Learning the Non-differentiable Optimization for Blind Super-Resolution

Zheng Hui[1] Jie Li[1] Xiumei Wang[1] Xinbo Gao[1,2,*]
[1] Visual Information Processing Lab, Xidian University, China
[2] The Chongqing Key Laboratory of Image Cognition,
Chongqing University of Posts and Telecommunications, China
zheng_hui@aliyun.com, leejie@mail.xidian.edu.cn, wangxm@xidian.edu.cn, xbgao@mail.xidian.edu.cn

Abstract

Previous convolutional neural network (CNN) based blind super-resolution (SR) methods usually adopt an iterative optimization way to approximate the ground-truth (GT) step-by-step. This solution always involves more computational costs to bring about time-consuming inference. At present, most blind SR algorithms are dedicated to obtaining high-fidelity results; their loss function generally employs L1 loss. To further improve the visual quality of SR results, perceptual metric, such as NIQE, is necessary to guide the network optimization. However, due to the non-differentiable property of NIQE, it cannot be as the loss function. Towards these issues, we propose an adaptive modulation network (AMNet) for multiple degradations SR, which is composed of the pivotal adaptive modulation layer (AMLayer). It is an efficient yet lightweight fusion layer between blur kernel and image features. Equipped with the blur kernel predictor, we naturally upgrade the AMNet to the blind SR model. Instead of considering iterative strategy, we make the blur kernel predictor trainable in the whole blind SR model, in which AMNet is well-trained. Also, we fit deep reinforcement learning into the blind SR model (AMNet-RL) to tackle the non-differentiable optimization problem. Specifically, the blur kernel predictor will be the actor to estimate the blur kernel from the input low-resolution (LR) image. The reward is designed by the pre-defined differentiable or non-differentiable metric. Extensive experiments show that our model can outperform state-of-the-art methods in both fidelity and perceptual metrics.

1. Introduction

Single image super-resolution (SISR) refers to estimating the plausible and sharp detailed high-resolution (HR) image from its counterpart low-resolution (LR) image. It has been widely used in image/video enhancement, remote sensing imaging, and video surveillance. Recently, the introduction of convolutional neural networks (CNNs) makes the SISR performance reach a new height. Numerous CNN-based SISR methods [6, 7, 8, 16, 18, 13, 39, 19, 27] have explored network architecture designs and training strategies. They have focused on supervised settings with a fixed degradation model, *e.g*., *bicubic downsampling*. These algorithms achieved impressive results for the *bicubic downsampling* condition but produced undesirable artifacts when the images with a different degradation. Zhang *et al*. [35] proposed SRMD to handle multiple degradations via a single model to address the issue of multiple degradations. Different from previous CNN-based methods, SRMD explicitly takes both LR image and its degradation maps as input. Following SRMD, Xu *et al*. [30] proposed a single unified dynamic network trained for variational degradations (UDVD) to improve performance; its primary contribution is two types of dynamic convolutions. Note that the predefined blur kernel is given; thus, SRMD [35] and UDVD [30] are both non-blind settings. However, in most practical applications, blur kernels are not provided. Thus, the SR problem with unknown blur kernels, *i.e*., blind SR, is a more attractive field for academia and industry.

In general, to tackle the blind SR problem, previous techniques [35, 34] decompose the blind SR problem into two sequential subproblems, *i.e*., estimating blur kernel from input LR image and generating SR image based on estimated kernel. As stated in [22], this solution is not an end-to-end training approach, causing a suboptimal problem. Based on the observation of artifacts caused by kernel mismatch, Gu *et al*. [10] made efforts to correct an inaccurate blur kernel. They proposed an iterative kernel correction (IKC) method to correct the estimated kernel only by observing the previous SR results. In a deep alternating network (DAN) [22], the authors make the estimation of blur kernel much easier through sending both LR and SR images to *Estimator*. This iterative principle can make generated SR images gradually approach the ground-truth, but

*Corresponding author

it will consume more computational costs and make training/testing processing slower.

Besides, the current multiple degradations SR methods (including non-blind and blind settings) [35, 10, 25, 22] mainly adopt mean absolute error (MAE) or mean square error (MSE) as the loss function to achieve high PSNR values. It is rare to explore the multiple degradations perceptual SR problem. Under the condition of *bicubic downsampling*, many perceptual SR methods incorporate the perceptual loss [14] and adversarial learning [19] to generate realistic textures and exact details. Following the training strategy in ESRGAN [27], Zhang *et al*. [34] trained USRNet (complex degradations) with the MAE loss for PSNR performance and then fine-tuned the model with the weighted combination of MAE loss, VGG perceptual loss, and relativistic adversarial loss to pursue perceptual quality performance. The most challenging problem is the evaluation procedure, whether single degradation perceptual SR or multiple degradations perceptual SR. HR images (ground-truth) are not available in many applications. Thus, an objective metric like PSNR/SSIM and perceptual metric like LPIPS [36] cannot be used. At this time, some non-reference image quality assessment (NR-IQA) metrics can be utilized, such as NIQE [24].

Nevertheless, most of these NR-IQA metrics are not differentiable, which cannot serve as the loss functions to optimize the network. Zhang *et al*. [37] introduced a Ranker to learn the behavior of perceptual metrics. However, training this Ranker needs to make a rank dataset. Specifically, select two SR images and calculate their ranking order according to the perceptual metric's quality score. This method indirectly optimizes the network in the orientation of specific perceptual metrics. Therefore, there is also a lack of a solution that does not need to make a training dataset and explicitly optimize the non-differentiable objective function.

This paper is devoted to addressing the above issues, *i.e*., how to solve the non-differentiable evaluation metrics optimization for blind SR problems while maintaining fast training and testing speed (non-iterative). Following the standard approach, we model the LR image as degradation from the HR image with blurring and downsampling. First, given a blur kernel and a LR image, we need to train a single network for multiple degradations SR as in [35, 10, 30]. Motivated by style transfer [12] and image synthesis [15], we design the new generator architecture equipped with modified adaptive instance normalization (AdaIN) to control the image SR process. Our generator, namely adaptive modulation network (AMNet) for multiple degradations SR, adjusts the "blur/sharp style" of the image based on the embedding code of blur kernel. In this way, we can significantly reduce the tremendous amount of calculation caused by using the spatial feature transform (SFT) layers in SFTMD [10] without sacrificing performance. Second, we tune the input embedding code of blur kernel to optimize the output SR image towards the given non-differentiable metrics. To this end, a policy is adopted to select the blur kernel code to guide the optimization. Such a problem can be solved by a reinforcement learning (RL) framework where the agent models actions (blur kernel codes) from the observations (LR images). The reward is related to designative evaluation metrics (differentiable or non-differentiable). For implementing the high-speed training/inference, we use only single-step actions (inspired by [17]) in our whole blind SR framework – AMNet-RL (Adaptive Modulation Network with Reinforcement Learning). This paper makes the following contributions:

- We design a novel modified AdaIN module, which can be used in our proposed adaptive modulation network (AMNet) to better fulfill the multiple degradations SR problem while having the attributes of lower computational cost and higher speed than the previous multiple degradations SR methods [35, 10, 30]. To pursue the perceptual effect, we also construct a GAN-based version of AMNet, denoted as AMGAN.
- We introduce an efficient RL algorithm into our whole blind SR framework. It can optimize the policy to accomplish the blur kernel estimation task guided by the non-differentiable evaluation metrics. To the best of our knowledge, the proposed method is the first RL that optimizes blind SR with the in-differentiable perceptual metrics.
- We validate our AMNet-RL (PSNR-oriented), and AMGAN-RL (perception-driven) can achieve comparable results on commonly used datasets.

2. Related work

2.1. Non-blind Super-Resolution

Bicubic Interpolation Downsampling. In the past few years, various CNN-based SISR methods have focused on restoring HR images from LR images synthesized by the predefined downsampling setting (*bicubic interpolation*). Since the pioneering work SRCNN [6, 7], many CNN-based methods [16, 39, 13, 19, 27] have been proposed based on this downsampling setting. Note that SRGAN [19] and ESRGAN [27] pursue generating more realistic images. To optimize the SR model in the orientation of perceptual metrics, RankSRGAN [37] introduces a *Ranker* to learn the behavior of the perceptual metrics by learning to rank approach.

Multiple Degradations. Another kind of non-blind SR aims to present a single model for multiple degradations. These solutions send both the LR image and its corresponding blur kernel to the network. Zhang *et al*. [35] deal with multiple degradations via a single model under the maximum a posteriori (MAP) framework. They introduce a dimensionality stretching strategy to resolve the inputs' dimension inconsistency (LR image, blur kernel, noise level). This strategy has played a significant role in promoting the following SR methods for multiple degradations. Inspired by SFTGAN used in semantic super resolution [26], Gu *et al*. [10] propose a spatial feature transform (SFT) layer and insert it into each residual block. In this way, it can better keep the blur kernel information in a deeper network and provide better performance. Xu *et al*. [30] exploit dynamic convolutions to solve the SISR problem with variational degradations better. In likewise, Luo *et al*. [22] propose a conditional residual block (CRB), which concatenate the stretched kernel and LR image at the beginning of residual block in [38].

2.2. Blind Super-Resolution

For managing blind SR, the sequential combinations of a kernel estimation method and a non-blind SR method is a common scenario. Bell-Kligler *et al*. [2] find the underlying image-specific SR kernel of the input image through learning a downscaling generator that produces a downscaled version of the LR test image. It requires introducing a discriminator to judge whether the downscaled image has the same patch distribution as the original LR image. In [10], Gu *et al*. present a kernel predictor and a kernel corrector to predict the blur kernel iteratively. Luo *et al*. [22] alternate the Restorer (recovering SR image based on estimated blur kernel) and Estimator (utilizing LR and SR images to predict blur kernel) repeatedly to form an end-to-end trainable network.

2.3. Reinforcement Learning for Image Restoration

In recent years, some works have successfully applied deep RL to the image restoration field. Cao *et al*. [5] propose a deep RL-based attention mechanism to address the problem of face hallucination. Considering the contextual interdependency between patches, the authors use a recurrent policy network to specify a new attention region. As a result, it can learn a sequence of patches that need to be enhanced. Yu *et al*. [31] learn a policy to select appropriate tools from the toolbox to restore an image that is corrupted by mixed distortions progressively. For saving computing costs, Yu *et al*. also propose Path-Restore [32] that devises a pathfinder to select short paths for accessible regions. Most recently, Wei *et al*. [28] introduce RL into the PnP framework, yielding a tuning-free PnP proximal algorithm for compressed sensing MRI and phase retrieval.

3. Method

3.1. Problem Formulation

Following [10, 22], our objective is to solve the blind super-resolution problem mathematically formulated as:

$$\mathbf{I}^{LR}= \left(\mathbf{I}^{HR} \otimes \mathbf{k}\right) \downarrow_s, \tag{1}$$

where $\mathbf{I}^{HR}$, $\mathbf{I}^{LR}$, $\mathbf{k}$, $\otimes$, and $\downarrow_s$ indicate HR, LR image, blur kernel, convolution operation, bicubic downsampling operation with scale factor s, respectively. Previous methods solve this problem into two sequential steps:

$$\begin{cases} \mathbf{k} = \mathcal{P}\left(\mathbf{I}^{LR}; \Theta_{\mathcal{P}}\right) \\ \mathbf{I}^{SR} = \mathcal{G}\left(\mathbf{I}^{LR}, \mathbf{k}; \Theta_{\mathcal{G}}\right) \end{cases} \tag{2}$$

where $\mathcal{P}\left(\cdot\right)$ denotes the function that estimates $\mathbf{k}$ from $\mathbf{I}^{LR}$, $\mathcal{G}\left(\cdot\right)$ is a non-blind SR method that takes into account the degradation kernel $\mathbf{k}$ and LR image $\mathbf{I}^{LR}$, which allows the generator to be more flexible. $\Theta_{\mathcal{P}}$ and $\Theta_{\mathcal{G}}$ are the model parameters of $\mathcal{P}$ and $\mathcal{G}$, respectively. If training $\mathcal{P}\left(\cdot\right)$ and $\mathcal{G}\left(\cdot\right)$ respectively, it will usually lead to a significant decline in performance. This phenomenon has been pointed out in the recent blind SR literatures [10, 22]. Different from iterative way used in these recent works, we first train a non-blind SR network $\mathcal{G}$ and then optimize blur kernel estimation network $\mathcal{P}$ through

$$\arg\min_{\Theta_{\mathcal{P}}} \mathcal{L}\left(\mathbf{I}^{HR}, \mathcal{G}\left(\mathbf{I}^{LR}, \mathcal{P}\left(\mathbf{I}^{LR}; \Theta_{\mathcal{P}}\right); \Theta_{\mathcal{G}}\right)\right) \tag{3}$$

where $\Theta_{\mathcal{G}}$ is fixed. It means that we expect to obtain the $\Theta_{\mathcal{P}}$ that makes the SR result optimal. Here, $\mathcal{L}$ can choose fidelity related loss (*e.g*., L1 loss) or perception related loss (*e.g*., GAN loss) according to the task's requirement. In this way, we can build an end-to-end network while making blind SR training more comfortable and faster.

Moreover, we are committed to solving non-differentiable evaluation metrics-guided blind SR problems, building a bridge between the SR and quality assessment fields. Our goal is to choose a blur kernel code to make the resulting SR image $\mathbf{I}^{SR}$ better in the specified evaluation index. We formulate this task as an automatic parameter selection, which can be addressed via reinforcement learning (RL). In the RL framework, we need to define the tuple $(\mathcal{S}, \mathcal{A}, p, r)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, p is the transition function that maps input state $s \in \mathcal{S}$ to its outcome state $s' \in \mathcal{S}$ after taking action $a \in \mathcal{A}$, and r is the reward function. Specifically, in our task, $\mathcal{S}$ is the space of images, which includes the LR input image and the rendered SR image. $\mathcal{A}$ is the space of the blur kernel code. The transition function is the aforementioned non-blind SR model that renders the SR image based on input state s (LR image) and action a (reduced kernel), which can be expressed as $s' = p\left(s, a\right)$. RL's key element is the reward function r, which is constructed by predefined evaluation metrics, such as PSNR (differentiable) and NIQE (non-differentiable), and evaluates action given the state.

As shown in Figure 1, we propose an adaptive modulation network (AMNet) for our non-blind SR model. Following [35, 10, 22], the blur kernel is flattened and then reduced by principal component analysis (PCA). Equipped with kernel predictor, our AMNet can be upgraded to a blind SR model by using Equation 3 to optimize. Here, the kernel predictor plays the role of *Actor* in RL framework (please see in Figure 3(c)). Then, we will elaborate on the details in the following subsections.

3.2. Adaptive Modulation Network

Previous SR with multiple degradations algorithms [35, 10, 22, 30] stretch the input reduced kernel $\mathbf{k} \in \mathbb{R}^t$ into degradation maps $\mathbf{K} \in \mathbb{R}^{t \times H \times W}$, and then employ concatenation [35, 30, 22] or spatial feature transform [10] to integrate it with LR image or LR image features. Since the degradation maps will participate in the subsequent convolution operations, this approach will increase the computational costs. This phenomenon is most obvious in SFTMD [10]. Thus, we try to directly use reduced kernel $\mathbf{k}$ to control the characteristic of the network's output. Inspired by adopting the adaptive instance norm (AdaIN) [12] to implement the successful control of the image synthesis in StyleGAN [15], we propose a modified AdaIN to influence the output by the reduced kernel $\mathbf{k}$. We can regard the reduced kernel as a "blur/sharp" style code.

Given an input image features $x \in \mathbb{R}^{C \times H \times W}$ and a reduced kernel $\mathbf{k}$, AdaIN in our task can be defined as

$$y = \gamma\left(\mathbf{k}\right)\left(\frac{x - \mu\left(x\right)}{\sigma\left(x\right)}\right) + \beta\left(\mathbf{k}\right), \tag{4}$$

in which $\mu\left(x\right)$ and $\sigma\left(x\right)$ are the mean and standard deviation of the x across spatial dimensions independently for each channel:

$$\mu_c\left(x\right) = \frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W} x_{chw} \tag{5}$$

$$\sigma_c\left(x\right) = \sqrt{\frac{1}{HW}\sum_{h=1}^{H}\sum_{w=1}^{W}\left(x_{chw} - \mu_c\left(x\right)\right)^2 + \epsilon}. \tag{6}$$

We use the symbol $\gamma\left(\cdot\right)$ and $\beta\left(\cdot\right)$ to denote the functions that convert reduced kernel $\mathbf{k}$ to the scaling and bias values. However, only using $\gamma\left(\mathbf{k}\right)$ and $\beta\left(\mathbf{k}\right)$ to adjust the mean and standard deviation of the input image features cannot attain fine control, which manifests as a slow convergence speed and poor SR performance in our experiment. The reason why this approach shows breathtaking performance in StyleGAN [15] might be that each style code controls a specified level feature (related to feature resolution). In other words, the current style code knows the resolution information of the corresponding feature maps. Therefore, we construct the adaptive modulation layer (AMLayer), as illustrate in Figure 1(b), that modifies the mean and standard deviation of the current features using the guidance of the reduced kernel $\mathbf{k}$. Concretely, we can formulate this layer as follows:

$$y = \gamma\left(\mathbf{k}, \sigma\left(x\right)\right)\left(\frac{x - \mu\left(x\right)}{\sigma\left(x\right)}\right) + \beta\left(\mathbf{k}, \mu\left(x\right)\right), \tag{7}$$

where $\gamma\left(\mathbf{k}, \sigma\left(x\right)\right) = FC_1\left(Concat\left(FC\left(\mathbf{k}\right), \sigma\left(x\right)\right)\right)$, $\gamma\left(\mathbf{k}, \mu\left(x\right)\right) = FC_2\left(Concat\left(FC\left(\mathbf{k}\right), \mu\left(x\right)\right)\right)$. Here, FC, FC_1, and FC_2 are full-connected layers. $Concat$ represents concatenation operation across the channel dimension.

As shown in Figure 1(c), we insert AMLayer into the residual block to form the adaptive modulation residual block (AMRB), a basic block of our non-blind SR model – adaptive modulation network (AMNet) (please see in Figure 1(a)). To train AMNet, we employ widely used L1 loss for PSNR performance. Following [34, 27], once the AMNet is well-trained, we further adopt a weighted combination of L1 loss, VGG perceptual loss, and realness adversarial loss [29] for perceptual quality performance. We refer to such fine-tuned model as AMGAN. Due to limited pages, we provide the details of AMGAN in the supplementary file.

To accommodate the blind SR task, we construct a blur kernel predictor, and its structure is shown in Figure 3(c). The basic module is a residual block (RB), as depicted in Figure 3(d), which contains two 3×3 convolutional layers and one channel attention layer [38]. At the end of the network, following [22], we use global average pooling to aggregate spatial information to obtain the predicted kernel.

3.3. Adaptive Modulation Network with RL

As illustrated in Figure 3(a), the environment E is the combination of the renderer (AMNet or AMGAN) and reward function. To train our blind SR model, whether PSNR-oriented or perception-driven, we will utilize the following RL framework to accomplish. For the PSNR-oriented task, the pre-trained AMNet is the renderer. We adopt well-trained AMGAN as the renderer that translates the LR image and reduced kernel to the SR image for our perception-driven task. We define the reward function as

$$r = \zeta \cdot \left(\mathcal{M}\left(\mathcal{G}\left(\mathbf{I}^{LR}, \mathcal{P}\left(\mathbf{I}^{LR}\right)\right)\right) - \mathcal{M}\left(\mathcal{G}\left(\mathbf{I}^{LR}, \mathbf{k}_{GT}\right)\right)\right), \tag{8}$$

where $\mathcal{M}$ represents evaluation metrics, such as NIQE and PSNR. $\mathcal{G}$, $\mathcal{P}$ denote the renderer and actor, respectively. If $\mathcal{M}$ = NIQE (the lower value, the better), ζ is set to -1

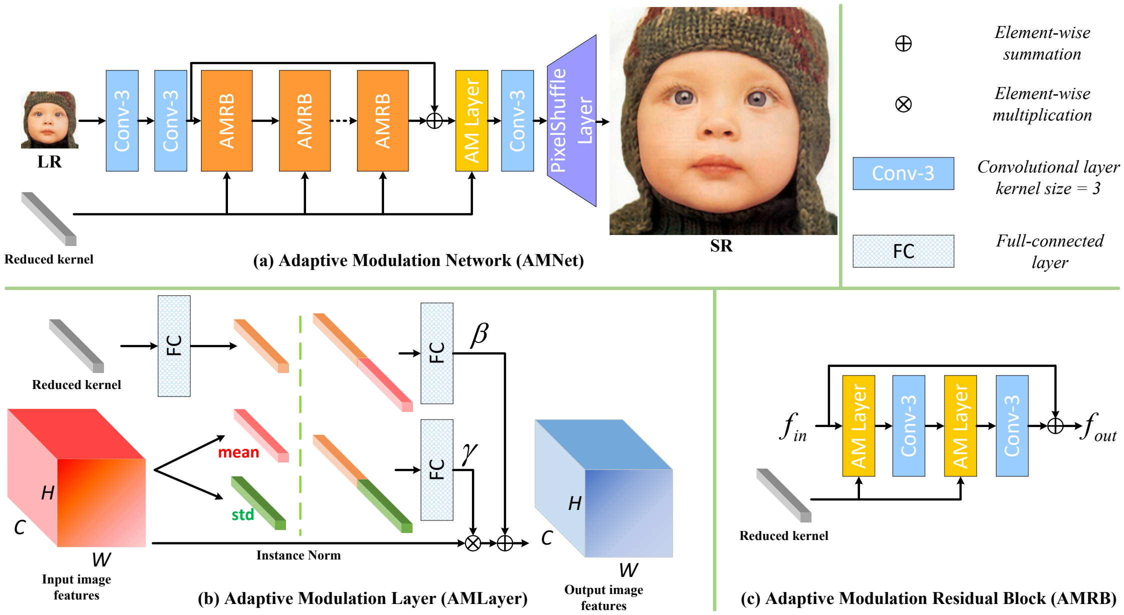

Figure 1. The overview architecture of the adaptive modulation network (AMNet).

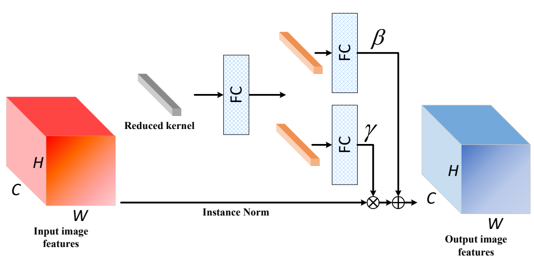

Figure 2. The structure of the AdaIN in Equation 4.

to ensure that the received reward is positive when the current policy is better than GT. Similarly, if $\mathcal{M}$ = PSNR (the higher value, the better), ζ is set to 1.

Since the action space is continuous, we select deep deterministic policy gradient (DDPG) [20] as a method of RL, that consists of a actor $\mu\left(s\left|\theta^{\mu}\right.\right)$ and a critic network $Q\left(s,a\left|\theta^{Q}\right.\right)$. The original DDPG optimizes the critic network parameter θ^{Q} by minimizing the loss:

$$L\left(\theta^{Q}\right)=\frac{1}{N}\sum\nolimits_{i}\left(y_i-Q\left(s_i,a_i\left|\theta^{Q}\right.\right)\right)^2, \tag{9}$$

where $y_i=r_i+\gamma Q'\left(s_{i+1},\mu'\left(s_{i+1}\left|\theta^{\mu'}\right.\right)\left|\theta^{Q'}\right.\right)$. Here, μ', Q', and γ are target critic network, target actor, and discounting factor, respectively. Since our task only needs to be executed in one step, Equation 9 can be simplified to:

$$L\left(\theta^{Q}\right)=\frac{1}{N}\sum\nolimits_{i}\left(r_i-Q\left(s_i,a_i\left|\theta^{Q}\right.\right)\right)^2. \tag{10}$$

If the actual reward $r_i=0.05$, the estimated $Q_i=-0.05$ and $Q_i=0.15$ can both obtain 0.1 error. However, we expect more penalty when $Q_i=-0.05$. Therefore, we further modify Equation 10 as follows:

$$\begin{aligned}L\left(\theta^{Q}\right)&=\frac{1}{N}\sum\nolimits_{i}\max\left(0,-\gamma*Q\left(s_i,a_i\left|\theta^{Q}\right.\right)\right)\\&+\left(r_i-Q\left(s_i,a_i\left|\theta^{Q}\right.\right)\right)^2.\end{aligned} \tag{11}$$

If $\gamma=1$, it indicates reward $r_i>0$, and vice-versa for $\gamma=-1$. The former term is to penalize the situation that r_i and Q_i are different signs. When r_i and Q_i are same signs, the first term is equal to zero.

Like the original DDPG, we update the actor by using policy gradient:

$$\begin{aligned}&\nabla_{\theta^{\mu}}J=\\&\frac{1}{N}\sum\nolimits_{i}\nabla_{a}Q\left(s,a\left|\theta^{Q}\right.\right)|_{s=s_i,a=\mu(s_i)}\nabla_{\theta^{\mu}}\mu\left(s\left|\theta^{\mu}\right.\right)|_{s=s_i}.\end{aligned} \tag{12}$$

4. Experiments

4.1. Implementation Details

Following [10, 22, 30], we collect 3450 high-quality RGB images from DIV2K [1] and Flickr2K [21] for training. We synthesize the degraded LR images according to Equation 1. For fair comparison with other methods, we adopt the same degradation setting that used in [10, 22], *i.e.*, we use isotropic Gaussian blur kernels. The range of kernel width is set to $[0.2, 4.0]$ for scale factor 4, and the kernel size is fixed to 21×21. During training, the input patch size is 64×64, and the mini-batch size is set to 64. We also perform randomly horizontal flip and 90 degree rotation for

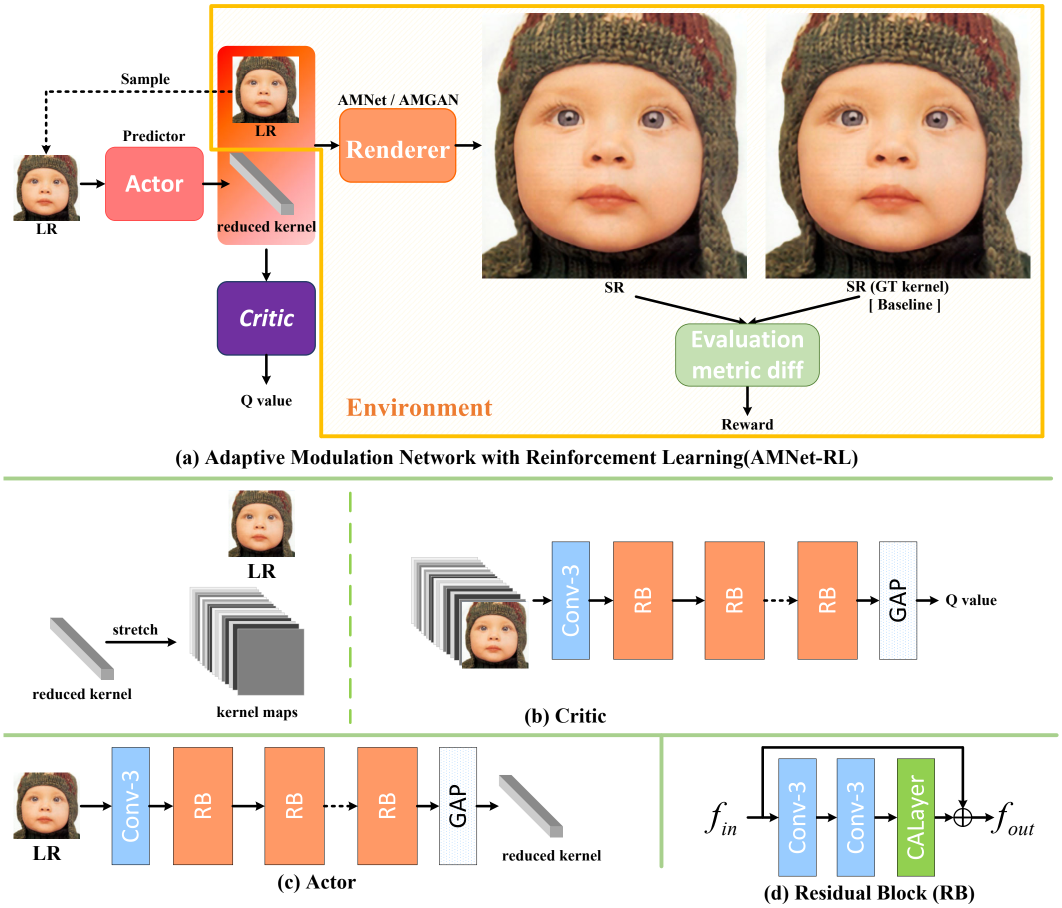

Figure 3. The overview architecture of the adaptive modulation network with reinforcement learning (AMNet-RL).

data augmentation. We use Adam optimizer with learning rate $lr = 2 \times 10^{-4}$. We train our models with PyTorch on 2 RTX 2080Ti GPUs. For AMGAN, its learning rate is set 1×10^{-4}, and the batch size is 16.

For quantitative evaluation, we use 6 widely used benchmark datasets: Set5 [3], Set14 [33], BSD100 [23], Urban100 [11], Manga109 [18], and PIRM_Val [4]. As in [10, 22], we also uniformly select 8 kernels, denoted as *Gaussian8*, from $[1.8, 3.2]$ for the quantitative evaluation of blind SR methods. The HR images are first blurred by the selected blur kernels and then downsampled to synthesize test images. The fidelity-oriented SR results are evaluated with PSNR and SSIM, and the perception-driven SR images are measured by learned perceptual image patch similarity (LPIPS) [36], and NIQE [24]. The lower the values of LPIPS and NIQE, the better.

We set the number of AMRB as 16 to form our AMNet, as shown in Figure 1(a). To further improve the performance, we also construct the larger AMNet, named AMNet_L, which contains 32 AMRBs. The reduced kernel dimension is set to 15 in all our experiments.

To stably train our AMNet-RL (or AMGAN-RL), the actor network depicted in Figure 3(c) should be pre-trained to make the actor has a better initialization. We also use Adam optimizer with batch size 96 and 5000 iterations, with a base learning rate of 1×10^{-5} for the actor network and 1×10^{-4} for the critic network. The learning rate is exponentially decayed to 10^{-2} of the original values during training.

4.2. Quantitative Results

We evaluate the performance of the proposed AMNet on isotropic Gaussian blur kernels with widths 0.2, 1.3, and 2.6. Table 1 shows the quantitative comparisons with the state-of-the-art non-blind SR methods SRMDNF [35], SFTMD [10], and UDVD [30]. It should be noted that AMNet_L uses fewer parameters than SFTMD and UDVD to achieve similar or higher performance. Since the topological structure of AMNet_L is SRResNet [19], which is the same as SFTMD and UDVD. It demonstrated the effect of AMLayer. Comparing with SRMDNF, the proposed AMNet with a similar model size achieves significantly better performance on all settings and datasets. It means that AM-

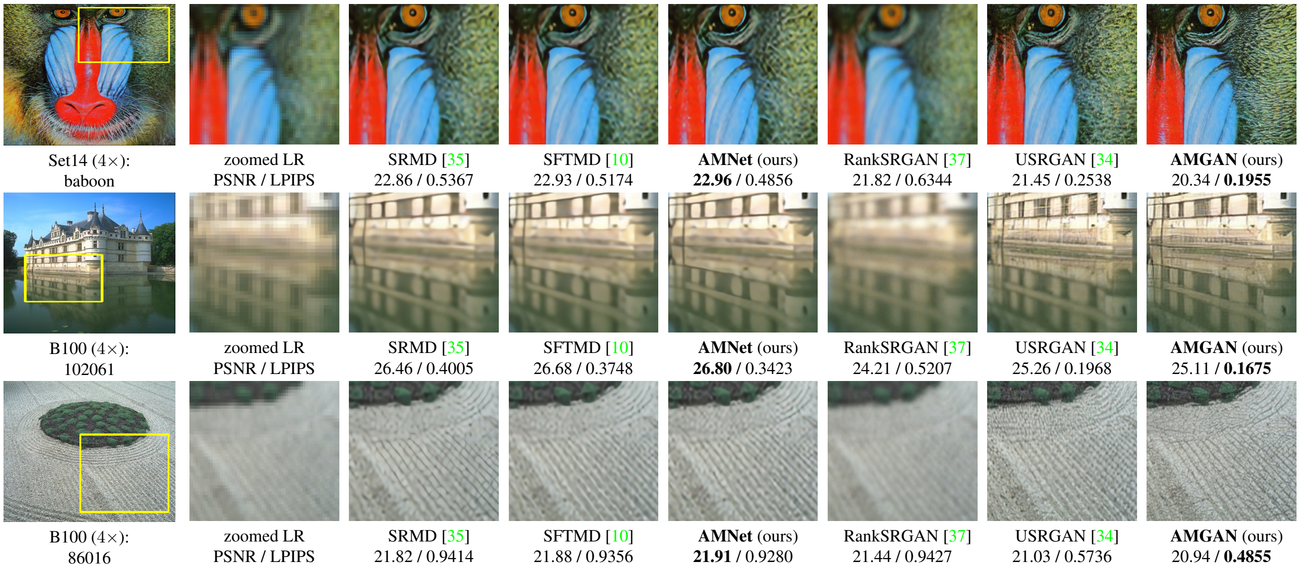

Figure 4. Visual results of various methods at scaling factor of 4. Note that RankSRGAN, USRGAN, and our AMGAN are perception-driven approaches.

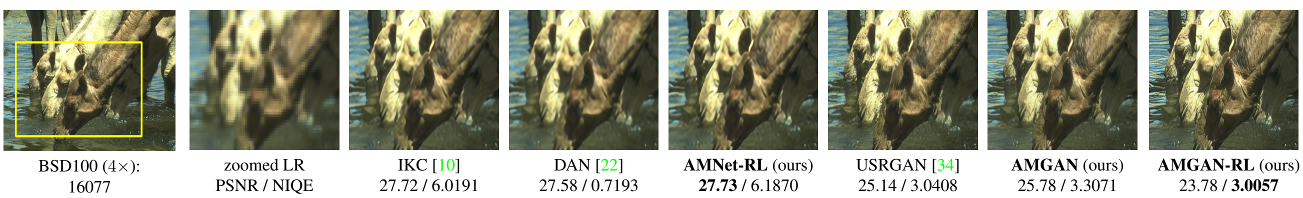

Figure 5. Visual results of various blind SR methods at scaling factor of 4.

Table 1. Quantitative evaluation (PSNR) with state-of-the-art non-blind SR methods for scale factor of 4. The comparison is conducted using three different isotropic Gaussian kernels on Set5, Set14, and BSD100 datasets. The best results are **highlighted**.

Methods	Kernel width	Params	Set5	Set14	BSD100
SRMDNF [35]	0.2	1,552K	31.96	28.35	27.49
SFTMD [10]		7,966K	32.39	28.77	27.58
UDVD [30]		4,933K	32.31	**28.78**	**27.70**
AMNet (ours)		1,599K	32.28	28.66	27.60
AMNet_L (ours)		3,121K	**32.46**	**28.78**	27.68
SRMDNF [35]	1.3	1,552K	32.00	28.42	27.53
SFTMD [10]		7,966K	32.41	28.82	27.64
UDVD [30]		4,933K	32.37	**28.85**	**27.75**
AMNet (ours)		1,599K	32.39	28.71	27.67
AMNet_L (ours)		3,121K	**32.57**	**28.85**	**27.75**
SRMDNF [35]	2.6	1,552K	31.77	28.26	27.43
SFTMD [10]		7,966K	32.05	28.55	27.47
UDVD [30]		4,933K	31.99	28.55	27.55
AMNet (ours)		1,599K	32.46	28.53	27.65
AMNet_L (ours)		3,121K	**32.65**	**28.75**	**27.77**

Net can better balance performance and model size. Besides, our method is stable under different blur kernels. For instance, AMNet and AMNet_L still perform well when kernel width is set to 2.6, while other algorithms show a serious performance drop compared with kernel width of 1.3. In Figure 4, we present visual comparisons on different datasets. Apart from SRMD [35] and SFTMD [10], we also include RankSRGAN [37], USRGAN [34] to compare with our AMGAN. For image "baboon" from Set14, we observe that our AMNet can produce much better visual results (sharper edges) than SRMD and SFTMD. Obviously, AMGAN yields much better pleasant details than AMNet. We can see that RankSRGAN does not perform well since the actual degradation deviates from bicubic degradation. This phenomenon has appeared in [34] and studied well in [9]. When comparing USRGAN and AMGAN, the former fails to generate natural textures while the latter synthesizes plausible textural details. In image "102061" from BSD100, our AMGAN can even clearly generate a reflection of the building in the water.

For blind setting, we show our quantitative evaluation results in Table 2, which is compared with two blind SR state-of-the-art approaches: IKC [10], and DAN [22]. The proposed AMNet_L-RL with fewer parameters achieves the best PSNR and SSIM performance on most evaluation datasets. For image "16077" in Figure 5, we can observe that IKC [10] and DAN [22] can produce fine results. As a comparison, our AMNet-RL generates a shaper outline.

Table 2. Average PSNR/SSIM results of different blind SR methods for scale factor of 4. The comparison is conducted using *Gaussian8* kernels on Set5, Set14, BSD100, Urban100, and Manga109. The best two results are highlighted in **red** and blue colors, respectively.

Methods	Params	Set5	Set14	BSD100	Urban100	Manga109
IKC [10] (blind)	9,047K	31.64 / 0.8820	28.29 / 0.7632	27.36 / 0.7183	25.31 / 0.7495	28.86 / 0.8774
DAN [22] (blind)	4,334K	31.89 / 0.8864	28.43 / 0.7687	27.51 / 0.7248	**25.86** / 0.7721	**30.50** / **0.9037**
AMNet-RL (blind)	1,982K	32.27 / 0.8941	28.47 / 0.7750	27.61 / 0.7326	25.63 / 0.7750	30.03 / 0.9004
AMNet_L-RL (blind)	3,504K	**32.34** / **0.8964**	**28.57** / **0.7779**	**27.66** / **0.7343**	**25.86** / **0.7794**	30.39 / 0.9024

Table 3. Average LPIPS/NIQE/PSNR results. The comparison is conducted using kernel widths 0.2, 1.3 and 2.6.

Dataset	Scores	USRGAN [34] (non-blind)	AMGAN (non-blind)	AMGAN-RL (blind)
		17.0M	1.60M	1.98M
Set5	LPIPS ↓	0.0967	**0.0707**	0.0778
	NIQE ↓	5.1147	5.047	**4.8206**
	PSNR ↑	29.55	**31.02**	29.36
Set14	LPIPS ↓	0.1648	**0.1254**	0.1366
	NIQE ↓	**3.7727**	4.2598	3.8897
	PSNR ↑	26.05	**27.14**	26.67
BSD100	LPIPS ↓	0.1975	0.1698	**0.1682**
	NIQE ↓	3.5579	4.0750	**3.4968**
	PSNR ↑	25.03	**26.36**	25.63
PIRM_Val	LPIPS ↓	0.1528	**0.1197**	0.1418
	NIQE ↓	3.0580	3.3672	**3.0021**
	PSNR ↑	25.235	**26.50**	25.36

Comparing with USRGAN [34], our AMGAN synthesize finer textures. AMGAN-RL shows stronger contrast than AMGAN in Figure 5.

4.3. Study of AMLayer

Table 4. Comparison with AdaIN and the presented AMLayer. The kernel width of the isotropic Gaussian blur is set to 1.8.

Module	Set5	Set14	BSD100	Urban100
AdaIN	32.12	28.53	27.56	25.66
AMLayer	**32.43**	**28.67**	**27.68**	**25.81**

To validate the effectiveness of the adaptive modulation layer (AMLayer), we replace all AMLayers in our AMNet with the AdaIN layer that is described in Equation 4 (please see Figure 2). From Table 4, we can find out that the AMLayer leads to performance improvement (PSNR: **+0.31dB** for Set5). It indicates that our AMLayer is very suitable for the reduced kernel information transformation to the image feature maps.

4.4. Investigation of $L\left(\theta^Q\right)$

To verify the necessity of using Equation 11, we train our AMGAN-RL by Equation 10 and Equation 11, receptively. From Table 5, we observe that adding different signs penalty can assist the converge of our critic network, which further helps actor network optimize better.

4.5. Running Time

Without an iteration scheme, our AMNet-RL has a higher inference speed. From Table 6, we evaluate the average speed of different methods on the same platform with one RTX 2080Ti GPU. When the input sizes are 64×64, the computational cost of IKC [10] has reached 199 GFLOPs. DAN [22] has a much fewer computational cost, which is about 78 GFLOPs. As a comparison, our AMNet-RL only requires 14.50 ms to process a 64×64 image, nearly four times faster than DAN, and 11 times faster than IKC. Combined with the results in Table 2, our AMNet-RL achieves dominant performance in terms of the trade-off between the running time and PSNR value.

Table 5. Comparison with Equation 10 and Equation 11. The average NIQE values are evaluated on Set5 with kernel widths 0.2, 1.3 and 2.6.

Loss function	0.2	1.3	2.6
Equation 10	4.98	4.83	4.87
Equation 11	**4.88**	**4.77**	**4.80**

Table 6. FLOPs and average inference time (measured by *torch.cuda.Event(enable_timing=True)*).

Input size	Methods	FLOPs	Time
64×64	IKC [10] (iterations=7)	199 G	171.66ms
	DAN [22] (iteration=4)	78 G	63.51 ms
	AMNet-RL (ours)	6.6 G	14.50 ms
	AMNet_L-RL (ours)	11.5 G	26.18 ms

5. Conclusion

This paper proposes an adaptive modulation layer to compose a novel multiple degradations SR model, which yields better performance and less computational costs. To alleviate the non-differentiable evaluation metrics optimization problem, we introduce RL into the blind SR framework. Through numerical experiments, we demonstrate that our whole blind SR framework can reach comparable performance. Our blind SR framework with RL can also convert any evaluation metrics (differentiable of non-differentiable) to the reward function, guiding the actor to sample the correct action. We believe that our work can build a bridge in the blind SR and quality assessment fields.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants 62036007, 61772402, 62050175, 61972305 and 61871308.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *CVPR Workshops*, pages 126–135, 2017. 5

[2] Sefi Bell-Kligler, Assaf Shocher, and Michal Irani. Blind super-resolution kernel estimation using an internal-gan. In *NeurIPS*, pages 284–293, 2019. 3

[3] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie line Alberi Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In *BMVC*, pages 135.1–135.10, 2012. 6

[4] Yochai Blau, Roey Mechrez, Radu Timofte, Tomer Michaeli, and Lihi Zelnik-Manor. 2018 pirm challenge on perceptual image super-resolution. In *ECCV Workshop*, 2018. 6

[5] Qingxing Cao, Liang Lin, Yukai Shi, Xiaodan Liang, and Guanbin Li. Attention-aware face hallucination via deep reinforcement learning. In *CVPR*, pages 1656–1664, 2017. 3

[6] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *ECCV*, pages 184–199, 2014. 1, 2

[7] 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, 2016. 1, 2

[8] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *ECCV*, pages 391–407, 2016. 1

[9] Netalee Efrat, Daniel Glasner, Alexander Apartsin, Boaz Nadler, and Anat Levin. Accurate blur models vs. image priors in single image super-resolution. In *ICCV*, pages 2832–2839, 2013. 7

[10] Jinjin Gu, Hannan Lu, Wangmeng Zuo, and Chao Dong. Blind super-resolution with iterative kernel correction. In *CVPR*, pages 1604–1613, 2019. 1, 2, 3, 4, 5, 6, 7, 8

[11] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *CVPR*, pages 5197–5206, 2015. 6

[12] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, pages 1501–1510, 2017. 2, 4

[13] Zheng Hui, Xiumei Wang, and Xinbo Gao. Fast and accurate single image super-resolution via information distillation network. In *CVPR*, pages 723–731, 2018. 1, 2

[14] Justin Johnson, Alexandre Alahi, and Fei-Fei Li. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, pages 694–711, 2016. 2

[15] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial. In *CVPR*, pages 4401–4410, 2019. 2, 4

[16] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *CVPR*, pages 1646–1654, 2016. 1, 2

[17] Satoshi Kosugi and Toshihiko Yamasaki. Unpaired image enhancement featuring reinforcement-learning-controlled image editing software. In *AAAI*, pages 11296–11303, 2020. 2

[18] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *CVPR*, pages 624–632, 2017. 1, 6

[19] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, pages 4681–4690, 2017. 1, 2, 6

[20] Timothy P. Lillicrap, Jonathan J. Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. In *ICLR*, 2016. 5

[21] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *CVPR Workshops*, pages 136–144, 2017. 5

[22] Zhengxiong Luo, Yan Huang, Shang Li, Liang Wang, and Tieniu Tan. Unfolding the alternation optimization for blind super resolution. In *NeurIPS*, 2020. 1, 2, 3, 4, 5, 6, 7, 8

[23] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *CVPR*, pages 416–423, 2001. 6

[24] Anish Mittal, Rajiv Soundararagan, and Alan C. Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal Processing Letters (SPL)*, 20(3):209–212, 2013. 2, 6

[25] Cornillère Victor, Djelouah Abdelaziz, Yifan Wang, Sorkine-Hornung Olga, and Schroers Christopher. Blind image super-resolution with spatially variant degradations. *ACM Transactions on Graphics (TOG)*, 38(6), 2019. 2

[26] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *CVPR*, pages 606–615, 2018. 3

[27] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Chen Change Loy, Yu Qiao, and Xiaoou Tang. Esrgan: Enhanced super-resolution generative adversarial networks. In *ECCV Workshops*, pages 63–79, 2018. 1, 2, 4

[28] Kaixuan Wei, Angelica Aviles-Rivero, Jingwei Liang, Ying Fu, Carola-Bibiane Schnlieb, and Hua Huang. Tuning-free plug-and-play proximal algorithm for inverse imaging problems. In *ICML*, 2020. 3

[29] Yuanbo Xiangli, Yubin Deng, Bo Dai, Chen Change Loy, and Dahua Lin. Real or not real, that is the question. In *ICLR*, 2020. 4

[30] Yu-Syuan Xu, Shou-Yao Tseng, Yu Tseng, Hsien-Kai Kuo, and Yi-Min Tsai. Unified dynamic convolutional network for super-resolution with variational degradations. In *CVPR*, pages 12496–12505, 2020. 1, 2, 3, 4, 5, 6, 7

[31] Ke Yu, Chao Dong, Liang Lin, and Chen Change Loy. Crafting a toolchain for image restoration by deep reinforcement learning. In *CVPR*, pages 2443–2452, 2018. 3

[32] Ke Yu, Xintao Wang, Chao Dong, Xiaoou Tang, and Chen Change Loy. Path-restore: Learning network path se-

lection for image restoration. *arXiv:1904.10343 [cs.CV]*, 2019. 3

[33] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *Curves and Surfaces*, pages 711–730, 2010. 6

[34] Kai Zhang, Luc Van Gool, and Radu Timofte. Deep unfolding network fro image super-resolution. In *CVPR*, pages 3217–3226, 2020. 1, 2, 4, 7, 8

[35] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Learning a single convolutional super-resolution network for multiple degradations. In *CVPR*, pages 3262–3271, 2018. 1, 2, 4, 6, 7

[36] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pages 586–595, 2018. 2, 6

[37] Wenlong Zhang, Yihao Liu, Chao Dong, and Yu Qiao. Ranksrgan: Generative adversarial networks with ranker for image super-resolution. In *ICCV*, pages 3096–3105, 2019. 2, 7

[38] 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*, pages 294–310, 2018. 3, 4

[39] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. Residual dense network for image super-resolution. In *CVPR*, pages 2472–2481, 2018. 1, 2