# Uncertainty Learning in Kernel Estimation for Multi-Stage Blind Image Super-Resolution

Zhenxuan Fang[1], Weisheng Dong[1](✉), Xin Li[2], Jinjian Wu[1], Leida Li[1], and Guangming Shi[1]

[1] School of Artificial Intelligence, Xidian University, Xi'an, China
zxfang@stu.xidian.edu.cn, {wsdong, jinjian.wu}@mail.xidian.edu.cn
{ldli, gmshi}@xidian.edu.cn
[2] Lane Dep. of CSEE, West Virginia University, Morgantown WV, USA
xin.li@mail.wvu.edu

**Abstract.** Conventional wisdom in blind super-resolution (SR) first estimates the unknown degradation from the low-resolution image and then exploits the degradation information for image reconstruction. Such sequential approaches suffer from two fundamental weaknesses - i.e., the lack of robustness (the performance drops when the estimated degradation is inaccurate) and the lack of transparency (network architectures are heuristic without incorporating domain knowledge). To address these issues, we propose a joint Maximum a Posteriori (MAP) approach for estimating the unknown kernel and high-resolution image simultaneously. Our method first introduces uncertainty learning in the latent space when estimating the blur kernel, aiming at improving the robustness to the estimation error. Then we propose a novel SR network by unfolding the joint MAP estimator with a learned Laplacian Scale Mixture (LSM) prior and the estimated kernel. We have also developed a novel approach of estimating both the scale prior coefficient and the local means of the LSM model through a deep convolutional neural network (DCNN). All parameters of the MAP estimation algorithm and the DCNN parameters are jointly optimized through end-to-end training. Extensive experiments on both synthetic and real-world images show that our method achieves state-of-the-art performance for the task of blind image SR.

## 1 Introduction

Single image super-resolution (SISR) is a typical low-level vision problem that aims to reconstruct the high-resolution (HR) image from its low-resolution (LR) observation. Since the pioneering work of applying convolutional neural networks to SR (SRCNN) [10], extensive deep learning-based methods [11,17,26,30,34,38, 52,53] have been developed and achieved impressive performance. Most existing methods are based on the assumption that the degradation is known and predefined (e.g., bicubic downsampling), so numerous training data can be manually synthesized and used to train powerful networks. However, these methods will suffer a dramatic performance drop when the degradation involved in the test image is different from the assumption. To tackle this problem, several methods

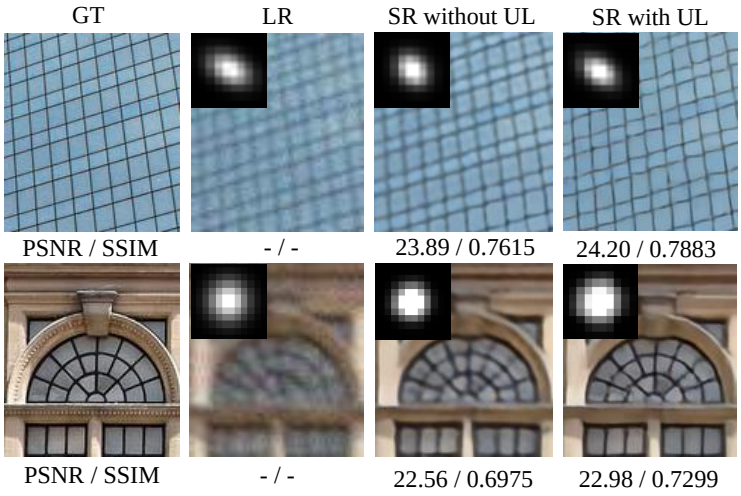


**Fig. 1.** SR results produced by the multi-stage network, the corresponding estimated blur kernels with or without uncertainty learning (UL) are illustrated on the top left.

have been proposed [50, 51] by taking the blur kernel as an additional input of the network to utilize the degradation prior knowledge. Recently, [42, 43] only focused on exploiting the internal information of the test images, also known as zero-shot SR. However, the above methods need to provide the blur kernels of test images, so kernel estimation [4, 37] is crucial before these non-blind SR methods, but if the estimated kernel deviates from the ground truth, the kernel mismatch will lead to undesired artifacts [15].

Generally, the degradation processes of real-world LR images are probably complicated and unknown [7, 15], so studying the problem of blind SR is particularly valuable. Most early blind SR methods are model-based [3, 18, 20, 47], they exploit internal self-similarity and edge prior to estimate the underlying blur kernels of LR images before performing SR. But their optimization procedures are usually time-consuming due to complex and iterative computation. Deep learning-based iterative kernel correction (IKC) [15] uses the SR result to correct the estimated kernel in an iterative manner, where the estimation is integrated into reconstruction by spatial feature transform (SFT) layers [48]. An unsupervised degradation representation learning scheme was proposed in [46] by contrastive learning based on the assumption that the degradation is invariant within the same image but varies from image to image.

However, these methods have several obvious limitations. First, accurate estimation is often impossible - due to the ill-posed nature of inverse problems, there exist multiple candidates of the kernel for a single LR input [15]. Meanwhile, the estimated kernel is sensitive to the input noise, leading to inaccurate estimation results. Second, the fusion modules such as [51] will face the domain interference problem because it directly concatenates the degradation represen-

tations with image features [46]. Meanwhile, the SR networks are designed based on the black-box principle, making it difficult for interpretation or optimization.

The motivation behind this work is twofold. On the one hand, we advocate a joint optimization of kernel estimation and image reconstruction. Such end-to-end training is desirable to alleviate the catastrophic error propagation in sequential approaches. On the other hand, it is desirable to quantify the uncertainty of kernel estimation so that we can incorporate such ambiguity into the process of image reconstruction. In this paper, we first introduce data uncertainty learning with kernel estimation. Instead of using fixed feature maps in the estimation network, the feature (mean) and uncertainty (variance) are learned simultaneously. Then we propose a transparent blind SR method with learned Laplacian Scale Mixture (LSM) prior. The contributions of this paper are listed as follows.

- The blind SR problem is formulated as a joint Maximum a Posteriori (MAP) approach for estimating the blur kernel and reconstructing the HR image. Then we propose a novel multi-stage SR network by converting the MAP estimator with a learned LSM prior and estimated kernel into a multi-stage deep network, all parameters in the MAP estimator are optimized in an end-to-end manner.
- To improve the performance and robustness of kernel estimation, we introduce uncertainty learning to the kernel estimation network. Both the feature (mean) and uncertainty (variance) in the latent space of the blur kernel are learned, which is proved can produce more accurate kernel than deterministic model.
- Extensive experimental results on both synthetic and real-world datasets show that the proposed method outperforms existing state-of-the-art blind SR methods, especially in the presence of heavy noise contamination. Subjective evaluation of SR images is also convincingly in favor of our method.

## 2 Related Work

### 2.1 Blind Image Super-Resolution

Blind SR assumes that the blur kernels of test images are unknown. Previous model-based methods [20,21] are time-consuming because most of them involve complicated optimization procedures. In [37], an optimal kernel can be recovered by utilizing the internal patch recurrence property in an image. With the development of deep learning, CNN-based blind SR methods become more popular [15,24,27,32,33,35,44,46]. IKC method [15] performs blind SR by using the intermediate reconstruction results to iteratively correct the estimation of blur kernels. Luo *et al.* [35] proposed a deep alternating network (DAN) by concatenating the estimator and restorer module alternately. By utilizing the degenerative similarity of small patches in an image, [46] introduces an unsupervised contrastive learning scheme to extract various degradation representations for further reconstruction. Recently, [27] proposed a blind SR framework based on

kernel-oriented adaptive local adjustment of SR features. MANet [32] proposes a kernel estimation framework using the mutual affine convolution layer.

### 2.2 Uncertainty in Deep Learning

The uncertainty in deep learning can be divided into two categories [9]: epistemic/model uncertainty and aleatoric/data uncertainty. The former describes how much the model is uncertain about its predictions. The latter refers to the noise inherent in the observation data. Many works [2,6,16,25] have been studied to model the uncertainty in deep learning tasks, including image classification, image segmentation, and face recognition. By introducing uncertainty, they have improved the performance and robustness of deep networks. GAMA [31] analyses the effect of aleatoric/data uncertainty on SISR reconstruction by decreasing the loss attenuation of large variance pixels. Recently, [39] proposed a novel uncertainty-driven loss (UDL) to enforce the network concentrating more on the pixels with large variance, which is beneficial for better reconstruction of texture and edge regions.

### 2.3 LSM Model for Image Restoration

As a probability model, the Laplacian Scale Mixture (LSM) model is an analogy to the classical Gaussian scale mixture model, which has been used for various image restoration tasks [23,40,41]. The early work [14] proposed a class of sparse coding models that utilizes a LSM prior to model dependencies among coefficients. In [22], the LSM distribution has also been used to model impulse noise and remove mixture noise effectively. [12] propose a novel robust tensor approximation framework for the LSM modeling of three-dimensional data. Different from the existing LSM model for image restoration with manually selected scale priors, we use the DCNNs to learn both the scale prior and local means in the LSM model. Through end-to-end training, all parameters are learned jointly.

## 3 Method

### 3.1 Problem Formulation

The widely accepted degradation model assumes that the LR image is produced by downsampling HR image after the convolution with blur kernel, which can be mathematically expressed as $\boldsymbol{y} = (\boldsymbol{x} * \boldsymbol{k}) \downarrow_s + \boldsymbol{n}$, where $\boldsymbol{x}$ is the original HR image, $\boldsymbol{y}$ is the degraded LR image, $*$ denotes the convolution with blur kernel $\boldsymbol{k}$, $\downarrow_s$ is the s-fold downsampling operation and $\boldsymbol{n}$ denotes the additional noise. The matrix-vector form can be formulated as

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

where $\mathbf{A} = \mathbf{DH}$ denotes the degradation operator ($\mathbf{H}$ is the blur matrix constructed from the kernel $\boldsymbol{k}$ and $\mathbf{D}$ is the downsampling matrix). Then blind SR

refers to the process of estimating $\mathbf{H}$ and recovering $\boldsymbol{x}$ from $\boldsymbol{y}$ and $\mathbf{H}$, which is a highly ill-posed inverse problem. We formulate it as a maximum a posteriori (MAP) estimation problem

$$p(\mathbf{H}, \boldsymbol{x}|\boldsymbol{y}) = p(\mathbf{H}|\boldsymbol{y})\ p(\boldsymbol{x}|\mathbf{H}, \boldsymbol{y}). \tag{2}$$

Take logarithms on both sides of the equation

$$\log p(\mathbf{H}, \boldsymbol{x}|\boldsymbol{y}) \propto \log p(\mathbf{H}|\boldsymbol{y}) + \log p(\boldsymbol{y}|\mathbf{H}, \boldsymbol{x}) + \log p(\boldsymbol{x}). \tag{3}$$

Then solving the MAP problem can be expressed as

$$(\mathbf{H}^*, \boldsymbol{x}^*) = \underset{\mathbf{H}, \boldsymbol{x}}{\operatorname{argmax}} \log p(\mathbf{H}|\boldsymbol{y}) + \log p(\boldsymbol{y}|\mathbf{H}, \boldsymbol{x}) + \log p(\boldsymbol{x}). \tag{4}$$

The above optimization problem can be converted into two subproblems

$$\mathbf{H}^* = \underset{\mathbf{H}}{\operatorname{argmax}} \log p(\mathbf{H}|\boldsymbol{y}), \tag{5a}$$

$$\boldsymbol{x}^* = \underset{\boldsymbol{x}}{\operatorname{argmax}} \log p(\boldsymbol{y}|\mathbf{H}, \boldsymbol{x}) + \log p(\boldsymbol{x}). \tag{5b}$$

Their specific meanings are clear: Eq. (5a) denotes the estimation of blur kernel and Eq. (5b) denotes reconstructing HR image from LR image and the estimated kernel.

### 3.2 Uncertainty Learning in Kernel Estimation

For the estimation of blur kernel, there exist some inevitable errors in the prediction results due to noise interference and the ill-posed nature. To properly take the uncertainty of the prediction into account, we introduce uncertainty learning (UL) to the process of blur kernel estimation. For the likelihood term $p(\mathbf{H}|\boldsymbol{y})$ in Eq. (5a), we propose to model it by the following Gaussian distribution,

$$p(\mathbf{H}|\boldsymbol{y}) \sim \mathcal{N}(\boldsymbol{k}|\mu(\boldsymbol{y}), \sigma^2(\boldsymbol{y})), \tag{6}$$

where $\mu(\boldsymbol{y})$ and $\sigma^2(\boldsymbol{y})$ denote the mappings from $\boldsymbol{y}$ to the posterior distribution parameters ($\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$) of $\boldsymbol{k}$. However, it is difficult to calculate the mappings explicitly. As shown in Fig. 2(a), we parameterize the two mappings into deep networks, - i.e. $\boldsymbol{\mu} = f_{\boldsymbol{\Theta}_1}(\boldsymbol{y}), \boldsymbol{\sigma} = f_{\boldsymbol{\Theta}_2}(\boldsymbol{y})$, where $\boldsymbol{\Theta_1}$ and $\boldsymbol{\Theta_2}$ represent the parameters of mean and variance branches respectively. Specifically, the LR image $\boldsymbol{y}$ is input into a DCNN to extract the feature maps of the underlying blur kernel. Then the features go through two $3 \times 3$ convolution layers to learn the mean and variance of prediction result simultaneously. From another perspective, $\boldsymbol{\mu}$ can be interpreted as the identity mapping of the blur kernel and $\boldsymbol{\sigma}$ is the uncertainty of the predicted $\boldsymbol{\mu}$. Then we generate an equivalent sampling representation $\boldsymbol{z}$ through re-parameterization method [29]

$$\boldsymbol{z} = \boldsymbol{\mu} + \epsilon\boldsymbol{\sigma}, \quad \epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \tag{7}$$

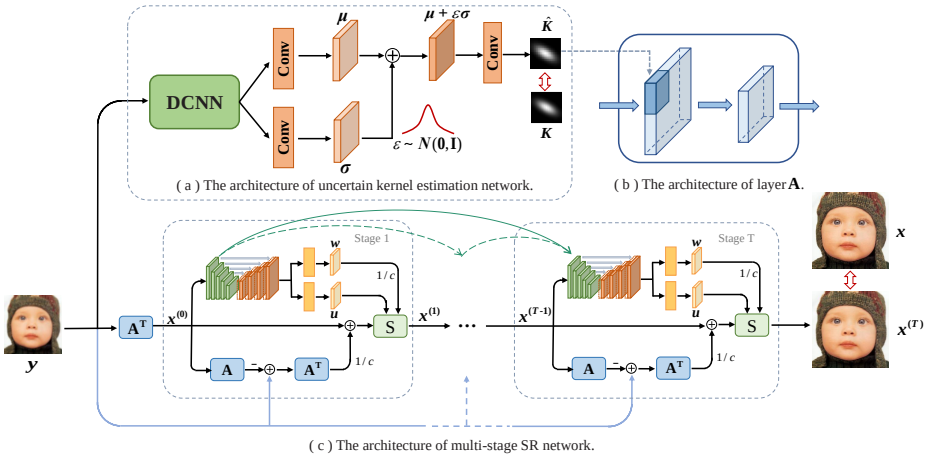


**Fig. 2.** Overview of the proposed KULNet for blind SR. The architectures of (a) the uncertain kernel estimation network, (b) the layer $\mathbf{A}$, which contains a convolution layer with the estimated kernel and a downsampling layer, (c) the multi-stage SR network.

where $\epsilon$ denotes a random noise sampled from the normal distribution. Since $\boldsymbol{\mu}$ is corrupted by $\boldsymbol{\sigma}$ during the training period, $\boldsymbol{z}$ is not a deterministic point embedding anymore. However, we notice that the model tends to predict small $\boldsymbol{\sigma}$ for all samples to suppress the instable components if there are no constraints on the embeddings. Similar to [6], we adopt the Kullback-Leibler (KL) divergence regularization term to enforce $\mathcal{N}\left(\boldsymbol{\mu}, \boldsymbol{\sigma}^2\right)$ to be close to the standard normal distribution $\mathcal{N}(\mathbf{0}, \mathbf{I})$,

$$\begin{aligned} \mathcal{L}_{kl} &= KL\left[\mathcal{N}\left(\boldsymbol{\mu}, \boldsymbol{\sigma}^2\right) \| \mathcal{N}(\mathbf{0}, \mathbf{I})\right] \\ &= -\frac{1}{2}\left(1 + \log \boldsymbol{\sigma}^2 - \boldsymbol{\mu}^2 - \boldsymbol{\sigma}^2\right). \end{aligned} \tag{8}$$

Then the sampled embedding $\boldsymbol{z}$ is input to the final convolution layer to obtain the kernel estimation.

### 3.3 Multi-Stage SR Network

***LSM model for SR.*** To solve Eq. (5b), we note that $p(\boldsymbol{y}|\mathbf{H}, \boldsymbol{x})$ is the likelihood term and $p(\boldsymbol{x})$ is the prior distribution of $\boldsymbol{x}$. The likelihood term can be generally modeled by a Gaussian distribution

$$p(\boldsymbol{y}|\mathbf{H}, \boldsymbol{x}) = \frac{1}{\sqrt{2\pi}\sigma_n} \exp\left(-\frac{\|\boldsymbol{y} - \mathbf{D}\mathbf{H}\boldsymbol{x}\|_2^2}{2\sigma_n^2}\right). \tag{9}$$

For the prior term $p(\boldsymbol{x})$ of the HR image, we propose to characterize each pixel $x_i$ with a *nonzero-mean* Laplacian distribution of variance $2\theta_i^2$ and mean $u_i$

$$p\left(x_i|\theta_i\right) = \frac{1}{2\theta_i} \exp\left(-\frac{|x_i - u_i|}{\theta_i}\right). \tag{10}$$

With the assumption that $x_i$ and $\theta_i$ are independent, we can model $\boldsymbol{x}$ with the following LSM model

$$p(\boldsymbol{x}) = \prod_i p\left(x_i\right), \; p\left(x_i\right) = \int_0^{\infty} p\left(x_i|\theta_i\right) p\left(\theta_i\right) d\theta_i, \tag{11}$$

where the scale prior $p(\theta_i)$ can be modeled by a general energy function - e.g., $p\left(\theta_i\right) \propto \exp\left(-J\left(\theta_i\right)\right)$. Then Eq. (5b) is equivalent to a bivariate estimation problem - i.e.,

$$(\boldsymbol{x}^*, \boldsymbol{\theta}^*) = \underset{\boldsymbol{x},\boldsymbol{\theta}}{\operatorname{argmax}} \log p(\mathbf{H}, \boldsymbol{y}|\boldsymbol{x}) + \log p(\boldsymbol{x}|\boldsymbol{\theta}) + \log p(\boldsymbol{\theta}). \tag{12}$$

By substituting the Gaussian likelihood term of Eq. (9), the prior terms of Eq. (10) into the MAP estimator Eq. (12), we can obtain the following objective function

$$(\boldsymbol{x}^*, \boldsymbol{\theta}^*) = \underset{\boldsymbol{x},\boldsymbol{\theta}}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{y} - \mathbf{D}\mathbf{H}\boldsymbol{x}\|_2^2 + \sum_{i=1}^{N} \frac{\sigma_n^2}{\theta_i}|x_i - u_i| + \boldsymbol{\Omega}(\boldsymbol{\theta}), \tag{13}$$

where $\boldsymbol{\Omega}(\boldsymbol{\theta}) = \sigma_n^2 \sum_{i=1}^{N} \log \theta_i + \sigma_n^2 J(\boldsymbol{\theta})$, then the SR problem can be solved by alternating optimizing $\boldsymbol{x}$ and $\boldsymbol{\theta}$. For the $\boldsymbol{x}$-subproblem, with fixed $\boldsymbol{\theta}$, we can solve $\boldsymbol{x}$ by

$$\boldsymbol{x}^* = \underset{\boldsymbol{x}}{\operatorname{argmin}} \frac{1}{2}\|\boldsymbol{y} - \mathbf{D}\mathbf{H}\boldsymbol{x}\|_2^2 + \sum_{i=1}^{N} w_i|x_i - u_i|, \tag{14}$$

where $w_i = \sigma_n^2/\theta_i$. Inspired by recent advances in image denoising [13,38], the mean $u_i$ can be predicted by a deep denoising module, i.e. $u_i = f(x_i)$, where $f(\cdot)$ denotes a denoiser. Then Eq. (14) can be solved by the iterative shrinkage-thresholding algorithm [8] as

$$\boldsymbol{x}^{(t+1)} = \mathcal{S}_{\boldsymbol{\tau}^{(t)},\boldsymbol{u}^{(t)}}\left(\boldsymbol{x}^{(t)} + \frac{1}{c}\mathbf{A}^{\top}\left(\boldsymbol{y} - \mathbf{A}\boldsymbol{x}^{(t)}\right)\right), \tag{15}$$

where $\mathbf{A} = \mathbf{D}\mathbf{H}$, $\mathbf{A}^{\top} = \mathbf{H}^{\top}\mathbf{D}^{\top}$ and $c$ is chosen to ensure convergence. $\mathcal{S}_{\boldsymbol{\tau}^{(t)},\boldsymbol{u}^{(t)}}(\cdot)$ denotes a generalized shrinkage operator with threshold $\boldsymbol{\tau}^{(t)} = \frac{\boldsymbol{w}^{(t)}}{c}$ and $\boldsymbol{u}^{(t)}$, which is defined by

$$\mathcal{S}_{\boldsymbol{\tau},\boldsymbol{u}}(\boldsymbol{t}) = \begin{cases} \boldsymbol{t} + \boldsymbol{\tau}, & \boldsymbol{t} < \boldsymbol{u} - \boldsymbol{\tau} \\ \boldsymbol{u}, & \boldsymbol{u} - \boldsymbol{\tau} \leq \boldsymbol{t} \leq \boldsymbol{u} + \boldsymbol{\tau} \\ \boldsymbol{t} - \boldsymbol{\tau}, & \boldsymbol{t} > \boldsymbol{u} + \boldsymbol{\tau} \end{cases} \tag{16}$$

Similarly, the $\boldsymbol{\theta}$-subproblem is equivalent to solve the $\boldsymbol{w}$-subproblem. With a fixed $\boldsymbol{x}$, we have

$$\boldsymbol{w}^* = \underset{\boldsymbol{w}}{\operatorname{argmin}} \sum_{i=1}^{N} w_i|x_i - u_i| + \boldsymbol{\Omega}(\boldsymbol{w}). \tag{17}$$

Iterative algorithms [40] can be used to solve $\boldsymbol{w}$, which depends on a hand-crafted prior $p(\boldsymbol{\theta})$ in $\boldsymbol{\Omega}(\boldsymbol{w})$ - e.g., Jeffrey's prior. Instead of using a fixed prior, we propose to estimate $\boldsymbol{w}^{(t)}$ from $\boldsymbol{x}^{(t)}$ using a universal DCNN-based denoiser [13,38].

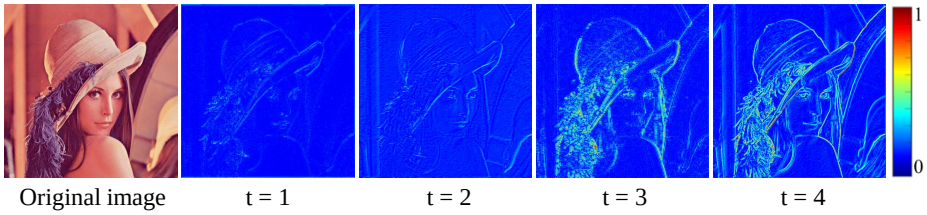


**Fig. 3.** The visualization of the learned regularization parameter $\boldsymbol{w}$ estimated in 4 stages.

***Multi-stage network for SR.*** Despite the theoretical rigor, alternatively solving $\boldsymbol{x}$ and $\boldsymbol{w}$ requires many iterations to converge and need a hand-crafted prior $p(\boldsymbol{\theta})$. Meanwhile, all parameters and the denoiser can not be jointly optimized. To address these issues, we replace all variables in Eq. (15) with a common expression containing $\boldsymbol{x}$, so that $\boldsymbol{x}$ and $\boldsymbol{w}$ can be jointly optimized in a unified framework.

$$\boldsymbol{x}^{(t+1)} = \mathcal{S}_{\frac{\mathcal{G}_{\boldsymbol{w}}(\boldsymbol{x}^{(t)})}{c}, \mathcal{G}_{\boldsymbol{u}}(\boldsymbol{x}^{(t)})} \left( \boldsymbol{x}^{(t)} + \frac{1}{c}\mathbf{A}^{\top}\left(\boldsymbol{y} - \mathbf{A}\boldsymbol{x}^{(t)}\right)\right), \tag{18}$$

where $\mathcal{G}_{\boldsymbol{w}}(\cdot)$ denotes the CNN generator for estimating $\boldsymbol{w}$, and the mean $\boldsymbol{u}$ is also predicted by a generator - i.e., $\boldsymbol{u}^{(t)} = \mathcal{G}_{\boldsymbol{u}}(\boldsymbol{x}^{(t)})$. Note that the blur kernel $\mathbf{H}$ in $\mathbf{A}$ has been estimated from $\boldsymbol{y}$ by our uncertain kernel estimation network. Similar to [13], we can unfold the iterative optimization in Eq. (18) into a multi-stage network implementation.

***Network architecture.*** The architecture of the proposed multi-stage SR network is shown in Fig. 2(c). All modules in the network strictly correspond to the steps in the optimization process, the network executes $T$ iterations of Eq. (18). The input LR image $\boldsymbol{y} \in \mathbb{R}^{C\times H\times W}$ first goes through a convolution (Conv) layer parameterized by the degradation matrix $\mathbf{A}^{\top}$ for an initial estimate $\boldsymbol{x}^{(0)} \in \mathbb{R}^{C\times sH\times sW}$, $s$ denotes the scale factor. For the upper branch, $\boldsymbol{x}^{(t)}$ is fed to a U-Net followed by two generators to estimate the weight $\boldsymbol{w}^{(t)}$ and mean $\boldsymbol{u}^{(t)}$. The lightweight U-net consists of five encoding blocks (EBs) and four decoding blocks (DBs), each EB and DB contain two Conv layers with ReLU activation function. The average pooling and bilinear interpolation layer are used to downsample and upsample the feature maps. The channel number of the output features in 5 EBs and 4 DBs are set to 32, 64, 64, 128, 128, 128, 64, 64, and 32, respectively. The weight and mean generator both contain three Conv layers. The estimated weight $\boldsymbol{w}$ in each stage are visualized (with normalization) in Fig. 3, we can see that $\boldsymbol{w}$ is sparse and helps the network concentrate more and more on high-frequency edges and textures.

To leverage information of multiple stages, we use long connections to concatenate previous features with current features, leading to more faithful reconstruction of the missing high-frequency information. As illustrated in Fig. 2(b),

the layers $\mathbf{A}$ and $\mathbf{A}^\top$ are designed for a specific blur kernel $\mathbf{H}$ obtained by our uncertain kernel estimation network. For Gaussian degradation, $\mathbf{A} = \mathbf{DH}$, where $\mathbf{H}$ and $\mathbf{D}$ denote the Gaussian blur matrix and the downsampling matrix respectively. In layer $\mathbf{A}$, the input feature maps are convoluted with the estimated blur kernel $\mathbf{H}$ and then downsampled via bicubic interpolation. Similarly, the layer $\mathbf{A}^\top = \mathbf{H}^\top\mathbf{D}^\top$ corresponds to first upsample the LR image and then put the upsampled image into a transpose convolution layer with the blur kernel. Module $\boldsymbol{S}$ denotes a shrinkage operator with threshold $\boldsymbol{w}$ and $\boldsymbol{u}$.

### 3.4 Network Training

We combine the above uncertain kernel estimation network and multi-stage SR network into a whole training framework, called Kernel Uncertianty Learning network (KULNet). For the kernel estimation network, we use a combination loss of the $\mathcal{L}_1$ loss between the estimated kernel $\hat{\boldsymbol{K}}$ and the GT kernel $\boldsymbol{K}$ ($\mathcal{L}_e = \frac{1}{m}\sum_{i=1}^{m}\|\hat{\boldsymbol{K}}_i - \boldsymbol{K}_i\|_1$) and the $KL$ loss in Eq. (8), denoted by $\mathcal{L}_K = \mathcal{L}_e + \lambda\mathcal{L}_{kl}$, where $\lambda$ is set to be 0.001. For the SR network, all parameters of each stage are shared except $c$. The $\mathcal{L}_1$ loss function is adopted to train the proposed deep network, written as

$$\mathcal{L}_1 = \frac{1}{m}\sum_{i=1}^{m}\left\|\mathcal{F}\left(\boldsymbol{y}_i\right) - \boldsymbol{x}_i\right\|_1, \tag{19}$$

where $m$ denotes the total number of the training samples, $\boldsymbol{y}_i$ and $\boldsymbol{x}_i$ denote the $i$-th pair of LR and HR image patches, and $\mathcal{F}(\boldsymbol{y}_i)$ denotes the SR image by the network. The total loss is described as $\mathcal{L}_{total} = \mathcal{L}_K + \mathcal{L}_1$. The ADAM optimizer [28] is used to train the network with setting $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. The learning rate is set as $2{\times}10^{-4}$. The parameters of the convolutional layers are initialized by the Xavier initialization [19]. We implement the proposed method by PyTorch and train the network using an Nvidia RTX 2080Ti GPU.

## 4 Experimental Results

### 4.1 Datasets and Settings

Following [15, 46], the training set consists of 3450 HR images, including 800 images in DIV2K [1] and 2650 images in Flickr2K [45]. The LR training patches are generated by Gaussian blur and bicubic downsampling with a size of $48 \times 48$. Four standard benchmark datasets: Set5 [5], Set14 [49], BSD100 [36] and Urban100 [21] are used for testing. We train our network on the general degradation with anisotropic Gaussian kernels and noises, the Gaussian kernels are generated by randomly selecting the kernel width determined by a diagonal covariance matrix with $\sigma_1, \sigma_2 \sim U(0.2, 4)$ and a random rotation angle $\theta \sim U(0, \pi)$, the range of noise level is set to $[0, 25]$. The kernel size is fixed to $21 \times 21$. For evaluation, 9 typical blur kernels in [46] and different noise levels are used to generate the test images. Performances in terms of PSNR and SSIM metrics are conducted on the Y channel of YCbCr space.

**Fig. 4.** The visualization of anisotropic blur kernels used for testing.

**Table 1.** MAE (↓) results of the estimated kernels by kernel estimation network with or without Uncertainty Learning (UL).

| Kernel estimation | Noise | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|
| w/o UL | 0 | 0.252 | 0.174 | 0.204 | 0.167 | 0.151 | 0.226 | 0.181 | 0.157 | 0.195 | 0.190 |
| | 10 | 0.259 | 0.183 | 0.218 | 0.186 | 0.171 | 0.240 | 0.201 | 0.183 | 0.216 | 0.206 |
| | 20 | 0.269 | 0.202 | 0.251 | 0.235 | 0.223 | 0.271 | 0.251 | 0.267 | 0.302 | 0.252 |
| w/ UL | 0 | **0.191** | **0.120** | **0.181** | **0.140** | **0.095** | **0.208** | **0.168** | **0.098** | **0.117** | **0.146** |
| | 10 | **0.212** | **0.141** | **0.186** | **0.143** | **0.110** | **0.210** | **0.170** | **0.119** | **0.145** | **0.160** |
| | 20 | **0.239** | **0.152** | **0.189** | **0.148** | **0.119** | **0.217** | **0.176** | **0.136** | **0.169** | **0.172** |

**Table 2.** PSNR results produced by the proposed multi-stage network with (✓) or without (✗) Uncertainty Learning (UL).

| Scale | UL | Set5 | | | Set14 | | | BSD100 | | | Urban100 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Noise | | 0 | 10 | 20 | 0 | 10 | 20 | 0 | 10 | 20 | 0 | 10 | 20 |
| ×2 | ✗ | 33.78 | 30.76 | 29.27 | 30.55 | 28.21 | 27.10 | 29.87 | 27.56 | 26.54 | 27.75 | 25.82 | 24.93 |
| | ✓ | **34.36** | **31.03** | **29.54** | **30.93** | **28.42** | **27.27** | **30.43** | **27.69** | **26.62** | **28.16** | **26.10** | **25.08** |
| ×3 | ✗ | 31.77 | 29.72 | 28.08 | 28.39 | 27.24 | 26.20 | 27.72 | 26.67 | 25.77 | 25.88 | 24.95 | 24.22 |
| | ✓ | **32.27** | **29.91** | **28.34** | **28.92** | **27.53** | **26.45** | **28.17** | **26.83** | **25.88** | **26.26** | **25.19** | **24.31** |
| ×4 | ✗ | 30.39 | 28.80 | 27.34 | 27.49 | 26.52 | 25.53 | 26.82 | 25.97 | 25.15 | 24.86 | 24.15 | 23.49 |
| | ✓ | **30.79** | **29.07** | **27.56** | **27.81** | **26.76** | **25.74** | **27.02** | **26.12** | **25.27** | **25.07** | **24.36** | **23.62** |

### 4.2 Comparing UL with Deterministic Network

In the proposed kernel estimation network, uncertainty learning is used to improve the robustness to the degraded images. To demonstrate the effectiveness of uncertainty learning, we modify the network into a deterministic model by removing the Conv layer of variance $\boldsymbol{\sigma}$ and the Gaussian sampling operation. The KL loss is also excluded during training. The test images are generated by applying the blur kernels to the datasets followed by subsampling of scale factors 2, 3 and 4, then added with additive Gaussian noise. The visualization of the 9 anisotropic blur kernels used for testing is shown in Fig. 4.

We first compare the mean absolute error (MAE) of the blur kernels estimated by our uncertain network and the modified deterministic network. As shown in Table 1, by introducing uncertainty learning, we can estimate more accurate kernels with lower error compared with deterministic network. Note that when the LR image is disturbed by large noise, the deterministic network suffer an obvious performance drop. And our uncertain network can handle the noise inherent in the observation data well, thus produces more stable results robustly. We further verify the blind SR performance of the multi-stage network with two kinds of kernel estimations. Table 2 and Fig. 1 prove that our multi-stage network can produce higher PSNR results and shaper edges due to more accurate estimation of blur kernels.

### 4.3 Comparison with State-of-the-Art Methods

We have compared our method with several recent state-of-the-art blind SR methods, including ZSSR [42], IKC [15], DASR [46], KOALAnet [27] and MANet [32]. For a fair comparison, the results are generated by the official codes released by authors or directly cited from the original papers; all models are trained under the same training settings. Since IKC model was trained under a noise-free isotropic setting, we also retrained it under an anisotropic blur kernel setting with noise.

***Quantitative and visual comparison.*** Following [46], anisotropic blur kernels (as illustrated in Fig. 4) and different noise levels are used to evaluate the performance. The average PSNR results of the test methods for blind SR are reported in Table 3. Since zero-shot method ZSSR (blur kernel estimated by KernelGAN [4]) only leverages the internal information of test images, it has a relatively limited performance. We noticed that the image noise is preserved and magnified significantly after SR by ZSSR, so it is only tested under a noise-free setting. It can be seen that MANet slightly outperforms KOALAnet. And we have achieved some superior advantages over the other methods, especially at higher noise levels and scale factors. We compare the blind SR visualization results produced by different methods in Fig. 5, the specific blur kernel used for generating the LR image is displayed on the upper left. The LR images with a scale factor of 4 are added with Gaussian noise of level 10. It is obvious that the proposed method can reconstruct more high-frequency details and sharper edges than other methods.

***Computational complexity comparison.*** We have further compared the proposed network with other methods in terms of computational complexity. The total number of parameters of each deep network are listed in Table 4. We can see that the MANet contains the largest number of parameters over three times of the proposed network, as its RRDB-SFT architecture is very deep. Since we enforce the DCNNs in each stage to share the same parameters, the total number of parameters of the proposed multi-stage network is much smaller. Though there are $T = 4$ stages in the proposed network, the running time is similar to that of MANet. This is because the feature maps in the U-Net structure are gradually downsampled, thus the computational complexity can be much reduced.

### 4.4 Results on Real-World Images

We also conduct experiments on real-world images to demonstrate the generalization property and effectiveness of our method. We have only compared the visualization results of different methods, as there is no ground-truth. All models are trained on the anisotropic setting with noise, since real-world degradation is complicated. As shown in Fig. 6, our method can produce more natural and visually more pleasant results than other methods.

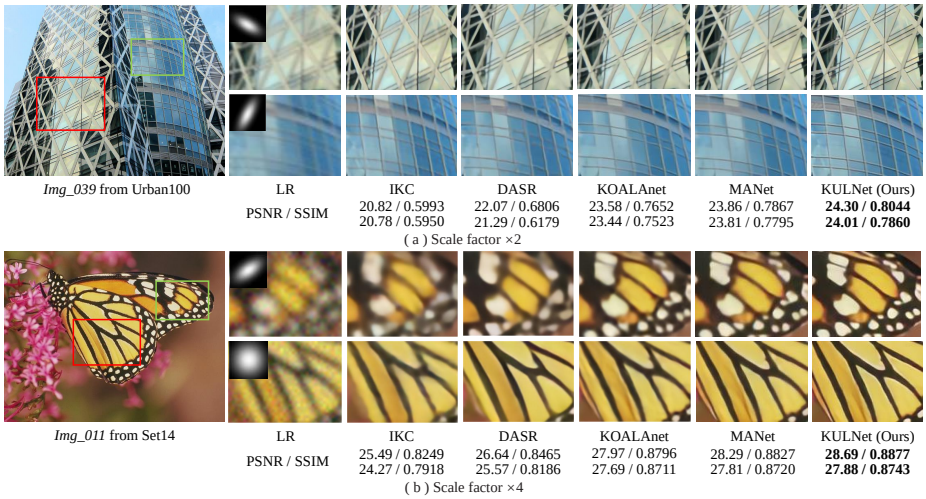


**Fig. 5.** Visual comparison to other methods. The blur kernels are illustrated on the top left. Noise levels are set to 0 and 10 for scale factor ×2 and ×4, respectively.

**Table 3.** Quantitative comparison of the SOTA blind SR methods and the proposed method on various datasets and noise levels.

| Method | Scale | Set5 | | | Set14 | | | BSD100 | | | Urban100 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Noise | | 0 | 10 | 20 | 0 | 10 | 20 | 0 | 10 | 20 | 0 | 10 | 20 |
| KernelGAN [4]+ZSSR [42] | ×2 | 26.94 | - | - | 23.96 | - | - | 23.17 | - | - | 21.69 | - | - |
| IKC [15] | | 27.89 | 27.62 | 26.86 | 26.29 | 25.90 | 25.26 | 26.03 | 25.67 | 25.12 | 23.84 | 23.35 | 22.82 |
| DASR [46] | | 29.89 | 28.13 | 27.18 | 27.25 | 26.06 | 25.41 | 26.97 | 25.78 | 25.21 | 24.58 | 23.54 | 22.98 |
| KOALAnet [27] | | 33.96 | 30.59 | 29.05 | 30.53 | 27.98 | 26.87 | 29.77 | 27.23 | 26.28 | 27.56 | 25.59 | 24.52 |
| MANet [32] | | 33.99 | 30.77 | 29.28 | 30.61 | 28.22 | 27.11 | 29.85 | 27.48 | 26.49 | 27.64 | 25.71 | 24.76 |
| KULNet (Ours) | | **34.36** | **31.03** | **29.54** | **30.93** | **28.42** | **27.27** | **30.43** | **27.69** | **26.62** | **28.16** | **26.10** | **25.08** |
| IKC [15] | ×3 | 28.40 | 27.01 | 26.00 | 26.42 | 25.32 | 24.54 | 26.20 | 25.20 | 24.56 | 23.59 | 23.24 | 22.35 |
| DASR [46] | | 29.40 | 27.54 | 26.43 | 26.92 | 25.68 | 24.89 | 26.65 | 25.42 | 24.76 | 24.23 | 23.45 | 22.57 |
| MANet [32] | | 31.78 | 29.65 | 28.10 | 28.50 | 27.22 | 26.18 | 27.79 | 26.64 | 25.74 | 25.42 | 24.62 | 23.85 |
| KULNet (Ours) | | **32.27** | **29.91** | **28.34** | **28.92** | **27.53** | **26.45** | **28.17** | **26.83** | **25.88** | **26.26** | **25.19** | **24.31** |
| KernelGAN [4]+ZSSR [42] | ×4 | 23.85 | - | - | 22.55 | - | - | 21.37 | - | - | 19.12 | - | - |
| IKC [15] | | 27.91 | 26.51 | 25.52 | 26.06 | 24.92 | 24.19 | 25.80 | 24.83 | 24.21 | 23.26 | 22.47 | 21.90 |
| DASR [46] | | 30.33 | 27.29 | 25.94 | 27.31 | 25.48 | 24.54 | 26.77 | 25.16 | 24.42 | 24.34 | 22.98 | 22.28 |
| KOALAnet [27] | | 30.36 | 28.56 | 27.13 | 27.35 | 26.19 | 25.33 | 26.72 | 25.73 | 24.95 | 24.37 | 23.76 | 23.01 |
| MANet [32] | | 30.38 | 28.73 | 27.31 | 27.41 | 26.46 | 25.50 | 26.78 | 25.96 | 25.16 | 24.49 | 23.91 | 23.23 |
| KULNet (Ours) | | **30.79** | **29.07** | **27.56** | **27.81** | **26.76** | **25.74** | **27.02** | **26.12** | **25.27** | **25.07** | **24.36** | **23.62** |

**Table 4.** Complexity comparison with other methods. The average running time is measured on the Set14 dataset for ×4.

| Method | IKC | DASR | KOALAnet | MANet | Ours |
|---|---|---|---|---|---|
| #Params. | 5.2M | 5.8M | 6.2M | 14.3M | 3.9M |
| Run Time(ms/image) | 568 | 96 | 991 | 157 | 176 |
| PSNR(dB) | 24.92 | 25.48 | 26.29 | 26.46 | 26.76 |


**Fig. 6.** Visualization results of different methods on real-world images upscaled by $\times 4$.

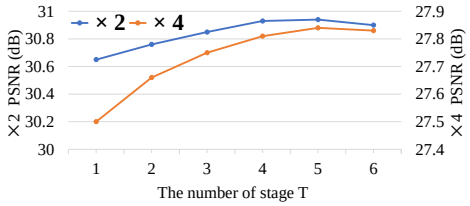


**Fig. 7.** Ablation study on the effect of the number of stage $T$.

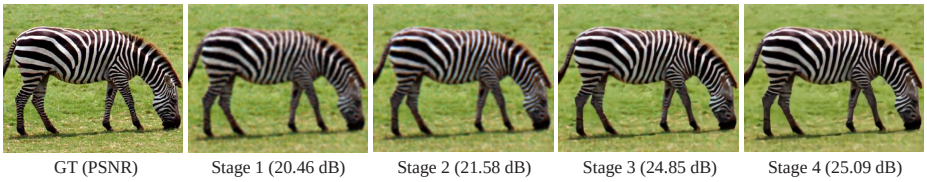


**Fig. 8.** Intermediate visual results of different stages for $\times 4$ blind SR.

### 4.5 Ablation Study

We conduct several ablation studies to verify the impacts of different modules in the proposed network, including the number of stages, the value of hyperparameter $\lambda$ and the effect of dense connections.

Fig. 7 shows the $\times 2$ and $\times 4$ PSNR results on Set14 produced by the proposed method with different number of stages, we can draw a conclusion that increasing the stage number $T$ leads to better results. We set $T = 4$ in our implementation, targeting a good trade-off between SR performance and computational complexity. Moreover, we have shown the intermediate image comparison results of different stages in Fig. 8, from which we can see that more high-frequency information has been recovered along with the increasing number of stages during the process of SR reconstruction.

**Table 5.** Results trained with different value of hyperparameter $\lambda$.

| $\lambda$ | 0 | 0.0001 | 0.001 | 0.01 | 0.1 | 1 |
|---|---|---|---|---|---|---|
| Set14 | 26.49 | 26.67 | 26.76 | 26.71 | 26.15 | 25.89 |
| BSD100 | 25.95 | 26.01 | 26.12 | 26.05 | 25.62 | 25.43 |

We have studied the influence of KL divergence regularization term by adjusting the value of hyperparameter $\lambda$. The PSNR results on Set14 and BSD100 are shown in Table 5. As the previous analysis has shown, if there are no constraints on mean and variance ($\lambda = 0$), the network tends to predict small $\boldsymbol{\sigma}$ for all samples, thus there exists almost no uncertainty in the network, the results are also similar to the deterministic networks in section 4.2. As $\lambda$ increasing, uncertainty learning can effectively improve the performance. When the KL constraint is too strong ($\lambda = 1$), the network will predict large variance for all samples, making the mean $\boldsymbol{\mu}$ deviate from the original feature maps. Here we set $\lambda$ as 0.001.

Finally, we have conducted an ablation study on the proposed network with or without dense connections. The PSNR results increase 0.13 dB on the Set14 dataset for a scale factor of 4, justifying the effectiveness of dense connections to KULNet.

## 5 Conclusions

In this paper, we formulate the blind SR problem as a joint maximum a posteriori probability (MAP) problem for estimating the unknown kernel and high-resolution image simultaneously. To improve the robustness of the kernel estimation network, we introduce uncertainty learning in the latent space instead of using deterministic feature maps. Then we propose a novel multi-stage SR network by unfolding the MAP estimator with the learned LSM prior and the estimated kernel. Both the scale prior coefficient and the local means of the LSM model are estimated through deep convolutional neural networks. All parameters of the MAP estimation algorithm and the DCNN parameters are jointly optimized through end-to-end training. Extensive experimental results on both synthetic and real datasets demonstrate that the proposed method outperforms existing state-of-the-art methods. Future research directions include the extension of this work to spatially varying blur kernels and the generalization study to more real-world test images.

**Acknowledgement.** This work was supported in part by the National Key R&D Program of China under Grant 2018AAA0101400 and the Natural Science Foundation of China under Grant 61991451, Grant 61632019, Grant 61621005, and Grant 61836008.

## References


1. Agustsson, E., Timofte, R.: Ntire 2017 challenge on single image super-resolution: Dataset and study. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 126–135 (2017) 9
2. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence **39**(12), 2481–2495 (2017) 4
3. Begin, I., Ferrie, F.: Blind super-resolution using a learning-based approach. In: Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. vol. 2, pp. 85–89. IEEE (2004) 2
4. Bell-Kligler, S., Shocher, A., Irani, M.: Blind super-resolution kernel estimation using an internal-gan. In: NeurIPS. pp. 284–293 (2019) 2, 11, 12
5. Bevilacqua, M., Roumy, A., Guillemot, C., Alberi-Morel, M.L.: Low-complexity single-image super-resolution based on nonnegative neighbor embedding (2012) 9
6. Chang, J., Lan, Z., Cheng, C., Wei, Y.: Data uncertainty learning in face recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5710–5719 (2020) 4, 6
7. Cornillere, V., Djelouah, A., Yifan, W., Sorkine-Hornung, O., Schroers, C.: Blind image super-resolution with spatially variant degradations. ACM Transactions on Graphics (TOG) **38**(6), 1–13 (2019) 2
8. Daubechies, I., Defrise, M., De Mol, C.: An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences **57**(11), 1413–1457 (2004) 7
9. Der Kiureghian, A., Ditlevsen, O.: Aleatory or epistemic? does it matter? Structural safety **31**(2), 105–112 (2009) 4
10. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: European conference on computer vision. pp. 184–199. Springer (2014) 1
11. Dong, C., Loy, C.C., Tang, X.: Accelerating the super-resolution convolutional neural network. In: European conference on computer vision. pp. 391–407. Springer (2016) 1
12. Dong, W., Huang, T., Shi, G., Ma, Y., Li, X.: Robust tensor approximation with laplacian scale mixture modeling for multiframe image and video denoising. IEEE Journal of Selected Topics in Signal Processing **12**(6), 1435–1448 (2018) 4
13. Dong, W., Wang, P., Yin, W., Shi, G., Wu, F., Lu, X.: Denoising prior driven deep neural network for image restoration. IEEE transactions on pattern analysis and machine intelligence **41**(10), 2305–2318 (2018) 7, 8
14. Garrigues, P., Olshausen, B.: Group sparse coding with a laplacian scale mixture prior. Advances in neural information processing systems **23** (2010) 4
15. Gu, J., Lu, H., Zuo, W., Dong, C.: Blind super-resolution with iterative kernel correction. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 1604–1613 (2019) 2, 3, 9, 11, 12
16. Gu, Y., Jin, Z., Chiu, S.C.: Active learning combining uncertainty and diversity for multi-class image classification. IET Computer Vision **9**(3), 400–407 (2015) 4
17. Haris, M., Shakhnarovich, G., Ukita, N.: Deep back-projection networks for super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1664–1673 (2018) 1

18. He, H., Siu, W.C.: Single image super-resolution using gaussian process regression. In: CVPR 2011. pp. 449–456. IEEE (2011) 2
19. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE international conference on computer vision. pp. 1026–1034 (2015) 9
20. He, Y., Yap, K.H., Chen, L., Chau, L.P.: A soft map framework for blind super-resolution image reconstruction. Image and Vision Computing **27**(4), 364–373 (2009) 2, 3
21. Huang, J.B., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5197–5206 (2015) 3, 9
22. Huang, T., Dong, W., Xie, X., Shi, G., Bai, X.: Mixed noise removal via laplacian scale mixture modeling and nonlocal low-rank approximation. IEEE Transactions on Image Processing **26**(7), 3171–3186 (2017) 4
23. Huang, T., Dong, W., Yuan, X., Wu, J., Shi, G.: Deep gaussian scale mixture prior for spectral compressive imaging. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16216–16225 (2021) 4
24. Jo, Y., Oh, S.W., Vajda, P., Kim, S.J.: Tackling the ill-posedness of super-resolution through adaptive target generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 16236–16245 (2021) 3
25. Kendall, A., Gal, Y.: What uncertainties do we need in bayesian deep learning for computer vision? NeurIPS (2017) 4
26. Kim, J., Lee, J.K., Lee, K.M.: Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1646–1654 (2016) 1
27. Kim, S.Y., Sim, H., Kim, M.: Koalanet: Blind super-resolution using kernel-oriented adaptive local adjustment. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10611–10620 (2021) 3, 11, 12
28. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) 9
29. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. ICLR (2014) 5
30. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4681–4690 (2017) 1
31. Lee, C., Chung, K.S.: Gram: Gradient rescaling attention model for data uncertainty estimation in single image super resolution. In: 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA). pp. 8–13. IEEE (2019) 4
32. Liang, J., Sun, G., Zhang, K., Van Gool, L., Timofte, R.: Mutual affine network for spatially variant kernel estimation in blind image super-resolution. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4096–4105 (2021) 3, 4, 11, 12
33. Liang, J., Zhang, K., Gu, S., Van Gool, L., Timofte, R.: Flow-based kernel prior with application to blind super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10601–10610 (2021) 3
34. Lim, B., Son, S., Kim, H., Nah, S., Mu Lee, K.: Enhanced deep residual networks for single image super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 136–144 (2017) 1
35. Luo, Z., Huang, Y., Li, S., Wang, L., Tan, T.: Unfolding the alternating optimization for blind super resolution. arXiv preprint arXiv:2010.02631 (2020) 3

36. Martin, D., Fowlkes, C., Tal, D., Malik, J.: 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 2001. vol. 2, pp. 416–423. IEEE (2001) 9
37. Michaeli, T., Irani, M.: Nonparametric blind super-resolution. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 945–952 (2013) 2, 3
38. Ning, Q., Dong, W., Shi, G., Li, L., Li, X.: Accurate and lightweight image super-resolution with model-guided deep unfolding network. IEEE Journal of Selected Topics in Signal Processing (2020) 1, 7
39. Ning, Q., Dong, W., Shi, G., Li, L., Li, X.: Uncertainty-driven loss for single image super-resolution. NeurIPS (2021) 4
40. Ning, Q., Dong, W., Wu, F., Wu, J., Lin, J., Shi, G.: Spatial-temporal gaussian scale mixture modeling for foreground estimation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 11791–11798 (2020) 4, 7
41. Shi, G., Huang, T., Dong, W., Wu, J., Xie, X.: Robust foreground estimation via structured gaussian scale mixture modeling. IEEE Transactions on Image Processing **27**(10), 4810–4824 (2018) 4
42. Shocher, A., Cohen, N., Irani, M.: "zero-shot" super-resolution using deep internal learning. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3118–3126 (2018) 2, 11, 12
43. Soh, J.W., Cho, S., Cho, N.I.: Meta-transfer learning for zero-shot super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3516–3525 (2020) 2
44. Tao, G., Ji, X., Wang, W., Chen, S., Lin, C., Cao, Y., Lu, T., Luo, D., Tai, Y.: Spectrum-to-kernel translation for accurate blind image super-resolution. Advances in Neural Information Processing Systems **34**, 22643–22654 (2021) 3
45. Timofte, R., Agustsson, E., Van Gool, L., Yang, M.H., Zhang, L.: Ntire 2017 challenge on single image super-resolution: Methods and results. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 114–125 (2017) 9
46. Wang, L., Wang, Y., Dong, X., Xu, Q., Yang, J., An, W., Guo, Y.: Unsupervised degradation representation learning for blind super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10581–10590 (2021) 2, 3, 9, 11, 12
47. Wang, Q., Tang, X., Shum, H.: Patch based blind image super resolution. In: Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1. vol. 1, pp. 709–716. IEEE (2005) 2
48. Wang, X., Yu, K., Dong, C., Loy, C.C.: Recovering realistic texture in image super-resolution by deep spatial feature transform. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 606–615 (2018) 2
49. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: International conference on curves and surfaces. pp. 711–730. Springer (2010) 9
50. Zhang, K., Gool, L.V., Timofte, R.: Deep unfolding network for image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3217–3226 (2020) 2
51. Zhang, K., Zuo, W., Zhang, L.: Learning a single convolutional super-resolution network for multiple degradations. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3262–3271 (2018) 2

52. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: Proceedings of the European conference on computer vision (ECCV). pp. 286–301 (2018) 1
53. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2472–2481 (2018) 1