Blind Image Super-resolution with Elaborate Degradation Modeling on Noise and Kernel

Zongsheng Yue[1], Qian Zhao[2], Jianwen Xie[3], Lei Zhang[4], Deyu Meng[2,5], Kwan-Yee K. Wong[1]

[1]The University of Hong Kong, Hong Kong, China [2]Xi'an Jiaotong University, Xi'an, China
[3]Cognitive Computing Lab, Baidu Research, Bellevue, USA
[4]The Hong Kong Polytechnic University, Hong Kong, China
[5]Peng Cheng Laboratory, Shenzhen, China

Abstract

While researches on model-based blind single image super-resolution (SISR) have achieved tremendous successes recently, most of them do not consider the image degradation sufficiently. Firstly, they always assume image noise obeys an independent and identically distributed (i.i.d.) Gaussian or Laplacian distribution, which largely underestimates the complexity of real noise. Secondly, previous commonly-used kernel priors (e.g., normalization, sparsity) are not effective enough to guarantee a rational kernel solution, and thus degenerates the performance of subsequent SISR task. To address the above issues, this paper proposes a model-based blind SISR method under the probabilistic framework, which elaborately models image degradation from the perspectives of noise and blur kernel. Specifically, instead of the traditional i.i.d. noise assumption, a patch-based non-i.i.d. noise model is proposed to tackle the complicated real noise, expecting to increase the degrees of freedom of the model for noise representation. As for the blur kernel, we novelly construct a concise yet effective kernel generator, and plug it into the proposed blind SISR method as an explicit kernel prior (EKP). To solve the proposed model, a theoretically grounded Monte Carlo EM algorithm is specifically designed. Comprehensive experiments demonstrate the superiority of our method over current state-of-the-arts on synthetic and real datasets. The source code is available at `https://github.com/zsyOAOA/BSRDM`.

1. Introduction

Single image super-resolution (SISR) is a fundamental problem in computer vision. It aims to recover the sharp detailed high-resolution (HR) counterpart from an observed low-resolution (LR) image. Image degradation, the functional opposite of image super-resolution, is the process of generating a LR image from the HR one. Unfortunately, the degradation model is always unknown while complicated, making the blind SISR problem extremely challenging. How to rationally and practically model the degradation is therefore of great significance in blind SISR.

Early methods [14, 24, 44, 45] simply regard SISR as an interpolation problem. They have fast processing speed but always blur high frequency details. Later methods begin to consider the image degradation, and can be roughly divided into two categories, namely model-based methods and learning-based methods. From the Bayesian perspective, model-based methods [11,20,25,36,40,42] firstly build a generative model based on the image degradation and then estimate the blur kernel and the HR image under the maximum a posteriori (MAP) framework. Such MAP estimation is implemented for each LR image individually, and thus tends to achieve better generalization for unknown degradations. Learning-based methods [10, 25, 56, 65], on the other hand, aim to learn a unified super-resolver based on a large amount of LR/HR image pairs synthesized according to the pre-assumed degradation model. Recently, to improve their generalization, some works [8,16,28,51,54] attempt to learn the degradation model from unpaired real image data. However, these learning-based methods rely heavily on the collected training data, and may suffer from a severe performance drop when unseen degradations show up in testing. In this paper, we follow the model-based methodology for its better generalization capability.

Most of the model-based blind SISR methods can be generally formulated as the following MAP problem:

$$\max_{\boldsymbol{x},\boldsymbol{k}} \log p(\boldsymbol{y}|\boldsymbol{x},\boldsymbol{k}) + \log p(\boldsymbol{k}) + \log p(\boldsymbol{x}), \qquad (1)$$

where $\boldsymbol{y}$, $\boldsymbol{x}$, and $\boldsymbol{k}$ denote the observed LR image, the underlying HR image, and the blur kernel, respectively. The last term represents the image prior, while the first and second terms deliver our knowledges on the degradation model (i.e., noise distribution and kernel prior). Most of the existing methods focus on designing more rational image priors,

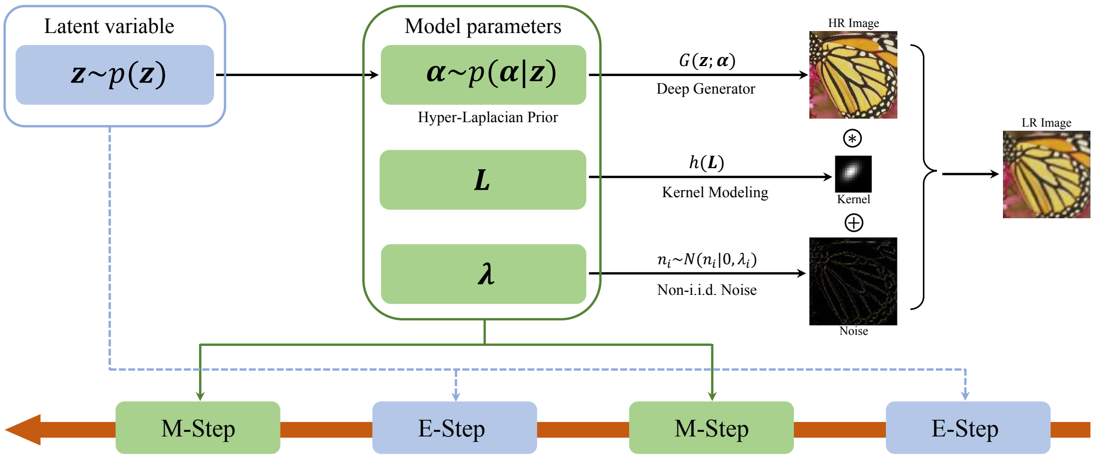


Figure 1. An overview of the proposed SISR method and the corresponding EM algorithm. A probabilistic model is constructed to depict the generation process of the observed LR image, which mainly involves two groups of parameters, including the latent variable z and the model parameters $\{\alpha, L, \lambda\}$. A Monte Carlo EM algorithm is designed to alternately update them in the E-Step and M-Step, respectively.

such as gradient profile [42], sparsity [7, 20], DIP [46] and so on [9, 11, 22, 32, 34, 38]. However, they often do not sufficiently consider the degradation model:

- As for noise modeling, most of current method adopt the independent and identically distributed (i.i.d.) Gaussian or Laplacian distribution to model the noise. Such a simple noise assumption, however, usually underestimates the complexity of real image noise and shows limited robustness in practical applications. For example, the most common camera sensor noise affected by the in-camera pipeline is signal-depedent, and thus exhibits evident non-i.i.d. property in statistics.
- As for kernel modeling, traditional methods often ignore it or adopt some heuristic priors, e.g., normalization (i.e., the kernel elements sum to 1) [17] and sparsity [3], which usually cannot guarantee a rational kernel solution. Recently, Liang *et al.* [25] trained an implicit mapping parameterized as a convolutional neural network (CNN) from the latent noises to anisotropic Gaussian kernels, and then embedded it into the blind SISR as a kernel prior. Albeit achieving evident performance improvement, this method depends on a time-consuming and labor-cubersome pre-training phase. Moreover, the fitting error, which is inevitable during training, may be enlarged in the alternate iterations between the kernel estimation and super-resolution tasks. The performance of blind SISR can be therefore further improved by designing an explicit yet effective kernel prior.

To address the above issues, this paper proposes a probabilistic blind SISR method that elaborately considers the noise and kernel modeling (see Fig. 1). To better model the complicated real noise, a patch-based non-i.i.d. Gaussian noise assumption is adopted instead of the conventional i.i.d. one. Under such setting, each $p \times p$ image patch has its own noise parameter, which complies better with the configurations of real noise. As for blur kernel, we observe that it can be formulated as an explicit and differentiable function in terms of the covariance matrix. This inspire us to construct an explicit kernel prior (EKP) for the generally used anisotropic Gaussian kernel, which can be easily embedded into current deep learning (DL)-based blind SISR methods. In summary, the contributions of this work is three-fold:

1. Different from the commonly-used i.i.d. Gaussian or Laplacian distribution, a patch-based non-i.i.d. noise distribution is employed in the proposed method, making it able to handle complicated real noise.
2. A generative kernel prior named EKP is novelly constructed for the blind SISR task. It is with explicit and concise form, and substantiated to be able to attain a more stable kernel estimation for SISR.
3. A theoretically grounded Monte Carlo EM algorithm (see Fig. 1) is designed to solve our proposed model.

2. Related Work

In this section, we briefly review the literatures on image degradation models and blind SISR.

2.1. Image Degradation Model

Image degradation model is a long-standing and open research topic in SISR. The most common and also simplest degradation model is bicubic downsampling, which is widely used to synthesize training and testing data in many SISR works [6, 11, 19, 23, 64]. More general degradation models, consisting of a sequence of blurring, downsampling and noise addition, are also widely adopted by previous works [37, 41, 58, 61, 65]. Recently, Zhang *et al.* [59] propose a more practical degradation model by introducing a random shuffle strategy among the blurring, downsampling and noise addition, and more practical camera sensor and JPEG compression noises in the noise addition procedure. Furthermore, Wang *et al.* [47] consider the common ringing and overshoot artifacts, and propose a high-order degradation model to cover a larger degradation space.

2.2. Blind SISR Methods

As mentioned in the introduction, other than the heuristic interpolation-oriented methods [14, 18, 24, 44], most of the existing methods can be loosely divided into two categories, namely model-based and learning-based methods. Even though this paper focuses on model-based methods, we also briefly review learning-based methods for completeness.

Model-based Methods. Model-based methods mainly focus on designing the three terms in Eq. (1), i.e, the likelihood, kernel prior, and image prior. The image prior has received more attentions in the past decades. Typical traditional image priors include total variation (TV) [38], hyper-Laplacian [22], gradient profile [42], sparsity [20], and non-local similarity [7]. With the prevalence of deep learning, more DL-base image priors have been proposed. A representative work is proposed by Ulyanov *et al.* [46], namely deep image prior (DIP), to capture the low-level image statistics. Shocher *et al.* [41] attempt to recover the HR image using the prior of patch recurrence across scales. More related works can be found in [2, 34].

Kernel prior is another important part in the MAP framework. Traditional blind SISR methods only consider some heuristic kernel priors, such as normalization [17] and sparsity [3]. Recently, some works begin to implicitly model the kernel by DNN. For example, Ren *et al.* [36] propose to model the kernel prior using a multilayer perceptron (MLP), while Liang *et al.* [25] train a flow-based kernel prior named FKP for blind SISR. Instead of such an implicitly modeling manner, this paper attempts to design an explicit and concise kernel prior, hoping to induce a more stable kernel estimation in blind SISR task.

As for the likelihood, most of the existing methods adopt an i.i.d. Gaussian or Laplacian distribution, which often fails to comply with the configurations of real noise and causes a performance drop in real scenarios. To address this issue, this work employs a non-i.i.d. noise modeling method to better deliver the real noise configurations and thus improve its generalized capability.

Learning-based Methods. The main idea of learning-based methods is to learn a super-resolver from large amount of pre-simulated LR/HR image pairs. Dong *et al.* [6] firstly propose to learn an end-to-end CNN mapping from LR to HR images. Later, plenty of CNN architectures are designed for SISR [13, 19, 26, 31, 43, 48, 64]. Recently, a flurry of unpaired SISR methods [8, 16, 28, 51, 54] have been proposed, due to the fact that real LR images rarely come with the corresponding HR images in practice.

3. The Proposed Method

3.1. Degradation Assumption

Various degradation models have been proposed in previous works. Most of them can be written as a downsampling with a subsequent noise addition process, i.e.,

$$\boldsymbol{y} = D(\boldsymbol{x}; \boldsymbol{k}, \downarrow_s) + \boldsymbol{n}, \tag{2}$$

where $\boldsymbol{y}$ and $\boldsymbol{x}$ denote the LR and HR images, respectively, $D(\boldsymbol{x}; \boldsymbol{k}, \downarrow_s)$ represents the downsampling process with a blur kernel $\boldsymbol{k}$ and s-fold downsampler $\downarrow_s$, and $\boldsymbol{n}$ is the noise. In fact, the real LR image may be also obtained by firstly adding noise and then downsampling the HR image [59], which makes the noise more complicated. This process can also be formulated in the same format as Eq. (2), i.e.,

$$\boldsymbol{y} = D(\boldsymbol{x} + \boldsymbol{n}; \boldsymbol{k}, \downarrow_s) = D(\boldsymbol{x}; \boldsymbol{k}, \downarrow_s) + \hat{\boldsymbol{n}}, \tag{3}$$

where $\hat{\boldsymbol{n}} = D(\boldsymbol{n}; \boldsymbol{k}, \downarrow_s)$. Hence, we only need to consider the degradation sequence in Eq. (2).

For the blur kernel $\boldsymbol{k}$, we assume it to be the general anisotropic Gaussian kernel which is sufficient for SISR as pointed out in [37, 58]. Furthermore, considering different settings for the downsampler $\downarrow_s$ (e.g., bicubic [19] and direct[1] [58]) and the imposed order between the blurring and downsampling procedures (i.e., $(\boldsymbol{x} * \boldsymbol{k}) \downarrow_s$ [58] and $(\boldsymbol{x} \downarrow_s) * \boldsymbol{k}$ [62], where $*$ is the convolution operator), we can obtain multiple different degradation assumptions based on Eq. (2). This paper aims to propose a blind SISR method with elaborate considerations on noise and kernel modeling, which does not depend on the format of the downsampler and the specific imposed order between the blurring and downsampling procedures. For the ease of presentation, we adopt the most widely used degradation assumption to construct our SISR model in the next subsection, i.e.,

$$\boldsymbol{y} = (\boldsymbol{x} * \boldsymbol{k}) \downarrow_s^d + \boldsymbol{n}, \tag{4}$$

where $\downarrow_s^d$ is the direct downsampler with a scale factor s.

3.2. Probabilistic SISR Model

In this subsection, we are going to build our blind SISR method based on the degradation model in Eq. (4),

Non-i.i.d. Noise Modeling. Different from the traditional i.i.d. Gaussian or Laplacian noise assumption on the whole image, a patch-based non-i.i.d. noise model is proposed in this work. Given any observed LR image $\boldsymbol{y} \in \mathbb{R}^{h \times w}$, where h and w denote the image height and width, respectively, we regard $\boldsymbol{y}$ as N ($N = hw$) highly overlapped $p \times p$ patches. Furthermore, we assume that the noises contained in each patch obey a different zero-mean Gaussian distribution with

[1] Direct downsampler with a scale factor s means keeping the upper-left pixel for each distinct $s \times s$ patch and discarding the rest.

its own variance parameter. Specifically, considering the i-th image patch centered at y_i, we have

$$y_i \sim \mathcal{N}\left(y_i|[(\boldsymbol{x} * \boldsymbol{k}) \downarrow_s^d]_i, \lambda_i\right), \ i = 1, 2, \cdots, N, \tag{5}$$

where λ_i is the noise variance for the i-th image patch.

In previous researches, they often assume the noise as additive white Gaussian noise (AWGN), which is indeed a special case of our non-i.i.d. noise distribution. By regarding the whole image as one large patch with size $h \times w$, our noise model then naturally degenerates to AWGN, but with noise variance parameter being automatically updated during learning (see Sec. 4) instead of manually adjusted.

Kernel Prior. Based on the anisotropic Gaussian assumption on the blur kernel, we construct a concise yet effective kernel prior. For any blur kernel $\boldsymbol{k}$ with size $(2r+1)\times(2r+1)$, it is defined as follows:

$$k_{ij} = \frac{1}{2\pi}\sqrt{|\boldsymbol{\Lambda}|}\exp\left\{-\frac{1}{2}\boldsymbol{S}^T\boldsymbol{\Lambda}\boldsymbol{S}\right\}, \ \ i,j \in \{-r, \cdots, r\}, \tag{6}$$

where $\boldsymbol{\Lambda}$ is the precision matrix, $\boldsymbol{S} = \left[\begin{smallmatrix} i \\ j \end{smallmatrix}\right]$ is the spatial coordinate. From Eq. (6), it can be observed that the blur kernel is completely determined by the precision matrix $\boldsymbol{\Lambda}$ after fixing the kernel size. Note that Eq. (6) is differentiable w.r.t. $\boldsymbol{\Lambda}$. This implies that it can be regarded as a kernel generator, in which $\boldsymbol{\Lambda}$ can be easily optimized with stochastic gradient descent (SGD) under the DL framework.

Another tricky issue is how to guarantee the positive-definiteness of the precision matrix $\boldsymbol{\Lambda}$ during optimization. Inspired by the Cholesky decomposition, we reparameterize $\boldsymbol{\Lambda}$ as follows:

$$\boldsymbol{\Lambda} = \boldsymbol{L}\boldsymbol{L}^T, \tag{7}$$

where $\boldsymbol{L} \in \mathcal{R}^{2\times 2}$ is a lower triangular matrix. By substituting Eq. (7) into Eq. (6), we obtain the following explicit kernel prior termed EKP,

$$k_{ij} = h(\boldsymbol{L}) = \frac{1}{2\pi}|\boldsymbol{L}|\exp\left\{-\frac{1}{2}\boldsymbol{S}^T\boldsymbol{L}\boldsymbol{L}^T\boldsymbol{S}\right\}. \tag{8}$$

In practice, to make $\boldsymbol{L}$ be triangular during optimization, we rewrite $\boldsymbol{L}$ as $\boldsymbol{L} = \boldsymbol{Q} \odot \boldsymbol{M}$, where $\boldsymbol{M} = \left[\begin{smallmatrix} 1 & 0 \\ 1 & 1 \end{smallmatrix}\right]$ and $\odot$ is the Hadamard product, and turn to optimize $\boldsymbol{Q}$.

To our best knowledge, the most effective kernel prior for SISR is FKP [25]. The main idea of FKP is to firstly train a deep generator that maps the latent noises to anisotropic Gaussian kernels, and then use the pre-trained generator to estimate the blur kernel by only adjusting the latent noises. The inevitable fitting error of this generator may be enlarged when applying it in blind SISR, and thus limits the final performance. Comparing with FKP, the advantages of our proposed EKP is three-fold: 1) EKP is an explicit kernel generator that does not rely on pre-training, making it more convenient to be used in SISR. 2) The generated kernel by EKP is always an exact anisotropic Gaussian kernel, which naturally avoids the issue of fitting error in FKP. 3) In EKP, the kernel $\boldsymbol{k}$ is completely controlled by $\boldsymbol{L}$, which contains much fewer parameters than that of the latent noise vector in FKP (3 vs. $11^2/15^2/19^2$ for scale 2/3/4, respectively). This makes EKP more easier to be optimized after pluging into the blind SISR as a kernel prior.

Image Prior. We employ a CNN-based generator G to generate the HR image from the latent space, i.e.,

$$\boldsymbol{x} = G(\boldsymbol{z}; \boldsymbol{\alpha}), \tag{9}$$

where $\boldsymbol{z}$ and $\boldsymbol{\alpha}$ denote the latent variable and network parameters, respectively. As demonstrated in [46], G is very easy to overfit onto the image noise due to the powerful fitting capability of CNN. Therefore, we introduce the conventional hyper-Laplacian prior to constrain the statistical regularity of the generated HR image through the following joint distribution of $\boldsymbol{\alpha}$ and $\boldsymbol{z}$:

$$(\boldsymbol{\alpha}, \boldsymbol{z}) \sim p(\boldsymbol{\alpha}, \boldsymbol{z}) = p(\boldsymbol{\alpha}|\boldsymbol{z})p(\boldsymbol{z}), \tag{10}$$

$$p(\boldsymbol{\alpha}|\boldsymbol{z}) \propto \exp\left(-\rho\sum_{k=1}^{2}|f_k * G(\boldsymbol{z};\boldsymbol{\alpha})|^{\gamma}\right), \tag{11}$$

$$p(\boldsymbol{z}) = \mathcal{N}(\boldsymbol{z}|0, \boldsymbol{I}), \tag{12}$$

where $\{f_k\}_{k=1}^2$ are the gradient filters along the horizontal and vertical directions, ρ and γ are both hyper-paramters.

As for the generator G, we follow the "hourglass" architecture in DIP [46] but use a tiny version that contains much fewer parameters. The detailed network architecture can be found in supplementary material (SM).

3.3. MAP Estimation

According to Eqs. (5)-(12), a full probabilistic model is constructed. Under the MAP framework, our goal turns to maximize the following posterior:

$$p(\boldsymbol{\alpha}, \boldsymbol{L}, \boldsymbol{\lambda}|\boldsymbol{y}) \propto \int p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{L}, \boldsymbol{\lambda}, \boldsymbol{z})p(\boldsymbol{\alpha}|\boldsymbol{z})p(\boldsymbol{z})\mathrm{d}\boldsymbol{z}. \tag{13}$$

Note that we have omitted the prior terms $p(\boldsymbol{L})$ and $p(\boldsymbol{\Lambda})$, since they are set as non-informative priors in our model. Taking the logarithm of both sides of Eq. (13), we have the following maximization problem:

$$\begin{aligned} &\max_{\boldsymbol{\alpha},\boldsymbol{L},\boldsymbol{\lambda}} \log p(\boldsymbol{\alpha}, \boldsymbol{L}, \boldsymbol{\lambda}|\boldsymbol{y}) \\ &= \log\int p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{L}, \boldsymbol{\lambda}, \boldsymbol{z})p(\boldsymbol{\alpha}|\boldsymbol{z})p(\boldsymbol{z})\mathrm{d}\boldsymbol{z} + \text{const.} \end{aligned} \tag{14}$$

4. Inference Algorithm

Inspired by [52, 53], we design a Monte Carlo expectation-maximization (EM) algorithm [5] to solve

Algorithm 1 Inference procedure for the proposed method

Input: observed LR image, hyper-paramter settings.
Output: the super-resolved HR image I^{HR}.

1: Initialize the model parameters $\{\boldsymbol{\alpha}, \boldsymbol{L}, \boldsymbol{\lambda}\}$ and the latent variable $\boldsymbol{z}$.
2: **while** not converged **do**
3: **E-Step:** Sample the latent variable $\boldsymbol{z}$ from $p_{\text{old}}(\boldsymbol{z}|\boldsymbol{y})$ following Eq. (15).
4: **M-Step:** (a) Update parameters $\boldsymbol{\alpha}$ and $\boldsymbol{L}$ with fixed $\boldsymbol{\lambda}$ according to Eq. (18).
5: (b) Update noise variance parameter $\boldsymbol{\lambda}$ with fixed $\boldsymbol{\alpha}$ and $\boldsymbol{L}$ according to Eq. (19).
6: **end while**
7: $I^{\text{HR}} = G(\boldsymbol{z}; \boldsymbol{\alpha})$.

Eq. (14), which alternately samples the latent variable $\boldsymbol{z}$ from its posterior $p(\boldsymbol{z}|\boldsymbol{y})$ in E-Step and updates the model parameters $\{\boldsymbol{\alpha}, \boldsymbol{L}, \boldsymbol{\lambda}\}$ in M-Step. The whole inference framework is illustrated in Fig. 1.

E-Step. Given current model parameters $\{\boldsymbol{\alpha}_{\text{old}}, \boldsymbol{L}_{\text{old}}, \boldsymbol{\lambda}_{\text{old}}\}$, we denote the posterior of $\boldsymbol{z}$ under them as $p_{\text{old}}(\boldsymbol{z}|\boldsymbol{y})$. In E-Step, our goal is to sample $\boldsymbol{z}$ from $p_{\text{old}}(\boldsymbol{z}|\boldsymbol{y})$ using Langevin dynamics [50]:

$$\boldsymbol{z}^{(\tau+1)} = \boldsymbol{z}^{(\tau)} + \frac{\delta^2}{2}\left[\frac{\partial}{\partial \boldsymbol{z}} \log p_{\text{old}}(\boldsymbol{z}|\boldsymbol{y})\right]\Bigg|_{\boldsymbol{z}=\boldsymbol{z}^{(\tau)}} + \delta\boldsymbol{\zeta}^{(\tau)}, \quad (15)$$

where τ indexs the time step for Langevin dynamics, δ denotes the step size, $\boldsymbol{\zeta}$ is the Gaussian white noise used to prevent trapping into local modes. A key note to calculate Eq. (15) is $\frac{\partial}{\partial \boldsymbol{z}} \log p_{\text{old}}(\boldsymbol{z}|\boldsymbol{y}) = \frac{\partial}{\partial \boldsymbol{z}} \log p_{\text{old}}(\boldsymbol{z}, \boldsymbol{y})$, and the detailed calculation can be found in SM.

In practice, a small trick to accelerate the convergence speed of Monte Carlo sampling in Eq. (15) is to start from the previous updated $\boldsymbol{z}$ in each learning iteration. We empirically found that it performs very stably and well by simply sampling 10 times according to Eq. (15).

M-Step. Let's denote the sampled latent variable in E-step as $\tilde{\boldsymbol{z}}$, M-Step aims to maximize the approximate lower bound of Eq. (14) w.r.t. the model parameters $\{\boldsymbol{\alpha}, \boldsymbol{L}, \boldsymbol{\lambda}\}$:

$$\begin{aligned}\max_{\boldsymbol{\alpha},\boldsymbol{L},\boldsymbol{\lambda}} Q(\boldsymbol{\alpha}, \boldsymbol{L}, \boldsymbol{\lambda}) &= \int p_{\text{old}}(\boldsymbol{z}|\boldsymbol{y}) \log p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{L}, \boldsymbol{\lambda}, \boldsymbol{z}) p(\boldsymbol{\alpha}|\boldsymbol{z}) p(\boldsymbol{z}) \mathrm{d}\boldsymbol{z} \\ &\approx \log p(\boldsymbol{y}|\boldsymbol{\alpha}, \boldsymbol{L}, \boldsymbol{\lambda}, \tilde{\boldsymbol{z}}) p(\boldsymbol{\alpha}|\tilde{\boldsymbol{z}}) p(\tilde{\boldsymbol{z}}). \quad (16)\end{aligned}$$

Equivalently, Eq. (16) can be reformulated into a minimization problem as follows:

$$\begin{aligned}\min_{\boldsymbol{\alpha},\boldsymbol{L},\boldsymbol{\lambda}} E(\boldsymbol{\alpha}, \boldsymbol{L}, \boldsymbol{\lambda}) &= \frac{1}{2}\left\|\frac{1}{\boldsymbol{\lambda}} \odot \left\{\boldsymbol{y} - \left[G(\tilde{\boldsymbol{z}}; \boldsymbol{\alpha}) * h(\boldsymbol{L})\right]\downarrow_s^d\right\}\right\|_2^2 \\ &+ \rho \sum_{k=1}^{2} |f_k * G(\tilde{\boldsymbol{z}}; \boldsymbol{\alpha})|^\gamma. \quad (17)\end{aligned}$$

To solve Eq. (17), we alternately update the model parameters $\{\boldsymbol{\alpha}, \boldsymbol{L}\}$ and $\boldsymbol{\lambda}$. Specifically, for $\boldsymbol{\alpha}$ and $\boldsymbol{L}$, they can be directly optimized by SGD based on the back-propagation (BP) algorithm [39]:

$$\boldsymbol{W}_{\text{new}} = \boldsymbol{W}_{\text{old}} - \eta \frac{\partial}{\partial \boldsymbol{W}} E(\boldsymbol{\alpha}, \boldsymbol{L}, \boldsymbol{\lambda}), \ \ \boldsymbol{W} \in \{\boldsymbol{\alpha}, \boldsymbol{L}\}, \quad (18)$$

where η is the learning rate. Actually, we adopt the more advanced Adam [21] algorithm to update $\boldsymbol{\alpha}$ and $\boldsymbol{L}$ instead of the SGD strategy of Eq. (18), which empirically makes it converge much faster.

For the noise variance $\boldsymbol{\lambda}$, we consider λ_i in the $p \times p$ patch centered at the i-th pixel. Fortunately, based on the i.i.d. Gaussian assumption within this image patch, we have the following closed-form solution for λ_i:

$$\lambda_i = \frac{1}{p^2} \sum_{j \in N(i)} \left\{y_j - \left[\left(G(\tilde{\boldsymbol{z}}; \boldsymbol{\alpha}_{\text{old}}) * h(\boldsymbol{L}_{\text{old}})\right)\downarrow_s^d\right]_j\right\}^2, \quad (19)$$

where $N(i)$ is the index set of the pixels in the $p \times p$ patch centered at i.

It should be noted that the first term of (17) can be regarded as a re-weighted L_2 loss with weight $\frac{1}{\boldsymbol{\lambda}}$, which is automatically updated through Eq. (19) during optimization. Detailed description of the proposed EM algorithm is presented in Algorithm 1.

5. Experimental Results

We conducted extensive experiments to verify the effectiveness of the proposed method in this section. For ease of presentation, we briefly denote our **b**lind **s**uper-**r**esolution method with elaborate **d**egradation **m**odeling on noise and kernel as BSRDM in the rest of this paper.

5.1. Experimental Setup

Model Settings. Throughout the experiments, we empirically set the hyper-paramters ρ and γ to be 0.2 and $2/3$, respectively. The setting on γ lies on the fact that the hyper-Laplacian with exponent $\gamma = 2/3$ is a better model of image gradient than a Laplacian or Gaussian [22]. To update the model parameters $\boldsymbol{\alpha}$ and $\boldsymbol{L}$ in M-Step, the Adam [21] algorithm with default settings in Pytorch [35] is used. The learning rates for $\boldsymbol{\alpha}$ and $\boldsymbol{L}$ are set as $2e$-3 and $5e$-3, respectively. As for the patch size p of the noise model, we provide two different settings. For the synthetic Gaussian noise in Sec. 5.2, we regard the whole image as one special image patch. While for synthetic camera sensor noise in Sec. 5.2 and real image noise in Sec. 5.3, we set p to 15. For fair comparison, the quantitative results of our method are averaged by running it five times with different random seeds.

Comparison Methods. To evaluate BSRDM, we compare it against five methods, including one learning-based method RCAN [64], and four model-based methods, namely CSC [11], ZSSR [41], DoubleDIP [36], and

Table 1. Averaged PSNR/SSIM/LPIPS results of the comparison methods under different degraded combinations on Set14. The best results are highlighted in **bold**. The gray results indicate unfair comparisons due to the mismatched degradations. Note that the results are averaged on six degradations with different blur kernels as shown in Fig. 2 on Set14.

Noise types	Scale	Metrics	Methods						
			CSC [11]	RCAN [64]	ZSSR-B [41]	ZSSR-NB [41]	DoubleDIP [36]	DIPFKP [25]	BSRDM (ours)
Case 1	×2	PSNR↑	24.87	24.99	25.04	**30.27**	23.98	27.45	29.56
		SSIM↑	0.686	0.690	0.701	**0.841**	0.637	0.752	0.815
		LPIPS↓	0.318	0.321	0.311	**0.263**	0.397	0.340	0.278
	×3	PSNR↑	21.96	22.02	22.06	26.49	20.38	26.59	**28.19**
		SSIM↑	0.551	0.553	0.566	0.741	0.498	0.712	**0.768**
		LPIPS↓	0.397	0.390	0.391	0.362	0.469	0.383	**0.328**
	×4	PSNR↑	20.18	20.08	20.23	23.73	17.98	25.66	**26.76**
		SSIM↑	0.475	0.474	0.490	0.618	0.394	0.679	**0.720**
		LPIPS↓	0.464	0.452	0.460	0.522	0.533	0.419	**0.381**
Case 2	×2	PSNR↑	24.43	24.52	24.72	26.73	23.42	26.95	**28.01**
		SSIM↑	0.648	0.651	0.671	0.723	0.618	0.734	**0.771**
		LPIPS↓	0.404	0.404	0.385	0.387	0.427	0.385	**0.359**
	×3	PSNR↑	21.70	21.73	21.81	24.74	20.03	25.31	**26.24**
		SSIM↑	0.523	0.526	0.544	0.657	0.475	0.662	**0.706**
		LPIPS↓	0.493	0.495	0.481	0.469	0.516	0.468	**0.443**
	×4	PSNR↑	20.03	19.99	19.98	23.79	18.02	24.18	**24.79**
		SSIM↑	0.454	0.454	0.475	0.619	0.376	0.608	**0.648**
		LPIPS↓	0.553	0.556	0.543	0.521	0.586	0.529	**0.507**

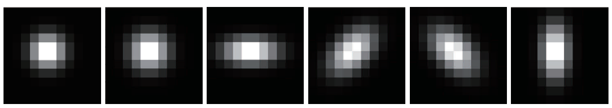


Figure 2. Six Gaussian kernels used to synthesize the LR images.

DIPFKP [25]. Specifically, RCAN is a blind SISR method trained under the bicubic degradation; CSC attempts to recover high frequency image details using convolutional sparse coding; ZSSR is a zero-shot method that exploits the patch recurrence across scales in a single image; DoubleDIP and DIPFKP are both blind SISR methods but with different kernel priors. In the synthetic experiments of Sec. 5.2, we consider both blind and non-blind settings for ZSSR, and denote them as "ZSSR-B" and "ZSSR-NB", respectively. For ZSSR-B, we use the default setting of its official code, in which the degradation model is assumed to be a bicubic downsampler followed by AWGN noise. While for ZSSR-NB, the ground truth blur kernels are pre-provided by us. Noted that ZSSR, DoubleDIP, DIPFKP, and BSRDM all employ deep CNN to generate HR image. Thus the comparison with them can better verify the marginal effects brought up by the noise and kernel modeling in BSRDM.

5.2. Evaluation on Synthetic Data

In this part, we quantitatively evaluate different methods on two commonly-used datasets, i.e., Set14 [57] and DIV2K100 [1]. DIV2K100 contains 100 high resolution images of the validation set of DIV2K, and we crop a 1024×1024 patch around the center from each image in our experiments due to GPU memory limitation. The LR images are synthesized via Eq. (4). To conduct a thorough comparison, we consider diverse degradations combined with different blur kernels and noise types. For blur kernels, two isotropic Gaussian kernels with different widths (i.e., 1.2 and 2.0) and four anisotropic Gaussian kernels are chosen as shown in Fig. 2. Furthermore, we consider two noise types as follows:

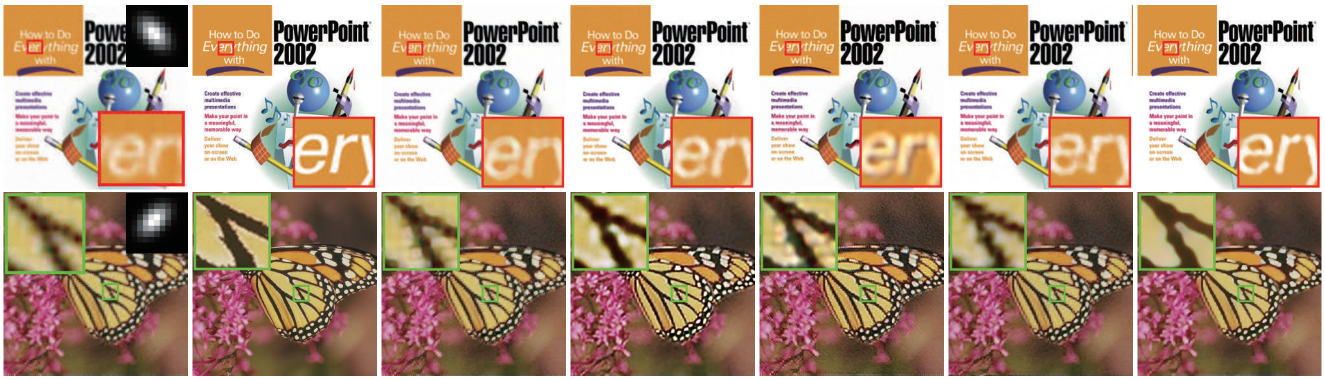
- Case 1: Gaussian noise with noise level 2.55, which is widely used in current SISR literatures [55, 58].
- Case 2: Camera sensor noise simulated by [4, 12], one typical example is shown in Fig. 3.

(a) Clean Image (b) Noisy Image (c) Noise

Figure 3. Illustration of camera sensor noise of Case 2. From left to right: (a) clean image; (b) simulated noisy image with camera sensor noise; (c) absolute residual (or noise) between (a) and (b).

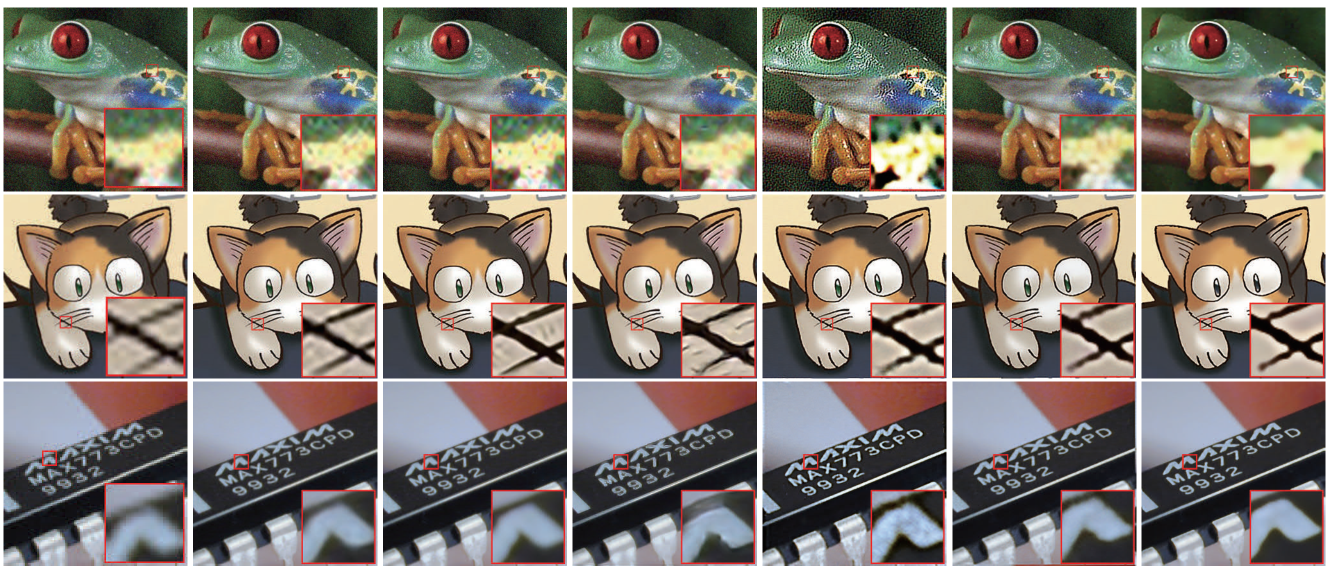
Especially, the noise in Case 2 is very close to real camera noise, and is thus suitable for evaluating different methods under the degradations with complicated real noise. As for the quantitative metrics, except for the commonly-used PSNR and SSIM [49], we also adopted LPIPS [63] to compare the perceptual similarity between the recovered HR image and ground-truth. Note that PSNR and SSIM are calculated in the luminance channel like most of the SISR literatures, while LPIPS is directly calculated in RGB channels.

Comparison with SotA Methods. Table 1 lists the PSNR, SSIM, and LPIPS results of different methods under diverse degradations on Set14. The comparison on DIV2K100 can be found in SM. From Table 1, we can see that the proposed BSRDM achieves the best or at least the second best results for all degradations. For the degradation with scale factor 2 and Gaussian noise, ZSSR-NB achieves the best performance. While for the degradation with scale factor 2 and camera sensor noise, BSRDM outperforms ZSSR-NB, indicating that BSRDM is able to handle more complicated noise due to its non-i.i.d. noise modeling. Comparing with current state-of-the-arts (SotA) method DIPFKP, the evi-



(a) Zoomed LR (x3) | (b) HR Image PSNR: top/bottom | (c) ZSSRB 17.79/21.96 | (d) DoubleDIP 18.58/23.07 | (e) ZSSRNB 23.05/25.64 | (f) DIPFKP 24.10/26.54 | (g) BSRDM (ours) 26.54/27.78

Figure 4. Super-resolution results of different methods for two degradations with Gaussian noise (top row) and camera sensor noise (bottom row) under scale factor 3 on Set14. The blur kernel is shown on the upper-right conner of the zoomed LR image.



(a) Zoomed LR (x4) | (b) CSC | (c) RCAN | (d) ZSSR | (e) DoubleDIP | (f) DIPFKP | (g) BSRDM (ours)

Figure 5. Three super-resolution results of different methods on real LR images with scale factor 4. Please zoom in for best view.

dently superiority of BSRDM demonstrates the importance of the noise and kernel modeling in SISR, since they both use the same network architecture to generate HR image even though BSRDM has fewer parameters (see Sec. 5.4).

Two visual results on Gaussian noise (top row) and camera sensor noise (bottom row) are shown in Fig. 4. Note that we only display the five best methods due to page limitation, and the complete results can be found in SM. We can easily observe that: 1) In the case of Gaussian noise, all the comparison methods can remove such simple AWGN noise. Due to the better kernel modeling, the proposed BSRDM evidently achieves sharper results. 2) Under camera sensor noise, the recovered images of the four comparison methods still contain some obvious noises or artifacts, mainly because their i.i.d. Gaussian noise assumption largely deviates from the true noise distribution. On the contrary, BSRDM is able to remove most of the noises and preserves clear image details. This demonstrates the effectiveness of the proposed non-i.i.d. noise assumption under the complicated noise.

Ablation Studies. The core contributions of this paper mainly include the non-i.i.d. noise modeling manner and the constructed kernel prior EKP. To justify their effectiveness, we design two baseline methods. In the first baseline (denoted as *Baseline1*), we replace the non-i.i.d. noise assumption with the conventional i.i.d. one. Similarly, in the second baseline (denoted as *Baseline2*), the proposed EKP kernel prior is replaced with FKP [25], which is the current most effective kernel prior to our best knowledge.

We compare BSRDM with these two baselines on different degradations that combine the six blur kernels in Fig. 2 and camera sensor noise under scale factor 2 on Set14. The detailed results are listed in Table 2. Firstly, comparing with *Baseline1*, the performance gain of BSRDM is mainly brought up by the non-i.i.d. noise assumption, which makes

Table 2. Ablation studies under camera sensor noise with scale factor 2 on Set14. The PSNR/SSIM/LPIPS results are averaged on the six kernel settings as shown in Fig. 2.

Methods	Noise Assumption		Kernel Prior		PSNR / SSIM / LPIPS
	I.i.d	Non-i.i.d.	FKP	EKP	
Baseline1	✓			✓	27.76 / 0.768 / 0.373
Baseline2		✓	✓		27.52 / 0.765 / 0.362
BSRDM (ours)		✓		✓	28.01 / 0.771 / 0.359

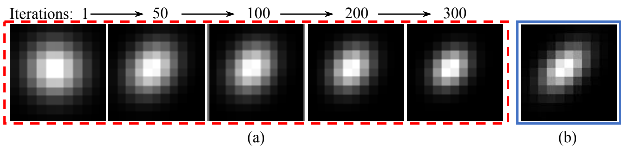


Figure 6. (a) Estimated kernels by our method at the 1st, 50th, 100th, 200th, and 300th iterations, (b) Ground-truth blur kernel.

it be able to better deal with such signal-depedent camera sensor noise. Secondly, the superiority of BSRDM over *Baseline2* indicates that our proposed kernel prior EKP is more effective than FKP as analysed in Sec. 3.2.

Figure 6 displays the estimated blur kernels by our method in different iterations during optimization. Note that the blur kernel is initialized as an isotropic Gaussian kernel with width s (i.e., the 1st iteration), where s is the scale factor. From this figure, we can see that the kernel is gradually adjusted toward the ground truth. After 300 iterations, the estimated kernel is very close to the ground truth, which facilitates a good super-resolution result.

5.3. Evaluation on Real Data

To further justify the effectiveness of BSRDM in real SISR task, we evaluate it on RealSRSet [59], which contains 20 real images from internet or the existing testing datasets [15, 29, 30, 60]. Figure 5 shows three typical examples that include different scenarios in SISR, i.e., natural image (top row), cartoon image (middle row), and text image (bottom row). It can be easily seen that the proposed BSRDM achieves evidently better visual results than the other comparison methods. In the first and second examples (bottom and middle row of Fig. 5), the LR images contain some obvious camera sensor noises or artifacts. Most of the comparison methods cannot finely deal with these cases, and tend to enlarge the noises or artifacts after super-resolution. The proposed BSRDM is able to remove most of these noises or artifacts and preserve clearer image structures due to its more robust non-i.i.d. noise modeling. For the commonly-used "chip" example (bottom row of Fig. 5) in SISR, the super-resolution results of the comparison methods are all very blurry, which may be caused by the fact that the estimated kernel does not match with the true one. On the contrary, BSRDM can obtain a relatively sharper and cleaner HR image because the proposed EKP makes it easier to estimate a rational blur kernel.

As pointed out by [59], we also find that current non-reference metrics (e.g., NIQE [33], NRQM [27], and PI [1]) are not consistent with our perceptual visual system in real SISR task. We put the detailed quantitative comparisons in terms of non-reference metrics and more visual results in SM due to page limitation.

Table 3. Comparison results of different methods on model size (K) and runtime (s).

Methods	ZSSR			DoubleDIP			DIPFKP			BSRDM		
Scale	×2	×3	×4	×2	×3	×4	×2	×3	×4	×2	×3	×4
Time (s)	56	117	235	90	194	361	91	190	333	53	108	190
# parameters (K)	225			2396			2396			762		

5.4. Comparison on Model Size and Runtime

Table 3 lists the comparison results on model size (number of parameters) and runtime with existing model-based SISR methods. For fair comparison, we consider three typical methods (i.e., ZSSR, DoubleDIP, and DIPFKP) that are all accelerated by GPU, and the runtime results in Table 3 are tested on a GeForce RTX 2080 Ti GPU. Specifically, we fix the LR image size as 256×256 and count the elapsed time of super-resolving it to size of 512×512, 768×768, and 1024×1024 with scale factor 2, 3, and 4, respectively. From Table 3, it can be easily observed that: 1) Our BSRDM has a moderate number of parameters comparing with other methods. 2) Even though BSRDM contains more parameters than ZSSR, it still has the similar speed with ZSSR. What's more, BSRDM is a little faster than ZSSR under scale factor 4. 3) Comparing with current SotA method DIPFKP, BSRDM is not only with faster speed but also much fewer parameters. Taking all of the comparisons on model size, runtime, and the performances on SISR into consideration, it should be rational to say that BSRDM is effective and potentially useful in real applications.

6. Conclusion

In this paper, we have proposed a new blind SISR method under the probabilistic framework, which elaborately considers the degradation modeling on noise and kernel. Specifically, to better fit the complicated real noise, a patch-based non-i.i.d. noise distribution is adopted in our method. As for the blur kernel, we construct an explicit yet effective kernel prior named EKP and apply it in the proposed method. Through extensive experiments, we have verifed the effectiveness and superiority of the proposed method on synthetic and real datasets. We believe that this work can benefit the blind SISR research community.

Acknowledgement. This work was partially supported by the National Key R&D Program of China (2020YFA0713900), the Hong Kong RGC GRF grant (project# 17203119), the Macao Science and Technology Development Fund under Grant 061/2020/A2, the China NSFC projects under contracts 61721002 and U1811461.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE/CF conference on computer vision and pattern recognition workshops (CVPRW)*, pages 126–135, 2017. 6, 8

[2] Siavash Arjomand Bigdeli, Matthias Zwicker, Paolo Favaro, and Meiguang Jin. Deep mean-shift priors for image restoration. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30. Curran Associates, Inc., 2017. 3

[3] Sefi Bell-Kligler, Assaf Shocher, and Michal Irani. Blind super-resolution kernel estimation using an internal-gan. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pages 284–293, 2019. 2, 3

[4] 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/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11036–11045, 2019. 6

[5] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977. 4

[6] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 184–199. Springer, 2014. 2, 3

[7] Weisheng Dong, Lei Zhang, Guangming Shi, and Xin Li. Nonlocally centralized sparse representation for image restoration. *IEEE transactions on Image Processing (TIP)*, 22(4):1620–1630, 2012. 2, 3

[8] Manuel Fritsche, Shuhang Gu, and Radu Timofte. Frequency separation for real-world super-resolution. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, pages 3599–3608. IEEE, 2019. 1, 3

[9] Daniel Glasner, Shai Bagon, and Michal Irani. Super-resolution from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 349–356. IEEE, 2009. 2

[10] Jinjin Gu, Hannan Lu, Wangmeng Zuo, and Chao Dong. Blind super-resolution with iterative kernel correction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1604–1613, 2019. 1

[11] Shuhang Gu, Wangmeng Zuo, Qi Xie, Deyu Meng, Xiangchu Feng, and Lei Zhang. Convolutional sparse coding for image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1823–1831, 2015. 1, 2, 5, 6

[12] Shi Guo, Zifei Yan, Kai Zhang, Wangmeng Zuo, and Lei Zhang. Toward convolutional blind denoising of real photographs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1712–1722, 2019. 6

[13] Xiangyu He, Zitao Mo, Peisong Wang, Yang Liu, Mingyuan Yang, and Jian Cheng. Ode-inspired network design for single image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1732–1741, 2019. 3

[14] Hsieh Hou and H Andrews. Cubic splines for image interpolation and digital filtering. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 26(6):508–517, 1978. 1, 3

[15] Andrey Ignatov, Nikolay Kobyshev, Radu Timofte, Kenneth Vanhoey, and Luc Van Gool. Dslr-quality photos on mobile devices with deep convolutional networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3277–3285, 2017. 8

[16] Xiaozhong Ji, Yun Cao, Ying Tai, Chengjie Wang, Jilin Li, and Feiyue Huang. Real-world super-resolution via kernel estimation and noise injection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 466–467, 2020. 1, 3

[17] Meiguang Jin, Stefan Roth, and Paolo Favaro. Normalized blind deconvolution. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 668–684, 2018. 2, 3

[18] Robert Keys. Cubic convolution interpolation for digital image processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(6):1153–1160, 1981. 3

[19] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1646–1654, 2016. 2, 3

[20] 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 (TPAMI)*, 32(6):1127–1133, 2010. 1, 2, 3

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

[22] Dilip Krishnan and Rob Fergus. Fast image deconvolution using hyper-laplacian priors. In *Advances in Neural Information Processing Systems*, volume 22, pages 1033–1041, 2009. 2, 3, 5

[23] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 624–632, 2017. 2

[24] Xin Li and Michael T Orchard. New edge-directed interpolation. *IEEE Transactions on Image Processing (TIP)*, 10(10):1521–1527, 2001. 1, 3

[25] Jingyun Liang, Kai Zhang, Shuhang Gu, Luc Van Gool, and Radu Timofte. Flow-based kernel prior with application to blind super-resolution. pages 10601–10610, 2021. 1, 2, 3, 4, 6, 7

[26] Jie Liu, Wenjie Zhang, Yuting Tang, Jie Tang, and Gangshan Wu. Residual feature aggregation network for image super-resolution. In *Proceedings of the IEEE/CVF Confer-*

ence on Computer Vision and Pattern Recognition (CVPR), pages 2359–2368, 2020. 3

[27] Chao Ma, Chih Yuan Yang, Xiaokang Yang, and Ming Hsuan Yang. Learning a no-reference quality metric for single-image super-resolution. *Computer Vision and Image Understanding*, 158:1–16, 2017. 8

[28] Shunta Maeda. Unpaired image super-resolution using pseudo-supervision. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 291–300, 2020. 1, 3

[29] 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 Eighth IEEE International Conference on Computer Vision (ICCV)*, volume 2, pages 416–423. IEEE, 2001. 8

[30] Yusuke Matsui, Kota Ito, Yuji Aramaki, Azuma Fujimoto, Toru Ogawa, Toshihiko Yamasaki, and Kiyoharu Aizawa. Sketch-based manga retrieval using manga109 dataset. *Multimedia Tools and Applications*, 76(20):21811–21838, 2017. 8

[31] Yiqun Mei, Yuchen Fan, Yuqian Zhou, Lichao Huang, Thomas S Huang, and Honghui Shi. Image super-resolution with cross-scale non-local attention and exhaustive self-exemplars mining. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5690–5699, 2020. 3

[32] Tomer Michaeli and Michal Irani. Nonparametric blind super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 945–952, 2013. 2

[33] Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. Making a "completely blind" image quality analyzer. *IEEE Signal processing letters*, 20(3):209–212, 2012. 8

[34] Xingang Pan, Xiaohang Zhan, Bo Dai, Dahua Lin, Chen Change Loy, and Ping Luo. Exploiting deep generative prior for versatile image restoration and manipulation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 262–277. Springer, 2020. 2, 3

[35] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in neural information processing systems*, volume 32, pages 8026–8037, 2019. 5

[36] Dongwei Ren, Kai Zhang, Qilong Wang, Qinghua Hu, and Wangmeng Zuo. Neural blind deconvolution using deep priors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3341–3350, 2020. 1, 3, 5, 6

[37] Gernot Riegler, Samuel Schulter, Matthias Ruther, and Horst Bischof. Conditioned regression models for non-blind single image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 522–530, 2015. 2, 3

[38] 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. 2, 3

[39] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986. 5

[40] Marshall F Tappen Bryan C Russell and William T Freeman. Exploiting the sparse derivative prior for super-resolution and image demosaicing. In *Procedings of Third International Workshop Statistical and Computational Theories of Vision (SCTV)*, 2003. 1

[41] Assaf Shocher, Nadav Cohen, and Michal Irani. "zero-shot" super-resolution using deep internal learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3118–3126, 2018. 2, 3, 5, 6

[42] Jian Sun, Zongben Xu, and Heung-Yeung Shum. Image super-resolution using gradient profile prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. IEEE, 2008. 1, 2, 3

[43] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4539–4547, 2017. 3

[44] Philippe Thévenaz, Thierry Blu, and Michael Unser. Image interpolation and resampling. *Handbook of Medical Imaging, Processing and Analysis*, 1(1):393–420, 2000. 1, 3

[45] Radu Timofte, Vincent De Smet, and Luc Van Gool. A+: Adjusted anchored neighborhood regression for fast super-resolution. In *Asian Conference on Computer Vision (ACCV)*, pages 111–126. Springer, 2014. 1

[46] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Deep image prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9446–9454, 2018. 2, 3, 4

[47] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1905–1914, 2021. 3

[48] 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/CVF conference on Computer Vision and Pattern Recognition (CVPR)*, pages 606–615, 2018. 3

[49] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing (TIP)*, 13(4):600–612, 2004. 6

[50] Max Welling and Yee W Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 681–688. Citeseer, 2011. 5

[51] Valentin Wolf, Andreas Lugmayr, Martin Danelljan, Luc Van Gool, and Radu Timofte. Deflow: Learning complex image degradations from unpaired data with conditional flows. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 94–103, 2021. 1, 3

[52] Jianwen Xie, Ruiqi Gao, Zilong Zheng, Song-Chun Zhu, and Ying Nian Wu. Learning dynamic generator model by alternating back-propagation through time. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 33, pages 5498–5507, 2019. 4

[53] Jianwen Xie, Ruiqi Gao, Zilong Zheng, Song-Chun Zhu, and Ying Nian Wu. Motion-based generator model: Unsupervised disentanglement of appearance, trackable and intrackable motions in dynamic patterns. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, volume 34, pages 12442–12451, 2020. 4

[54] Yuan Yuan, Siyuan Liu, Jiawei Zhang, Yongbing Zhang, Chao Dong, and Liang Lin. Unsupervised image super-resolution using cycle-in-cycle generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 701–710, 2018. 1, 3

[55] Zongsheng Yue, Hongwei Yong, Qian Zhao, Deyu Meng, and Lei Zhang. Variational denoising network: Toward blind noise modeling and removal. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pages 1690–1701, 2019. 6

[56] Zongsheng Yue, Hongwei Yong, Qian Zhao, Lei Zhang, and Deyu Meng. Variational image restoration network. *arXiv preprint arXiv:2008.10796*, 2020. 1

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

[58] Kai Zhang, Luc Van Gool, and Radu Timofte. Deep unfolding network for image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3217–3226, 2020. 2, 3, 6

[59] Kai Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. *arXiv preprint arXiv:2103.14006*, 2021. 2, 3, 8

[60] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Ffdnet: Toward a fast and flexible solution for cnn-based image denoising. *IEEE Transactions on Image Processing (TIP)*, 27(9):4608–4622, 2018. 8

[61] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Learning a single convolutional super-resolution network for multiple degradations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3262–3271, 2018. 2

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

[63] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pages 586–595, 2018. 6

[64] 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, 3, 5, 6

[65] zhengxiong luo, Yan Huang, Shang Li, Liang Wang, and Tieniu Tan. Unfolding the alternating optimization for blind super resolution. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 5632–5643, 2020. 1, 2