structure. This variant is represented as "w/o FGDM" in Tab. 6. Compared with DGUNet$^+$, there are $0.55\ dB$ degradations on the Rain100H. Thus, the result demonstrates that our interpretable design also has performance gains.

5. Conclusion and Discussion

In this paper, we propose a deep generalized unfolding network (DGUNet) for IR. We develop principles that aim to combine the merits of model-based methods and deep learning methods. To this end, we unfold the PGD optimization algorithm into a deep network and integrate a gradient estimation strategy into the gradient descent step, enabling it to be easily applied to complex and real-world applications. To compensate for the intrinsic information loss in DUN, we design inter-stage feature pathways that work with multiple scales and spatial-adaptive normalization. Extensive experiments on numerous IR tasks (including **twelve** synthetic and real-world test sets) demonstrate the superiority of our method in terms of state-of-the-art performance, interpretability, and generalizability. Our future work will support DGUNet on MindSpore [1] platform.

References

[1] Mindspore. https://www.mindspore.cn/, 2020. 8

[2] Abdelrahman Abdelhamed, Stephen Lin, and Michael S Brown. A high-quality denoising dataset for smartphone cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1692–1700, 2018. 6, 7

[3] Saeed Anwar and Nick Barnes. Real image denoising with feature attention. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 3155–3164, 2019. 7

[4] Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009. 2, 3

[5] Stephen Boyd, Neal Parikh, and Eric Chu. *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers, 2011. 2, 3

[6] Tim Brooks, Ben Mildenhall, Tianfan Xue, Jiawen Chen, Dillon Sharlet, and Jonathan T Barron. Unprocessing images for learned raw denoising. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11036–11045, 2019. 2

[7] Antoni Buades, Bartomeu Coll, and J-M Morel. A non-local algorithm for image denoising. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 60–65, 2005. 1, 2

[8] Harold C Burger, Christian J Schuler, and Stefan Harmeling. Image denoising: Can plain neural networks compete with bm3d? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2392–2399, 2012. 7

[9] Stanley H Chan, Xiran Wang, and Omar A Elgendy. Plug-and-play admm for image restoration: Fixed-point convergence and applications. *IEEE Transactions on Computational Imaging*, 3(1):84–98, 2016. 4

[10] Meng Chang, Qi Li, Huajun Feng, and Zhihai Xu. Spatial-adaptive network for single image denoising. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 171–187, 2020. 6, 7

[11] Shen Cheng, Yuzhi Wang, Haibin Huang, Donghao Liu, Haoqiang Fan, and Shuaicheng Liu. Nbnet: Noise basis learning for image denoising with subspace projection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4896–4906, 2021. 4

[12] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on Image Processing*, 16(8):2080–2095, 2007. 1, 2, 7

[13] Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang. Compression artifacts reduction by a deep convolutional network. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 576–584, 2015. 2

[14] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2):295–307, 2015. 2

[15] Weisheng Dong, Xin Li, Lei Zhang, and Guangming Shi. Sparsity-based image denoising via dictionary learning and structural clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 457–464, 2011. 2

[16] Weisheng Dong, Peiyao Wang, Wotao Yin, Guangming Shi, Fangfang Wu, and Xiaotong Lu. Denoising prior driven deep neural network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(10):2305–2318, 2018. 2, 3

[17] Weisheng Dong, Lei Zhang, Guangming Shi, and Xiaolin Wu. Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization. *IEEE Transactions on Image Processing*, 20(7):1838–1857, 2011. 1

[18] Michael Elad and Michal Aharon. Image denoising via sparse and redundant representations over learned dictionaries. *IEEE Transactions on Image Processing*, 15(12):3736–3745, 2006. 2

[19] Xueyang Fu, Jiabin Huang, Xinghao Ding, Yinghao Liao, and John Paisley. Clearing the skies: A deep network architecture for single-image rain removal. *IEEE Transactions on Image Processing*, 26(6):2944–2956, 2017. 5, 6

[20] Xueyang Fu, Jiabin Huang, Delu Zeng, Yue Huang, Xinghao Ding, and John Paisley. Removing rain from single images via a deep detail network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3855–3863, 2017. 5, 6

[21] Hongyun Gao, Xin Tao, Xiaoyong Shen, and Jiaya Jia. Dynamic scene deblurring with parameter selective sharing and nested skip connections. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3848–3856, 2019. 6, 7

[22] Dong Gong, Jie Yang, Lingqiao Liu, Yanning Zhang, Ian Reid, Chunhua Shen, Anton Van Den Hengel, and Qinfeng Shi. From motion blur to motion flow: A deep learning solution for removing heterogeneous motion blur. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2319–2328, 2017. 6

[23] Shuhang Gu, Lei Zhang, Wangmeng Zuo, and Xiangchu Feng. Weighted nuclear norm minimization with application to image denoising. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2862–2869, 2014. 2

[24] Shi Guo, Zifei Yan, Kai Zhang, Wangmeng Zuo, and Lei Zhang. Toward convolutional blind denoising of real photographs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1712–1722, 2019. 2, 7

[25] Kaiming He, Jian Sun, and Xiaoou Tang. Single image haze removal using dark channel prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(12):2341–2353, 2010. 1

[26] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceed-*

ings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 770–778, 2016. 5

[27] Ran He, Wei-Shi Zheng, Tieniu Tan, and Zhenan Sun. Half-quadratic-based iterative minimization for robust sparse representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(2):261–275, 2013. 2, 3

[28] Felix Heide, Markus Steinberger, Yun-Ta Tsai, Mushfiqur Rouf, Dawid Pajak, Dikpal Reddy, Orazio Gallo, Jing Liu, Wolfgang Heidrich, Karen Egiazarian, et al. Flexisp: A flexible camera image processing framework. *ACM Transactions on Graphics (ToG)*, 33(6):1–13, 2014. 2

[29] Tae Hyun Kim, Byeongjoo Ahn, and Kyoung Mu Lee. Dynamic scene deblurring. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 3160–3167, 2013. 6

[30] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 448–456, 2015. 2, 4, 5

[31] Kui Jiang, Zhongyuan Wang, Peng Yi, Chen Chen, Baojin Huang, Yimin Luo, Jiayi Ma, and Junjun Jiang. Multi-scale progressive fusion network for single image deraining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8346–8355, 2020. 5, 6

[32] Kwang In Kim and Younghee Kwon. Single-image super-resolution using sparse regression and natural image prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(6):1127–1133, 2010. 1

[33] Yoonsik Kim, Jae Woong Soh, Gu Yong Park, and Nam Ik Cho. Transfer learning from synthetic to real-noise denoising with adaptive instance normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3482–3492, 2020. 1, 2, 7

[34] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference of Learning Representations (ICLR)*, 2015. 5

[35] Kuldeep Kulkarni, Suhas Lohit, Pavan Turaga, Ronan Kerviche, and Amit Ashok. Reconnet: Non-iterative reconstruction of images from compressively sensed measurements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 449–458, 2016. 7

[36] Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jiří Matas. Deblurgan: Blind motion deblurring using conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8183–8192, 2018. 6

[37] Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang. Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 8878–8887, 2019. 6

[38] Xia Li, Jianlong Wu, Zhouchen Lin, Hong Liu, and Hongbin Zha. Recurrent squeeze-and-excitation context aggregation net for single image deraining. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 254–269, 2018. 5, 6

[39] Yu Li, Robby T Tan, Xiaojie Guo, Jiangbo Lu, and Michael S Brown. Rain streak removal using layer priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2736–2744, 2016. 6

[40] Ding Liu, Bihan Wen, Yuchen Fan, Chen Change Loy, and Thomas S Huang. Non-local recurrent network for image restoration. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, pages 1673–1682, 2018. 2

[41] Suhas Lohit, Kuldeep Kulkarni, Ronan Kerviche, Pavan Turaga, and Amit Ashok. Convolutional neural networks for noniterative reconstruction of compressively sensed images. *IEEE Transactions on Computational Imaging*, 4(3):326–340, 2018. 7, 8

[42] Julien Mairal, Michael Elad, and Guillermo Sapiro. Sparse representation for color image restoration. *IEEE Transactions on Image Processing*, 17(1):53–69, 2007. 2

[43] David Martin, Charless Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 416–423, 2001. 7

[44] Tim Meinhardt, Michael Moller, Caner Hazirbas, and Daniel Cremers. Learning proximal operators: Using denoising networks for regularizing inverse imaging problems. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 1781–1790, 2017. 2, 3

[45] Chong Mou, Jian Zhang, Xiaopeng Fan, Hangfan Liu, and Ronggang Wang. Cola-net: Collaborative attention network for image restoration. *IEEE Transactions on Multimedia*, 2021. 2

[46] Chong Mou, Jian Zhang, and Zhuoyuan Wu. Dynamic attentive graph learning for image restoration. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4328–4337, 2021. 2

[47] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3883–3891, 2017. 6

[48] Qian Ning, Weisheng Dong, Guangming Shi, Leida Li, and Xin Li. Accurate and lightweight image super-resolution with model-guided deep unfolding network. *IEEE Journal of Selected Topics in Signal Processing*, 15(2):240–252, 2020. 3

[49] Stanley Osher, Martin Burger, Donald Goldfarb, Jinjun Xu, and Wotao Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Modeling & Simulation*, 4(2):460–489, 2005. 2

[50] Dongwon Park, Dong Un Kang, Jisoo Kim, and Se Young Chun. Multi-temporal recurrent neural networks for progressive non-uniform single image deblurring with incremental temporal training. In *Proceedings of the European*

Conference on Computer Vision (ECCV)*, pages 327–343, 2020. 6, 7

[51] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2337–2346, 2019. 4, 5

[52] Tobias Plotz and Stefan Roth. Benchmarking denoising algorithms with real photographs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1586–1595, 2017. 1, 6, 7

[53] Chao Ren, Xiaohai He, Chuncheng Wang, and Zhibo Zhao. Adaptive consistency prior based deep network for image denoising. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8596–8606, 2021. 1, 2, 6, 7

[54] Dongwei Ren, Wangmeng Zuo, Qinghua Hu, Pengfei Zhu, and Deyu Meng. Progressive image deraining networks: A better and simpler baseline. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3937–3946, 2019. 5, 6

[55] Jaesung Rim, Haeyun Lee, Jucheol Won, and Sunghyun Cho. Real-world blur dataset for learning and benchmarking deblurring algorithms. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 184–201. Springer, 2020. 6

[56] Leonid I Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, 1992. 1

[57] Ernest Ryu, Jialin Liu, Sicheng Wang, Xiaohan Chen, Zhangyang Wang, and Wotao Yin. Plug-and-play methods provably converge with properly trained denoisers. In *International Conference on Machine Learning (ICML)*, pages 5546–5557, 2019. 3

[58] Ziyi Shen, Wenguan Wang, Xiankai Lu, Jianbing Shen, Haibin Ling, Tingfa Xu, and Ling Shao. Human-aware motion deblurring. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 5572–5581, 2019. 6

[59] Wuzhen Shi, Feng Jiang, Shengping Zhang, and Debin Zhao. Deep networks for compressed image sensing. In *Proceedings of the 2017 IEEE International Conference on Multimedia and Expo (ICME)*, pages 877–882, 2017. 7, 8

[60] Jiechong Song, Bin Chen, and Jian Zhang. Memory-augmented deep unfolding network for compressive sensing. In *Proceedings of the ACM International Conference on Multimedia*, pages 4249–4258, 2021. 3

[61] Maitreya Suin, Kuldeep Purohit, and AN Rajagopalan. Spatially-attentive patch-hierarchical network for adaptive motion deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3606–3615, 2020. 6

[62] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. MemNet: A persistent memory network for image restoration. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4539–4547, 2017. 2

[63] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8174–8182, 2018. 6

[64] Afonso M Teodoro, José M Bioucas-Dias, and Mário AT Figueiredo. Image restoration and reconstruction using variable splitting and class-adapted image priors. In *Proceedings of the IEEE International Conference on Image Processing (ICIP)*, pages 3518–3522, 2016. 2

[65] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6924–6932, 2017. 5

[66] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR)*, pages 606–615, 2018. 4

[67] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 5

[68] Wei Wei, Deyu Meng, Qian Zhao, Zongben Xu, and Ying Wu. Semi-supervised transfer learning for image rain removal. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3877–3886, 2019. 5, 6

[69] Oliver Whyte, Josef Sivic, Andrew Zisserman, and Jean Ponce. Non-uniform deblurring for shaken images. *International Journal of Computer Vision*, 98(2):168–186, 2012. 6

[70] Li Xu, Shicheng Zheng, and Jiaya Jia. Unnatural l0 sparse representation for natural image deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1107–1114, 2013. 6

[71] Wenhan Yang, Robby T Tan, Jiashi Feng, Jiaying Liu, Zongming Guo, and Shuicheng Yan. Deep joint rain detection and removal from a single image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1357–1366, 2017. 5, 6, 8

[72] Rajeev Yasarla and Vishal M Patel. Uncertainty guided multi-scale residual learning-using a cycle spinning cnn for single image de-raining. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8405–8414, 2019. 5, 6

[73] Di You, Jian Zhang, Jingfen Xie, Bin Chen, and Siwei Ma. COAST: Controllable arbitrary-sampling network for compressive sensing. *IEEE Transactions on Image Processing*, 30:6066–6080, 2021. 3

[74] Xin Yuan, Yang Liu, Jinli Suo, and Qionghai Dai. Plug-and-play algorithms for large-scale snapshot compressive imaging. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1447–1457, 2020. 2, 3, 4

[75] Zongsheng Yue, Hongwei Yong, Qian Zhao, Deyu Meng, and Lei Zhang. Variational denoising network: Toward blind noise modeling and removal. In *Proceedings of the Advances in Neural Information Processing Systems (NeurIPS)*, pages 1688–1699, 2019. 1, 2, 7

[76] Zongsheng Yue, Qian Zhao, Lei Zhang, and Deyu Meng. Dual adversarial network: Toward real-world noise removal and noise generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 41–58. Springer, 2020. 1, 6, 7

[77] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Cycleisp: Real image restoration via improved data synthesis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2696–2705, 2020. 6, 7

[78] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Learning enriched features for real image restoration and enhancement. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 492–511, 2020. 1, 7

[79] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14821–14831, 2021. 1, 2, 4, 5, 6, 7

[80] Hongguang Zhang, Yuchao Dai, Hongdong Li, and Piotr Koniusz. Deep stacked hierarchical multi-patch network for image deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5978–5986, 2019. 6

[81] He Zhang and Vishal M Patel. Density-aware single image de-raining using a multi-stream dense network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 695–704, 2018. 5, 6

[82] He Zhang, Vishwanath Sindagi, and Vishal M Patel. Image de-raining using a conditional generative adversarial network. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(11):3943–3956, 2019. 5, 6

[83] Jian Zhang and Bernard Ghanem. Ista-net: Interpretable optimization-inspired deep network for image compressive sensing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1828–1837, 2018. 2, 3, 7, 8

[84] Jiawei Zhang, Jinshan Pan, Jimmy Ren, Yibing Song, Linchao Bao, Rynson WH Lau, and Ming-Hsuan Yang. Dynamic scene deblurring using spatially variant recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2521–2529, 2018. 6

[85] Jian Zhang, Ruiqin Xiong, Chen Zhao, Yongbing Zhang, Siwei Ma, and Wen Gao. Concolor: Constrained non-convex low-rank model for image deblocking. *IEEE Transactions on Image Processing*, 25(3):1246–1259, 2016. 2

[86] Jian Zhang, Chen Zhao, and Wen Gao. Optimization-inspired compact deep compressive sensing. *IEEE Journal of Selected Topics in Signal Processing*, 14(4):765–774, 2020. 7, 8

[87] Jian Zhang, Debin Zhao, and Wen Gao. Group-based sparse representation for image restoration. *IEEE Transactions on Image Processing*, 23(8):3336–3351, 2014. 2

[88] Kai Zhang, Luc Van Gool, and Radu Timofte. Deep unfolding network for image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3217–3226, 2020. 2, 3

[89] Kai Zhang, Yawei Li, Wangmeng Zuo, Lei Zhang, Luc Van Gool, and Radu Timofte. Plug-and-play image restoration with deep denoiser prior. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021. 2

[90] Kaihao Zhang, Wenhan Luo, Yiran Zhong, Lin Ma, Bjorn Stenger, Wei Liu, and Hongdong Li. Deblurring by realistic blurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2737–2746, 2020. 6, 7

[91] Kai Zhang, Wangmeng Zuo, Yunjin Chen, Deyu Meng, and Lei Zhang. Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising. *IEEE Transactions on Image Processing*, 26(7):3142–3155, 2017. 2, 7

[92] Kai Zhang, Wangmeng Zuo, Shuhang Gu, and Lei 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. 2, 3

[93] Kai Zhang, Wangmeng Zuo, and Lei Zhang. FFDNet: toward a fast and flexible solution for CNN-based image denoising. *IEEE Transactions on Image Processing*, 27(9):4608–4622, 2018. 2

[94] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Deep plug-and-play super-resolution for arbitrary blur kernels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1671–1681, 2019. 2, 3

[95] Yulun Zhang, , Kunpeng Li, Bineng Zhong, and Yun Fu. Residual non-local attention networks for image restoration. In *Proceedings of the International Conference on Learning Representations (ICLR)*, pages 1–18, 2019. 2

[96] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun 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. 2

[97] Zhonghao Zhang, Yipeng Liu, Jiani Liu, Fei Wen, and Ce Zhu. Amp-net: Denoising-based deep unfolding for compressive image sensing. *IEEE Transactions on Image Processing*, 30:1487–1500, 2020. 7, 8

[98] Chen Zhao, Siwei Ma, Jian Zhang, Ruiqin Xiong, and Wen Gao. Video compressive sensing reconstruction via reweighted residual sparsity. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(6):1182–1195, 2016. 2

[99] Chen Zhao, Jian Zhang, Siwei Ma, Xiaopeng Fan, Yongbing Zhang, and Wen Gao. Reducing image compression artifacts by structural sparse representation and quantization constraint prior. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(10):2057–2071, 2016. 2

[100] Daniel Zoran and Yair Weiss. From learning models of natural image patches to whole image restoration. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 479–486, 2011. 2