EDPN: Enhanced Deep Pyramid Network for Blurry Image Restoration

Ruikang Xu* Zeyu Xiao* Jie Huang Yueyi Zhang Zhiwei Xiong†
University of Science and Technology of China

Abstract

*Image deblurring has seen a great improvement with the development of deep neural networks. In practice, however, blurry images often suffer from additional degradations such as downscaling and compression. To address these challenges, we propose an **E**nhanced **D**eep **P**yramid **N**etwork (EDPN) for blurry image restoration from multiple degradations, by fully exploiting the self- and cross-scale similarities in the degraded image. Specifically, we design two pyramid-based modules, i.e., the pyramid progressive transfer (PPT) module and the pyramid self-attention (PSA) module, as the main components of the proposed network. By taking several replicated blurry images as inputs, the PPT module transfers both self- and cross-scale similarity information from the same degraded image in a progressive manner. Then, the PSA module fuses the above transferred features for subsequent restoration using self- and spatial-attention mechanisms. Experimental results demonstrate that our method significantly outperforms existing solutions for blurry image super-resolution and blurry image deblocking. In the NTIRE 2021 Image Deblurring Challenge, EDPN achieves the best PSNR/SSIM/LPIPS scores in Track 1 (Low Resolution) and the best SSIM/LPIPS scores in Track 2 (JPEG Artifacts). The implementation code is available at* https://github.com/zeyuxiao1997/EDPN.

1. Introduction

Image deblurring has long been an important task in computer vision and image processing. Blurry images may be caused by camera shake [45, 38], object motion [70, 40, 56] or out-of-focus [1, 30, 2], and the goal of image deblurring is to recover a sharp latent image with necessary edge structures and details. Image deblurring is a highly ill-posed task especially due to the difficulties in estimating the spatially varying blur kernel with limited information from a single observation.

Early Bayesian-based iterative deblurring methods include the Wiener filter [55] and the Richardson-Lucy algorithm [41]. Later works commonly rely on developing effective image priors [26, 42, 61, 77] or sophisticated data terms [12]. More recently, convolutional neural networks (CNNs) have been exploited for image deblurring and produce promising results. For example, Nah *et al.* [35] propose a multi-scale loss function to implement a coarse-to-fine processing pipeline. Tao *et al.* [48] and Gao *et al.* [17] improve this work by using shared network parameters at different scales, achieving state-of-the-art performance.

Despite of the encouraging performance achieved by CNN-based methods for image deblurring, they fail to reconstruct sharp results from the blurry images with multiple degradations. In practice, however, the blurry images often suffer from additional degradations. For example, to save the storage and transmission bandwidth, the raw images are generally downscaled and/or compressed, resulting in low resolution and/or compression artifacts when the images are received by terminal users. Therefore, a more general blurry image restoration task should not only consider single blur degradation, but also cover more complex degradations, *e.g.*, blurry image super-resolution (BISR) and blurry image deblocking (BID).

To address the general blurry image restoration task (*i.e.*, BISR and BID), a straightforward strategy is to cascade deblurring and super-resolution/deblocking techniques, or vice versa. However, there are several issues with such approaches. First, a simple concatenation of two models is a sub-optimal solution due to error accumulation, *i.e.*, the estimated error of the first model will be propagated and magnified in the second model. Second, the two-stage network does not fully exploit the correlation between the two tasks. Third, the cascading approach is specifically designed for fixed tasks, which cannot be easily deployed in more general scenarios. Several recent methods [39, 63, 67, 72] jointly solve the image deblurring and super-resolution problems using end-to-end deep neural networks. However, these methods either focus on domain-specific applications, *e.g.*, face and text [63, 67] images, or address the uniform Gaussian blur only [73].

In this paper, we propose an **E**nhanced **D**eep **P**yramid **N**etwork (EDPN), which is extensible to various blurry image restoration tasks, including BISR and BID. The inputs

*These authors contribute equally to this work.
†Correspondence should be addressed to zwxiong@ustc.edu.cn

of EDPN are several replicated blurry images, which aims to fully exploit the self-similarity contained in the degraded image. The cores of EDPN are (1) an information transfer module named as pyramid progressive transfer (PPT) module, and (2) a feature fusion module named as pyramid self-attention (PSA) module. Specifically, the PPT module is designed to transfer the cross-scale similarity information from the same degraded image at the feature level with a pyramid structure, which performs the deformable convolution and generates attention masks to transfer the self-similarity information in a progressive manner. The PSA module is designed to aggregate information across the transferred features with a pyramid structure, which adopts self- and spatial-attention mechanisms to weight the multiple features.

Our contributions can be summarized as follows:

1. We propose a blurry image restoration network named EDPN, which can generate sharp results from blurry images with multiple degradations.

2. We design two core components, *i.e.*, the PPT module and the PSA module, for fully exploiting the self- and cross-scale similarities of the same degraded image.

3. Our method significantly outperforms existing solutions for blurry image super-resolution and blurry image deblocking. In the NTIRE 2021 Image Deblurring Challenge, EDPN achieves the best PSNR/SSIM/LPIPS scores in Track 1 (Low Resolution) and the best SSIM/LPIPS scores in Track 2 (JPEG Artifacts).

2. Related Work

Image deblurring. Image deblurring is a highly ill-posed problem which aims at generating a sharp image from a blurry observation. Various natural images and kernel priors have been developed to regularize the solution space of the latent sharp image, including heavy-tailed gradient prior [43], sparse kernel prior [14], l_0 gradient prior [62], normalized sparsity prior [27] and dark channels [37]. Recently, several CNN-based methods have been proposed for image deblurring. For example, Sun *et al.* [46] propose a CNN-based model to estimate a kernel and remove non-uniform motion blur. Chakrabarti [4] uses a network to compute estimations of sharp images that are blurred by an unknown motion kernel. Nah *et al.* [35] propose a multi-scale loss function to apply a coarse-to-fine strategy. Kupyn *et al.* propose DeblurGAN [28] and DeblurGAN-v2 [29] to remove blur based on adversarial learning.

Image super-resolution. A plenty of works have been proposed to solve image super-resolution (SR), including interpolation-based [71], model-based [18] and learning-based methods [59, 60, 11, 25, 33, 75, 6]. Traditional methods are usually limited in representing the complex local-image structures, while recently developed deep CNNs have shown great advantages in image structure representation and consequently boost the SR performance [11, 8, 33]. For example, Kim *et al.* [25] employ the residual learning strategy to design the VDSR model with 20 convolution layers. By introducing channel attention mechanism, Zhang *et al.* [75] propose RCAN which improves the SR performance a lot. Dai *et al.* [8] propose a second-order attention network for more powerful feature expression and feature correlation learning.

Image deblocking. Traditional JPEG artifacts removal methods pay attention to filter design. For example, Foi *et al.* [15] propose the shape-adaptive DCT-based filter for image denoising and de-blocking. The method in [5] utilizes sparse coding to restore compressed images. Others treat JPEG artifacts removal as an ill-posed inverse problem and solve it by using regression trees [23] and non-local self-similarity property [32]. CNN-based methods learn to minimize the reconstruction error with respect to ground truth reference images, and operate in the pixel domain [16, 10, 47, 52], the DCT domain [66], or both domains [20, 19, 74]. For example, Fan *et al.* [13] propose a decoupled learning framework to combine different parameterized operators. Fu *et al.* [16] introduce a more compact and explainable deep sparse coding architecture to generate high-quality deblocking results.

General blurry image restoration. A typical blurry image restoration task is to super-resolve a low-resolution (LR) image and deblur a blurry image jointly [72, 63, 73, 69, 65]. This joint problem is more challenging than the individual problems. Xu *et al.* [63] train a generative adversarial network to super-resolve blurry face and text images. Zhang *et al.* [73] propose a deep encoder-decoder network for joint image deblurring and super-resolution. However, they focus on LR images degraded by the uniform Gaussian blur. Zhang *et al.* [72] propose a dual-branch network to extract features for deblurring and super-resolution and learn a gate module to adaptively fuse the features for image restoration.

Attention mechanism. The attention mechanism in deep learning, which mimics the human visual attention mechanism, is originally developed in a non-local manner. For example, the matrix multiplication in self-attention draws global dependencies of each word in a sentence [51] or each pixel in an image [54]. The squeeze-and-excitation network squeezes global spatial information into a channel descriptor to capture channel-wise dependencies [22]. To alleviate the problems arising from scale variation and small objects, Dai *et al.* [9] propose the multi-scale channel attention module for aggregating contextual information from different receptive fields, which can simultaneously aggregate local and global feature contexts inside the channel at-

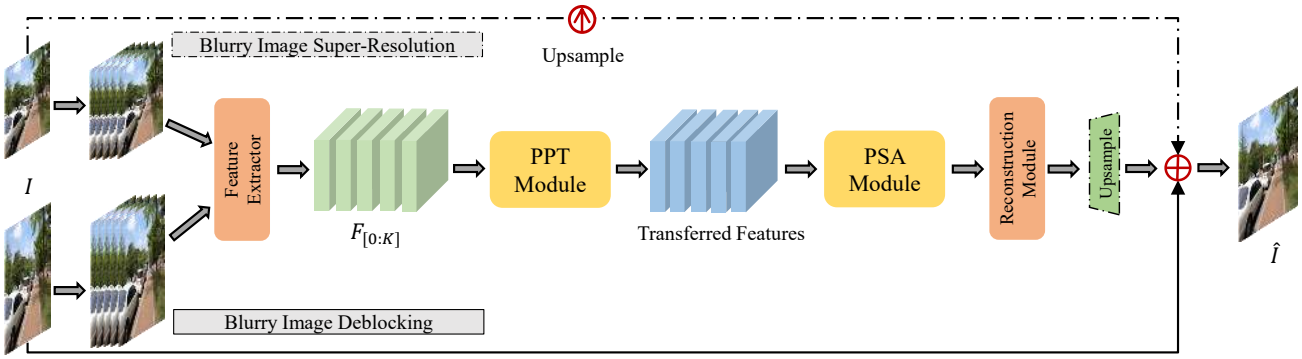

Figure 1. The EDPN network is suitable for various image deblurring tasks. Here, we focus on tasks of blurry image super-resolution and blurry image deblocking for example. The given blurry image is first replicated for several times, and then sent to the following components. We use five replicated images as an illustrative example.

tention mechanism.

3. Network Architecture

3.1. Overview

Given a blurry image I, our method aims to reconstruct a high-quality image $\hat{I}$, which should be close to the ground truth I^{GT}. As shown in Figure 1, our EDPN mainly consists of four parts: the feature extractor, the PPT module, the PSA module and the reconstruction module.

Take the BISR task as an example, we first replicate the given blurry image I for K times as the inputs of our EDPN, which can better exploit the self-similarity in the degraded image. Then, we extract the features from the inputs by the feature extractor. The feature extractor consists of 18 residual blocks. The extracted features are denoted as $F_{[0:K]}$, which will be utilized for subsequent operations.

After feature extraction, the features are fed into the PPT module to transfer the self- and cross-scale similarity information in a progressive manner. Then, the PSA module fuses the similarity information of the transferred features $\hat{F}_{[0:K]}$ and conducts feature aggregation. The details of these two modules are described in Section 3.2 and Section 3.3. After that, the fused features are fed into the reconstruction module with an upsampling operation. Finally, the restored image is obtained by adding the predicted image residual to a directly upsampled image. The reconstruction module is composed of 120 multi-scale residual channel-attention blocks [9]. For tasks with high spatial resolution inputs such as BID, the upsampling layer at the end is not necessary.

3.2. Pyramid Progressive Transfer Module

The detailed structure of the proposed PPT module is shown in Figure 2. The inputs of the PPT module are the features $F_{[0:K]}$. Given $K+1$ features, the PPT module also needs to be executed for $K+1$ times.

Inside the PPT module, we adopt the pyramid and progressive structure to learn the self- and cross-scale similarities. For the pyramid structure, strided convolution layers are utilized to downscale the features from the upper level by a factor of 2 for obtaining features at the current level. Assuming that the number of pyramid levels constructed in the PPT module is M, for each level, there are N progressive transfer blocks (PTBs) to extract the self-similarity progressively. Then at the m-th level, the inputs of the n-th PTB are the first feature F_0^m and the output of the previous block $(F_{i,PTB}^{n-1})^m (i \in [0, K])$. It should be noted that the inputs of the first PTB are F_0^m and F_i^m. Inspired by TDAN [50] and EDVR [53], we apply the deformable convolution [7] in the PTB. This process can be denoted as

$$(F_i^D)^{m,n} = \mathcal{F}_{Dconv}(F_0^m, (F_{i,PTB}^{n-1})^m)), \tag{1}$$

where $\mathcal{F}_{Dconv}(\cdot)$ stands for the deformable convolution and $(F_i^D)^{m,n}$ stands for the output of the deformable convolution of the n-th block at the m-th level. The learned offsets of the deformable convolution are predicted from the inputs, which is formulated as

$$(\Delta P_i)^{m,n} = \mathcal{F}_C(F_0^m \parallel (F_{i,PTB}^{n-1})^m), \tag{2}$$

where $(\Delta P_i)^{m,n}$ stands for the learned offset of the n-th block at the m-th level, $\parallel$ stands for the channel-wise concatenation and $\mathcal{F}_C(\cdot)$ stands for the convolution operation.

Then, we generate the feature-level mask of the n-th block at the m-th level $(Mask_i)^{m,n}$, which forces the PTB to focus on the most correlated information of features. Mathematically, the mask is calculated as

$$(Mask_i)^{m,n} = Softmax(\mathcal{F}_C(F_0^m) - \mathcal{F}_C((F_{i,PTB}^{n-1})^m)). \tag{3}$$

The motion attention mask is further multiplied with the output of the deformable convolution.

After a convolution layer, the generated feature is treated as the residual information of this block. The output feature

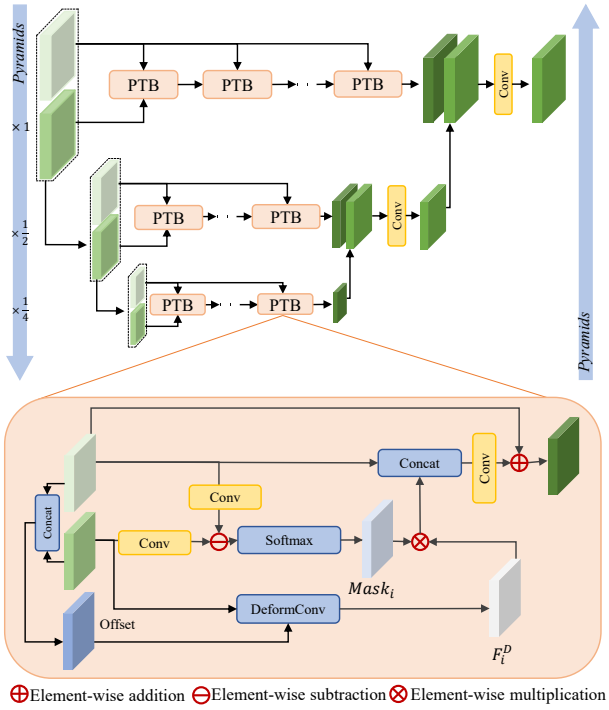

Figure 2. Structure of the PPT module.

of the n-th PTB at the m-th level is obtained by adding the residual information to the first feature, which can be formulated as

$$(F^n_{i,PTB})^m = F^m_0 + \mathcal{F}_C(F^m_0 \,\|\, (Mask_i)^{m,n} \otimes (F^D_i)^{m,n}), \tag{4}$$

where $\otimes$ represents the element-wise multiplication. Finally, the output feature of the PPT module at the m-th level $(F^{PPT}_i)^m$ can be depicted as

$$(F^{PPT}_i)^m = \mathcal{F}_C(Up((F^{PPT}_i)^{m+1})^{\uparrow s} \,\|\, (F^N_{i,PTB})^m), \tag{5}$$

where $(F^N_{i,PTB})^m$ represents the feature generated after N PTBs at the m-th level. $Up(\cdot)^{\uparrow s}$ refers to upscaling by a factor of s, which is implemented by bilinear interpolation. We construct our PPT module with 3-level pyramid structure, *i.e.*, $M = 3$. The PPT module can transfer self- and cross-scale similarities in such a progressive and coarse-to-fine manner. We demonstrate the effectiveness of the PPT module and analyze the relation between the performance and the number of PTBs in Section 4.2.

3.3. Pyramid Self-Attention Module

After the PPT module, the features with self- and cross-scale similarities for fusion and reconstruction have been extracted and transferred. Inspired by [53], we propose the PSA module to assign pixel-level aggregation weights with a pyramid structure. In addition, we adopt the 3D convolution operation to fuse the information of all features effectively as shown in Figure 3.

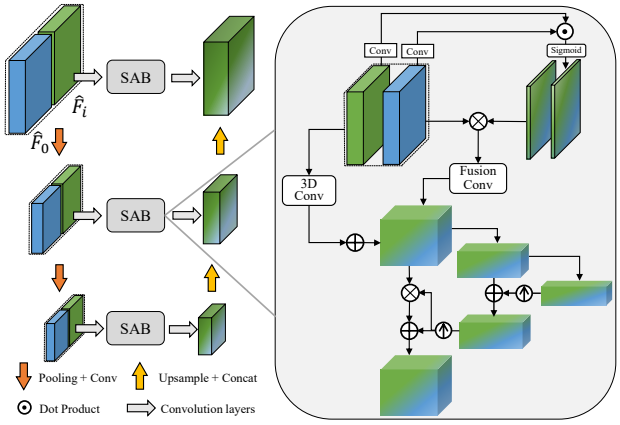

Figure 3. Structure of the PSA module.

Our PSA module also adopts the pyramid processing. At first, we define the output feature of the self-attention block (SAB) at the l-th level as F^l_{sa}. Then, we use strided convolution layers to downscale the features at the l-th pyramid level by a factor of 2, obtaining L-level pyramid of feature representation. At the l-th level, we generate the attention maps to compute similarity in an embedding space. For each feature, the similarity map can be calculated as

$$\Theta^l_i = Sigmoid(\mathcal{F}_C(\hat{F}^l_0)^T \odot \mathcal{F}_C(\hat{F}^l_i)), \tag{6}$$

where $\odot$ stands for the dot product operation. $\hat{F}^l_i$ stands for the transferred features at the l-th level. Specially, similarity maps are spatial-specific for each spatial location, *i.e.*, the spatial size of Θ^l_i is the same as that of $\hat{F}^l_i$. The similarity maps are then multiplied in a pixel-wise manner to the original transferred features, and an extra fusion convolution layer is adopted to aggregate these attention-modulated features $\tilde{F}^l_i$, denoted as

$$\begin{aligned} \tilde{F}^l_i &= \Theta^l_i \otimes \hat{F}^l_i, \\ F^l_{fusion} &= \mathcal{F}_C(\tilde{F}^l_{[0:K]}). \end{aligned} \tag{7}$$

Then, we add the original transferred features after the 3D convolution operation to the fused features. Meanwhile, the spatial attention masks are then computed from the fused features with the pyramid structure. Following [53], the fused features are modulated by the masks through element-wise multiplication and addition to generate the output F^l_{sa} at the l-th level, denoted as

$$F^l_{sa} = \mathcal{F}_C(F^l_{sa} \,\|\, Up(F^{l+1}_{sa})^{\uparrow 2}). \tag{8}$$

Here, we use a 3-level pyramid ($L = 3$). To reduce computational cost, we do not increase channel numbers as spatial sizes decrease. The PSA module in such a coarse-to-fine

manner improves the effectiveness of information aggregation and we demonstrate the effectiveness of the PSA module in Section 4.2.

4. Experiments

4.1. Experimental Settings

Dataset. The experiments are conducted strictly following the instructions of the NTIRE 2021 Image Deblurring Challenge [36]. There are two tracks in this challenge. Track 1 requires to restore and upscale a blurry image by a factor of 4. Track 2 requires to restore a blurry image with JPEG artifacts. The dataset used for these tracks is REDS [34], a real-world high-quality video dataset originally collected for video super-resolution [3, 44, 24, 50, 53, 58, 57] and video deblurring [45, 64], which consists of 240 scenes for training, and 30 scenes for validation and testing.

Loss functions. To optimize EDPN, we adopt the Charbonnier loss [53] defined as

$$\mathcal{L}_{Charb} = \sqrt{\left\| I^{GT} - \hat{I} \right\|^2 + \varepsilon^2}, \tag{9}$$

where ε is set to 1×10^{-3}. We also adpot the SSIM loss [21] defined as

$$\mathcal{L}_{SSIM} = 1 - SSIM(I^{GT}, \hat{I}). \tag{10}$$

The complete loss function for training EDPN is

$$\mathcal{L} = \mathcal{L}_{Charb} + \lambda \mathcal{L}_{SSIM}, \tag{11}$$

where λ is the weighting factor.

Training settings. The channel size in the feature extractor and the reconstruction module is set to 64. We use RGB patches of size 64×64 and 160×160 as inputs for BISR and BID tasks, respectively. The network takes five replicated images (*i.e.*, $K = 4$) as inputs. We augment the training data with random horizontal flips and rotations.

Implementation details. We utilize the Adam optimizer with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The training procedure follows the mini-batch strategy and the batch size is 4. The learning rate is initially set to 1×10^{-4} and is later down-scaled by a factor of 0.8 after every 100,000 iterations till 1,000,000 iterations. λ is set to 0.1. All the networks in the experiments are implemented using PyTorch 1.1 and trained with NVIDIA GeForce GTX1080Ti GPUs.

4.2. Ablation Study

In this subsection, we investigate the necessity of our proposed modules, the suitable design choices (*i.e.*, the number of the PTBs and input images) and ensemble schemes. All the ablation experiments are conducted for the BISR task.

Table 1. Ablation on the PPT and PSA modules.

PPT	PSA	RGB Channel		Y Channel	
		PSNR	SSIM	PSNR	SSIM
✗	✗	27.43	0.7839	28.80	0.8045
✓	✗	27.84	0.8010	29.23	0.8208
✗	✓	27.78	0.7997	29.18	0.8196
✓	✓	**28.01**	**0.8091**	**29.39**	**0.8282**

Table 2. Ablation on the number of PTBs.

Num. of PTB	RGB Channel		Y Channel	
	PSNR	SSIM	PSNR	SSIM
$N = 1$	27.91	0.8070	29.31	0.8264
$N = 2$	27.92	0.8071	29.32	0.8264
$N = 3$	28.01	0.8091	29.39	0.8282
$N = 4$	28.01	0.8097	29.40	0.8288
$N = 5$	**28.02**	**0.8102**	**29.41**	**0.8296**

Table 3. Ablation on the number of input images.

Num. of Input	RGB Channel		Y Channel	
	PSNR	SSIM	PSNR	SSIM
# 1 ($K = 0$)	27.89	0.8012	29.21	0.8244
# 3 ($K = 2$)	27.95	**0.8102**	29.35	**0.8292**
# 5 ($K = 4$)	**28.01**	0.8091	**29.39**	0.8282
# 7 ($K = 6$)	27.91	0.8069	29.31	0.8263

Table 4. Ablation on the loss functions.

$\mathcal{L}_{Charb}$	$\mathcal{L}_{SSIM}$	RGB Channel		Y Channel	
		PSNR	SSIM	PSNR	SSIM
✓	✗	**28.04**	0.8070	**29.44**	0.8247
✓	✓	28.01	**0.8091**	29.39	**0.8282**

Table 5. Ablation on the ensemble schemes.

Ensemble scheme	RGB Channel		Y Channel	
	PSNR	SSIM	PSNR	SSIM
Original Model	28.01	0.8091	29.39	0.8282
+Self-ensemble	28.16	0.8172	29.53	0.8433
+Model-ensemble	**28.32**	**0.8197**	**29.71**	**0.8452**

Network architecture. We perform an ablation experiment to demonstrate the effectiveness of the PPT and PSA modules. First, we construct a basic model for comparison. In the basic model, we remove the pyramid structure and substitute our proposed PTBs by cascading several residual blocks in the PPT module. The PSA module is replaced by a cascade of several residual blocks and convolution layers in the basic model. Then we recover our design by adding the PPT and PSA modules step by step. The comparison results are shown in Table 1. When the PPT module is added, the PSNR value in RGB channel is improved from 27.43 dB to 27.84 dB. When the PSA module is added, the PSNR value in RGB channel is improved from 27.43 dB to 27.78 dB. It demonstrates that the PPT and PSA modules are proved to be highly effective for the blurry image restoration tasks. After adding these two modules, the PSNR value is further

Table 6. Quantitative comparisons between EDPN and existing methods on the REDS validation set. **Top**: 4× BISR; **Bottom**: BID. **Red** and blue indicate the best and the second best performance, respectively.

Task	Method	PSNR↑	SSIM↑	LILPS↓	#Param (M)	Running time (s)
BISR	MSRN	26.65	0.7576	0.1147	5.80	0.0427
	GFN	26.91	0.7647	0.1139	12.21	0.0479
	RCAN	27.15	0.7740	0.1093	14.87	0.0982
	EDVR	27.63	0.8033	0.1002	12.38	0.1426
	EDPN	**28.01**	**0.8091**	**0.0819**	13.34	0.2224
BID	RNAN	26.73	0.7672	0.1071	8.54	0.1052
	MPRNet	27.52	0.7896	0.1002	19.19	0.0652
	SRN	27.71	0.7950	0.0984	9.76	0.0507
	EDVR	28.19	0.8156	0.0882	12.38	0.1553
	EDPN	**28.96**	**0.8203**	**0.0767**	13.34	0.2224

Table 7. Challenge results on the test set of REDS. **Left**: Track 1 (Low Resolution); **Right**: Track 2 (JPEG Artifacts). **Red** and blue indicate the best and the second best performance, respectively.

Track 1	PSNR ↑	SSIM ↑	LPIPS ↓
EDPN	**29.04**	**0.8416**	**0.2397**
2nd	28.91	0.8246	0.2569
3rd	28.51	0.8172	0.2547
4th	28.44	0.8158	0.2531
5th	28.44	0.8135	0.2704
6th	28.42	0.8132	0.2685
7th	28.36	0.8130	0.2666
8th	28.33	0.8132	0.2606
9th	28.28	0.8110	0.2651
10th	28.25	0.8108	0.2636

Track 2	PSNR ↑	SSIM ↑	LPIPS ↓
1st	**29.70**	0.8403	0.2319
2nd	29.62	0.8397	0.2304
3rd	29.60	0.8398	0.2302
4th	29.59	0.8381	0.2340
5th	29.56	0.8385	0.2322
6th	29.34	0.8355	0.2546
EDPN	29.33	**0.8565**	**0.2222**
8th	29.17	0.8325	0.2411
9th	29.11	0.8292	0.2449
10th	29.07	0.8286	0.2499

improved to 28.01 dB. Similar phenomenon also appears on other metrics. This ablation experiment demonstrates that our proposed modules are effective on the general blurring image restoration task.

PTB number. To determine the number of PTBs used at each level of the PPT module, we compare the quantitative performance under different settings with $N = 1, 2, 3, 4, 5$. The comparison results are shown in Table 2. It can be observed that better performance can be achieved with the increase of the number N, which indicates that using more PTBs is more effective in extracting the self-similarity. To balance the computing efficiency and the performance, we deploy 3 PTBs at each level of the PPT module.

Input image number. We train our EDPN with different numbers of replicated images ($K = 0, 2, 4, 6$) as inputs and compare the performance in Table 3. It can be observed that using 4 replicated images as inputs generates the best performance in terms of PSNR. It is worth noting that, the PSNR value in RGB channel is improved from 27.89 dB to 28.01 dB when the number of input images increases from 1 to 5. In the experiments, we always adopt 4 times replication for the inputs.

Loss function. We investigate the contribution of different loss terms by adjusting the weighting factors in Equation 11, and the results are shown in Table 4. Since $\mathcal{L}_{Charb}$ is optimized at the pixel level, the best result can be achieved in terms of PSNR. When $\mathcal{L}_{SSIM}$ is adopted together with $\mathcal{L}_{Charb}$, we can obtain a higher SSIM value. In the experiments, we train EDPN with both $\mathcal{L}_{Charb}$ and $\mathcal{L}_{SSIM}$ for a better tradeoff between PSNR and SSIM.

Ensemble scheme. We adopt two types of ensemble schemes to further enhance the performance of EDPN. The first one is self-ensemble. We rotate the input image 90°, 180° and 270°, and feed them into the network to obtain corresponding outputs. Then we average these outputs and the original output as the final result. The second one is model-ensemble, whose result is the linear combination of several models with different training iterations and loss functions. Experimental results listed in Table 5 demonstrate the performance improvement achieved using these two ensemble schemes.

4.3. Comparisons with Existing Methods

To validate the effectiveness of the proposed method, we compare our EDPN with the existing methods that can be directly applied to the BISR and BID tasks. For BISR, we compare with RCAN [75], MSRN [31], GFN [72] and EDVR [53]. For BID, we compare with SRN [49], RNAN [76], MPRNet [68] and EDVR [53]. All these methods are trained using the whole training set of the challenge, and the official validation set is adopted for evaluation. For EDVR, it takes the same number of input images as EDPN. The quantitative results are listed in Table 6. As can be seen, our EDPN significantly outperforms previous methods in PSNR, SSIM and LPIPS metrics. Compared with EDVR, our EDPN achieves 0.38 dB and 0.77 dB gain in PSNR for BISR and BID, respectively. The comparison results demonstrate that our EDPN can effectively exploit

MSRN GFN RCAN EDVR **EDPN (Ours)** Ground truth

Figure 4. Qualitative comparisons on the validation set of REDS for BISR. Please zoom in for better visualization.

RNAN MPRNet SRN EDVR **EDPN (Ours)** Ground Truth

Figure 5. Qualitative comparisons on the validation set of REDS for BID. Please zoom in for better visualization.

self- and cross-scale similarities and boost the performance of blurry image restoration from multiple degradations. In addition, we calculate the number of parameters and the average running time of different methods when the size of input images is 128×128 using a 1080Ti GPU as shown in Table 6.

To evaluate the perceptual quality, we show two examples of restored results in Figure 4 and Figure 5 for the BISR and BID tasks on the REDS validation set, respectively. Two more examples on the REDS test set are given in Figure 6 and Figure 7. It can be seen that our EDPN provides better qualitative results than other methods with more accurate details in both tasks. Specifically, the edge regions of the restored images from EDPN are notably shaper and clearer while other methods are only able to address small blur.

4.4. Challenge Results

In the NTIRE 2021 Image Deblurring Challenge, EDPN achieves the best PSNR/SSIM/LPIPS scores in Track 1 (Low Resolution) and the best SSIM/LPIPS scores in Track 2 (JPEG Artifacts). We list the results from the top 10 teams on the final test set in Table 7. Compared with the second best method, our EDPN achieves 0.13 dB increase in PSNR, 0.017 increase in SSIM and 0.0172 decrease in LILPS in Track 1. In Track 2, our EDPN achieves 0.0162 increase in SSIM and 0.008 decrease in LILPS. According to the challenge report [36], our EDPN is among the most efficient solutions in both tracks.

5. Conclusion

In this paper, we propose a blurry image restoration network named EDPN, which is designed to address multiple degradations, *e.g.*, blurry image super-resolution and blurry image deblocking. The proposed two core modules of EDPN, the PPT module and the PSA module, are proved to be highly effective for the above two tasks. We believe EDPN also has potential to advance other image/video restoration tasks, especially those with multiple degradations.

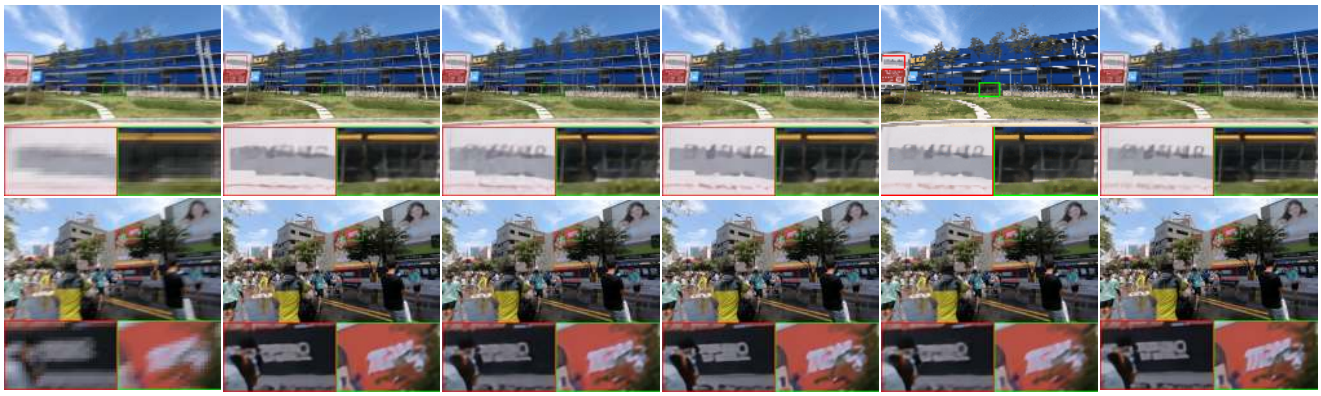

Figure 6. Qualitative comparisons on the test set of REDS for Track 1 (Low Resolution). Please zoom in for better visualization.

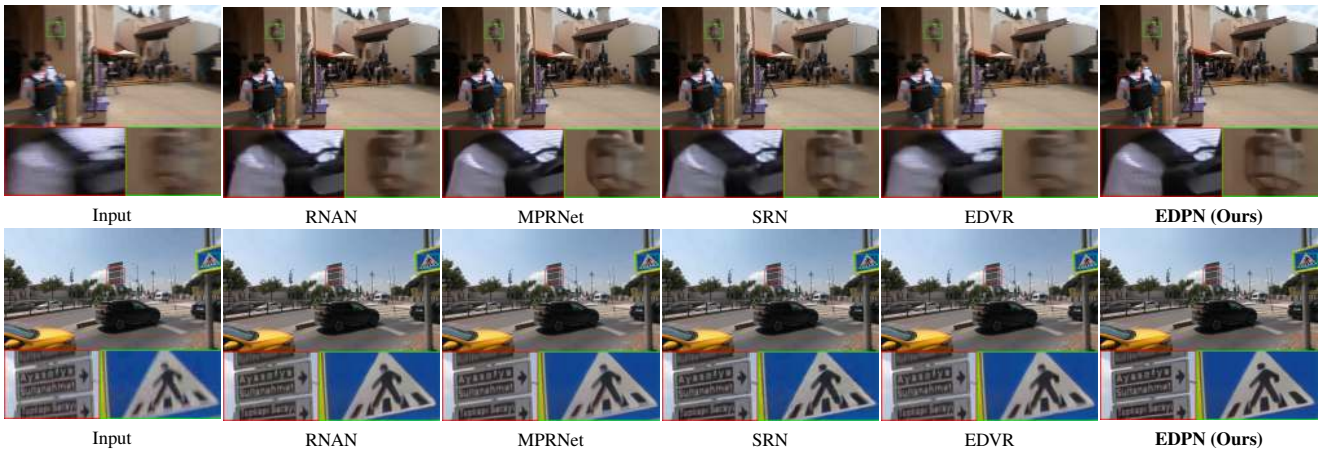

Figure 7. Qualitative comparisons on the test set of REDS for Track 2 (JPEG Artifacts). Please zoom in for better visualization.

Acknowledgement

We acknowledge funding from National Key R&D Program of China under Grant 2017YFA0700800, and National Natural Science Foundation of China under Grant 61901435.

References

[1] Abdullah Abuolaim and Michael S Brown. Defocus deblurring using dual-pixel data. In *ECCV*, 2020. 1

[2] Abdullah Abuolaim, Michael S Brown, and Radu Timofte. Ntire 2021 challenge for defocus deblurring using dual-pixel images: Methods and results. In *CVPRW*, 2021. 1

[3] Jose Caballero, Christian Ledig, Andrew Aitken, Alejandro Acosta, Johannes Totz, Zehan Wang, and Wenzhe Shi. Real-time video super-resolution with spatio-temporal networks and motion compensation. In *CVPR*, 2017. 5

[4] Ayan Chakrabarti. A neural approach to blind motion deblurring. In *ECCV*, 2016. 2

[5] Huibin Chang, Michael K Ng, and Tieyong Zeng. Reducing artifacts in jpeg decompression via a learned dictionary. *IEEE Transactions on Image Processing*, 62(3):718–728, 2013. 2

[6] Chang Chen, Zhiwei Xiong, Xinmei Tian, Zheng-Jun Zha, and Feng Wu. Camera lens super-resolution. In *CVPR*, 2019. 2

[7] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *ICCV*, 2017. 3

[8] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *CVPR*, 2019. 2

[9] Yimian Dai, Fabian Gieseke, Stefan Oehmcke, Yiquan Wu, and Kobus Barnard. Attentional feature fusion. In *WACV*, 2021. 2, 3

[10] Chao Dong, Yubin Deng, Chen Change Loy, and Xiaoou Tang. Compression artifacts reduction by a deep convolutional network. In *ICCV*, 2015. 2

[11] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *ECCV*, 2014. 2

[12] Jiangxin Dong, Jinshan Pan, Deqing Sun, Zhixun Su, and Ming-Hsuan Yang. Learning data terms for non-blind de-

blurring. In *ECCV*, 2018. 1

[13] Qingnan Fan, Dongdong Chen, Lu Yuan, Gang Hua, Nenghai Yu, and Baoquan Chen. Decouple learning for parameterized image operators. In *ECCV*, 2018. 2

[14] Rob Fergus, Barun Singh, Aaron Hertzmann, Sam T Roweis, and William T Freeman. Removing camera shake from a single photograph. In *SIGGRAPH*. 2006. 2

[15] Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Pointwise shape-adaptive dct for high-quality denoising and deblocking of grayscale and color images. *IEEE Transactions on Image Processing*, 16(5):1395–1411, 2007. 2

[16] Xueyang Fu, Zheng-Jun Zha, Feng Wu, Xinghao Ding, and John Paisley. Jpeg artifacts reduction via deep convolutional sparse coding. In *ICCV*, 2019. 2

[17] Hongyun Gao, Xin Tao, Xiaoyong Shen, and Jiaya Jia. Dynamic scene deblurring with parameter selective sharing and nested skip connections. In *CVPR*, 2019. 1

[18] Shuhang Gu, Wangmeng Zuo, Qi Xie, Deyu Meng, Xiangchu Feng, and Lei Zhang. Convolutional sparse coding for image super-resolution. In *ICCV*, 2015. 2

[19] Jun Guo and Hongyang Chao. Building dual-domain representations for compression artifacts reduction. In *ECCV*, 2016. 2

[20] Jun Guo and Hongyang Chao. One-to-many network for visually pleasing compression artifacts reduction. In *CVPR*, 2017. 2

[21] Alain Hore and Djemel Ziou. Image quality metrics: Psnr vs. ssim. In *ICPR*, 2010. 5

[22] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *CVPR*, 2018. 2

[23] Jeremy Jancsary, Sebastian Nowozin, and Carsten Rother. Loss-specific training of non-parametric image restoration models: A new state of the art. In *ECCV*, 2012. 2

[24] Armin Kappeler, Seunghwan Yoo, Qiqin Dai, and Aggelos K Katsaggelos. Video super-resolution with convolutional neural networks. *IEEE Transactions on Computational Imaging*, 2(2):109–122, 2016. 5

[25] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *CVPR*, 2016. 2

[26] Dilip Krishnan and Rob Fergus. Fast image deconvolution using hyper-laplacian priors. In *NeurIPS*, 2009. 1

[27] Dilip Krishnan, Terence Tay, and Rob Fergus. Blind deconvolution using a normalized sparsity measure. In *CVPR*, 2011. 2

[28] Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jiří Matas. Deblurgan: Blind motion deblurring using conditional adversarial networks. In *CVPR*, 2018. 2

[29] Orest Kupyn, Tetiana Martyniuk, Junru Wu, and Zhangyang Wang. Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In *ICCV*, 2019. 2

[30] Junyong Lee, Sungkil Lee, Sunghyun Cho, and Seungyong Lee. Deep defocus map estimation using domain adaptation. In *CVPR*, 2019. 1

[31] Juncheng Li, Faming Fang, Kangfu Mei, and Guixu Zhang. Multi-scale residual network for image super-resolution. In *ECCV*, 2018. 6

[32] Tao Li, Xiaohai He, Linbo Qing, Qizhi Teng, and Honggang Chen. An iterative framework of cascaded deblocking and superresolution for compressed images. *IEEE Transactions on Multimedia*, 20(6):1305–1320, 2017. 2

[33] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *CVPRW*, 2017. 2

[34] Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee. Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In *CVPRW*, 2019. 5

[35] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, 2017. 1, 2

[36] Seungjun Nah, Sanghyun Son, Suyoung Lee, Radu Timofte, and Kyoung Mu Lee. Ntire 2021 challenge on image deblurring. In *CVPRW*, 2021. 5, 7

[37] Jinshan Pan, Deqing Sun, Hanspeter Pfister, and Ming-Hsuan Yang. Blind image deblurring using dark channel prior. In *CVPR*, 2016. 2

[38] Liyuan Pan, Yuchao Dai, and Miaomiao Liu. Single image deblurring and camera motion estimation with depth map. In *WACV*, 2019. 1

[39] Yuhui Quan, Jieting Yang, Yixin Chen, Yong Xu, and Hui Ji. Collaborative deep learning for super-resolving blurry text images. *IEEE Transactions on Computational Imaging*, 6:778–790, 2020. 1

[40] Sainandan Ramakrishnan, Shubham Pachori, Aalok Gangopadhyay, and Shanmuganathan Raman. Deep generative filter for motion deblurring. In *ICCVW*, 2017. 1

[41] William Hadley Richardson. Bayesian-based iterative method of image restoration. *JoSA*, 62(1):55–59, 1972. 1

[42] Uwe Schmidt and Stefan Roth. Shrinkage fields for effective image restoration. In *CVPR*, 2014. 1

[43] Qi Shan, Jiaya Jia, and Aseem Agarwala. High-quality motion deblurring from a single image. *Acm Transactions on Graphics (TOG)*, 27(3):1–10, 2008. 2

[44] Sanghyun Son, Suyoung Lee, Seungjun Nah, Radu Timofte, and Kyoung Mu Lee. Ntire 2021 challenge on video super-resolution. In *CVPRW*, 2021. 5

[45] Shuochen Su, Mauricio Delbracio, Jue Wang, Guillermo Sapiro, Wolfgang Heidrich, and Oliver Wang. Deep video deblurring for hand-held cameras. In *CVPR*, 2017. 1, 5

[46] Jian Sun, Wenfei Cao, Zongben Xu, and Jean Ponce. Learning a convolutional neural network for non-uniform motion blur removal. In *CVPR*, 2015. 2

[47] Pavel Svoboda, Michal Hradis, David Barina, and Pavel Zemcik. Compression artifacts removal using convolutional neural networks. In *WSCG*, 2016. 2

[48] Xin Tao, Hongyun Gao, Renjie Liao, Jue Wang, and Jiaya Jia. Detail-revealing deep video super-resolution. In *ICCV*, 2017. 1

[49] Xin Tao, Hongyun Gao, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *CVPR*, 2018. 6

[50] Yapeng Tian, Yulun Zhang, Yun Fu, and Chenliang Xu. Tdan: Temporally-deformable alignment network for video super-resolution. In *CVPR*, 2020. 3, 5

[51] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2

[52] Menglu Wang, Xueyang Fu, Zepei Sun, and Zheng-Jun Zha. Jpeg artifacts removal via compression quality ranker-guided networks. In *IJCAI*, 2020. 2

[53] Xintao Wang, Kelvin C.K. Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *CVPRW*, 2019. 3, 4, 5, 6

[54] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018. 2

[55] N Weiner and Interpolation Extrapolation. Smoothing of stationary time series: With engineering applications, 1949. 1

[56] Patrick Wieschollek, Michael Hirsch, Bernhard Scholkopf, and Hendrik Lensch. Learning blind motion deblurring. In *ICCV*, 2017. 1

[57] Zeyu Xiao, Xueyang Fu, Jie Huang, Zhen Cheng, and Zhiwei Xiong. Space-time distillation for video super-resolution. In *CVPR*, 2021. 5

[58] Zeyu Xiao, Zhiwei Xiong, Xueyang Fu, Dong Liu, and Zheng-Jun Zha. Space-time video super-resolution using temporal profiles. In *ACM MM*, 2020. 5

[59] Zhiwei Xiong, Xiaoyan Sun, and Feng Wu. Image hallucination with feature enhancement. In *CVPR*, 2009. 2

[60] Zhiwei Xiong, Dong Xu, Xiaoyan Sun, and Feng Wu. Example-based super-resolution with soft information and decision. *IEEE Transactions on Multimedia*, 15(6):1458–1465, 2013. 2

[61] Li Xu and Jiaya Jia. Two-phase kernel estimation for robust motion deblurring. In *ECCV*, 2010. 1

[62] Li Xu, Shicheng Zheng, and Jiaya Jia. Unnatural l0 sparse representation for natural image deblurring. In *CVPR*, 2013. 2

[63] Xiangyu Xu, Deqing Sun, Jinshan Pan, Yujin Zhang, Hanspeter Pfister, and Ming-Hsuan Yang. Learning to super-resolve blurry face and text images. In *ICCV*, 2017. 1, 2

[64] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision*, 127(8):1106–1125, 2019. 5

[65] Chao-Hsun Yang and Long-Wen Chang. Deblurring and super-resolution using deep gated fusion attention networks for face images. In *ICASSP*, 2020. 2

[66] Jaeyoung Yoo, Sang-ho Lee, and Nojun Kwak. Image restoration by estimating frequency distribution of local patches. In *CVPR*, 2018. 2

[67] Xin Yu, Basura Fernando, Richard Hartley, and Fatih Porikli. Super-resolving very low-resolution face images with supplementary attributes. In *CVPR*, 2018. 1

[68] Syed Waqas Zamir, Aditya Arora, Salman Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *CVPR*, 2021. 6

[69] Dongyang Zhang, Zhenwen Liang, and Jie Shao. Joint image deblurring and super-resolution with attention dual supervised network. *Neurocomputing*, 412:187–196, 2020. 2

[70] 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 *CVPR*, 2018. 1

[71] Lei Zhang and Xiaolin Wu. An edge-guided image interpolation algorithm via directional filtering and data fusion. *IEEE Transactions on Image Processing*, 15(8):2226–2238, 2006. 2

[72] Xinyi Zhang, Hang Dong, Zhe Hu, Wei-Sheng Lai, Fei Wang, and Ming-Hsuan Yang. Gated fusion network for joint image deblurring and super-resolution. In *BMVC*, 2018. 1, 2, 6

[73] Xinyi Zhang, Fei Wang, Hang Dong, and Yu Guo. A deep encoder-decoder networks for joint deblurring and super-resolution. In *ICASSP*, 2018. 1, 2

[74] Xiaoshuai Zhang, Wenhan Yang, Yueyu Hu, and Jiaying Liu. Dmcnn: Dual-domain multi-scale convolutional neural network for compression artifacts removal. In *ICIP*, 2018. 2

[75] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *ECCV*, 2018. 2, 6

[76] Yulun Zhang, Kunpeng Li, Kai Li, Bineng Zhong, and Yun Fu. Residual non-local attention networks for image restoration. In *ICLR*, 2019. 6

[77] Daniel Zoran and Yair Weiss. From learning models of natural image patches to whole image restoration. In *ICCV*, 2011. 1