

# KOALAnet: Blind Super-Resolution using Kernel-Oriented Adaptive Local Adjustment

Soo Ye Kim\*

Hyeonjun Sim\*

Munchurl Kim<sup>†</sup>

Korea Advanced Institute of Science and Technology

{sooyekim, flhy5836, mkimee}@kaist.ac.kr

## Abstract

Blind super-resolution (SR) methods aim to generate a high quality high resolution image from a low resolution image containing unknown degradations. However, natural images contain various types and amounts of blur: some may be due to the inherent degradation characteristics of the camera, but some may even be intentional, for aesthetic purposes (e.g. Bokeh effect). In the case of the latter, it becomes highly difficult for SR methods to disentangle the blur to remove, and that to leave as is. In this paper, we propose a novel blind SR framework based on kernel-oriented adaptive local adjustment (KOALA) of SR features, called KOALAnet, which jointly learns spatially-variant degradation and restoration kernels in order to adapt to the spatially-variant blur characteristics in real images. Our KOALAnet outperforms recent blind SR methods for synthesized LR images obtained with randomized degradations, and we further show that the proposed KOALAnet produces the most natural results for artistic photographs with intentional blur, which are not over-sharpened, by effectively handling images mixed with in-focus and out-of-focus areas.

## 1. Introduction

When a deep neural network is trained under a specific scenario, its generalization ability tends to be limited to that particular setting, and its performance deteriorates under a different condition. This is a major problem in single image super-resolution (SR), where most neural-network-based methods have focused on the upscaling of low resolution (LR) images to high resolution (HR) images solely under the bicubic downsampling setting [13, 15, 16, 26], until very recently. Naturally, their performance tends to severely drop if the input LR image is degraded by even a slightly different downsampling kernel, which is often the case in real images [23]. Hence, more recent SR methods aim for *blind* SR,

where the true degradation kernels are unknown [5, 8].

However, this unknown blur may be of various types with different characteristics. Often, images are captured with a different depth-of-field (DoF) by manipulating the aperture sizes and the focal lengths of camera lenses, for aesthetic purposes (e.g. Bokeh effect) as shown in Fig. 1. Recent mobile devices even try to simulate this synthetically (e.g. portrait mode) for artistic effects [24]. Although a camera-specific degradation could be spatially-equivariant (similar to the way LR images are generated for SR), the blur generated due to DoF of the camera would be *spatially-variant*, where some areas are in focus, and others are out of focus. These types of LR images are extremely challenging for SR, since ideally, the intentional blur must be left unaltered (should not be over-sharpened) to maintain the photographer’s intent after SR. However, the SR results of such images are yet to be analyzed in literature.

In this paper, we propose a blind SR framework based on kernel-oriented adaptive local adjustment (KOALA) of SR features, called KOALAnet, by jointly learning the degradation and restoration kernels. The KOALAnet consists of two networks: a downsampling network that estimates *spatially-variant* blur kernels, and an upsampling network that fuses this information by mapping the predicted degradation kernels to the feature kernel space, predicting degradation-specific local feature adjustment parameters that are applied by *spatially-variant local filtering* on the SR feature maps. After training under a random anisotropic Gaussian degradation setting, our KOALAnet is able to accurately predict the underlying degradation kernels and effectively leverage this information for SR. Moreover, it demonstrates a good generalization ability on historic images containing unknown degradations compared to previous blind SR methods. We further provide comparisons on real aesthetic DoF images, and show that our KOALAnet effectively handles images with intentional blur. Our contributions are three-fold:

- We propose a blind SR framework that jointly learns *spatially-variant* degradation and restoration kernels. The restoration (upsampling) network leverages novel

\*Both authors contributed equally to this work.

<sup>†</sup>Corresponding author.

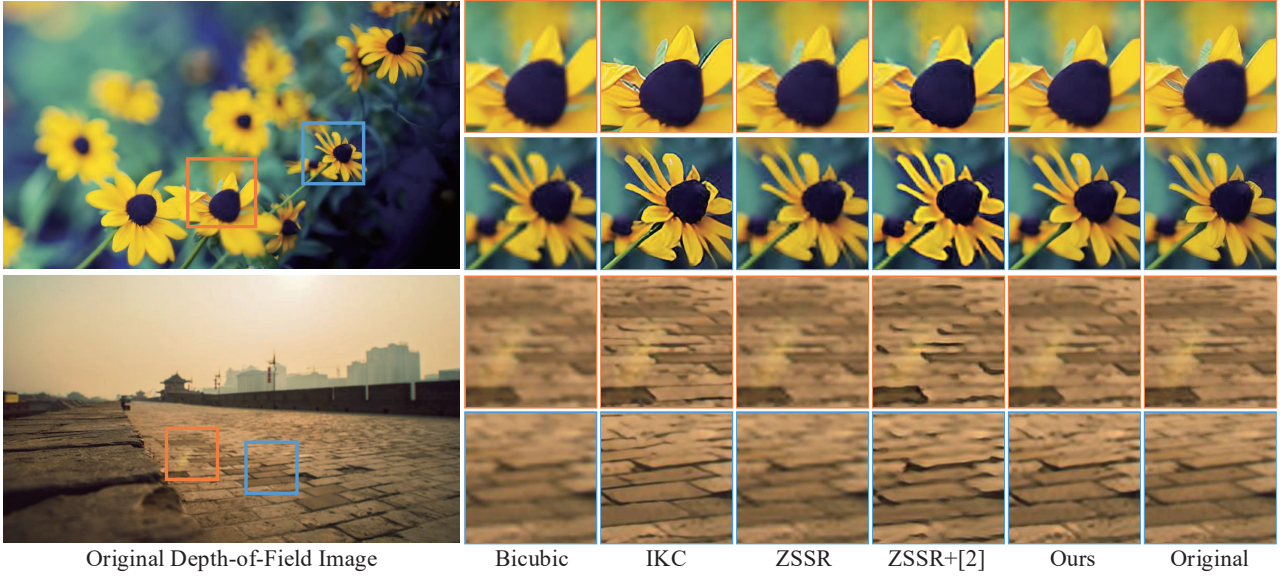


Figure 1: Qualitative comparison on artistic photographs with intentional blur for  $\times 4$ . Some methods (IKC [8], ZSSR [23]+KernelGAN [2]) over-sharpen even the background (out-of-focus) regions that should be left blurry, while others generate blurry foreground (in-focus) regions. Our KOALAnet handles both regions well, generating results with similar blurriness characteristics as the original image.

KOALA modules to adaptively adjust the SR features based on the predicted degradation kernels. The KOALA modules are *extensible*, and can be inserted into any CNN architecture for image restoration tasks.

- We empirically show that the proposed KOALAnet outperforms the recent state-of-the-art blind SR methods for synthesized LR images obtained under randomized degradation conditions, as well as for historic LR images with unknown degradations.
- We first analyze SR results on images mixed with *in-focus* and *out-of-focus* regions, showing that our KOALAnet is able to discern intentionally blurry areas and process them accordingly, leaving the photographer’s intent unchanged after SR.

## 2. Related Work

**Single degradation SR.** Since the first CNN-based SR method by Dong *et al.* [6], highly sophisticated deep learning networks have been proposed in image SR [13, 15, 16, 22, 25, 26, 30], achieving remarkable quantitative or qualitative performance. Especially, Wang *et al.* [25] introduced feature-level affine transformation based on segmentation prior to generate class-specific texture in the SR result. Although these methods perform promisingly under the ideal bicubic-degraded setting, they tend to produce *over-sharpened* or *blurry* results if the degradations present in the test images deviate from bicubic degradation.

**Multiple degradation SR.** Recent methods handling multiple types of degradations can be categorized into *non-blind*

SR [27, 29], where the LR images are coupled with the ground truth degradation information (blur kernel or noise level), or *blind SR* [2, 5, 8], where only the LR images are given without the ground truth degradation information that is then to be estimated. Among the former, Zhang *et al.* [29] provided the principal components of the Gaussian blur kernel and the level of additive Gaussian noise by concatenating them with the LR input for degradation-aware SR. Xu *et al.* [27] also integrated the degradation information in the same way, but with a backbone network using dynamic up-sampling filters [12], raising the SR performance. However, these methods require ground truth blur information at test time, which is unrealistic for practical application scenarios.

Among blind SR methods that predict the degradation information, an inspiring work by Gu *et al.* [8] inserted spatial feature transform modules [25] in the CNN architecture to integrate the degradation information with iterative kernel correction. However, the iterative framework can be time-consuming since the entire framework must be repeated many times during inference, and the optimal number of iteration loops varies among input images, requiring human intervention for maximal performance. Furthermore, their network generates vector kernels that are eventually stretched with repeated values to be inserted to the SR network, limiting the degradation modeling capability of local characteristics. Another prominent work is the KernelGAN [2] that generates downsampled LR images by learning the internal patch distribution of the test LR image. The downsampled LR patches, or the kernel information, and the original test LR images are then plugged into zero-shot SR [23] or

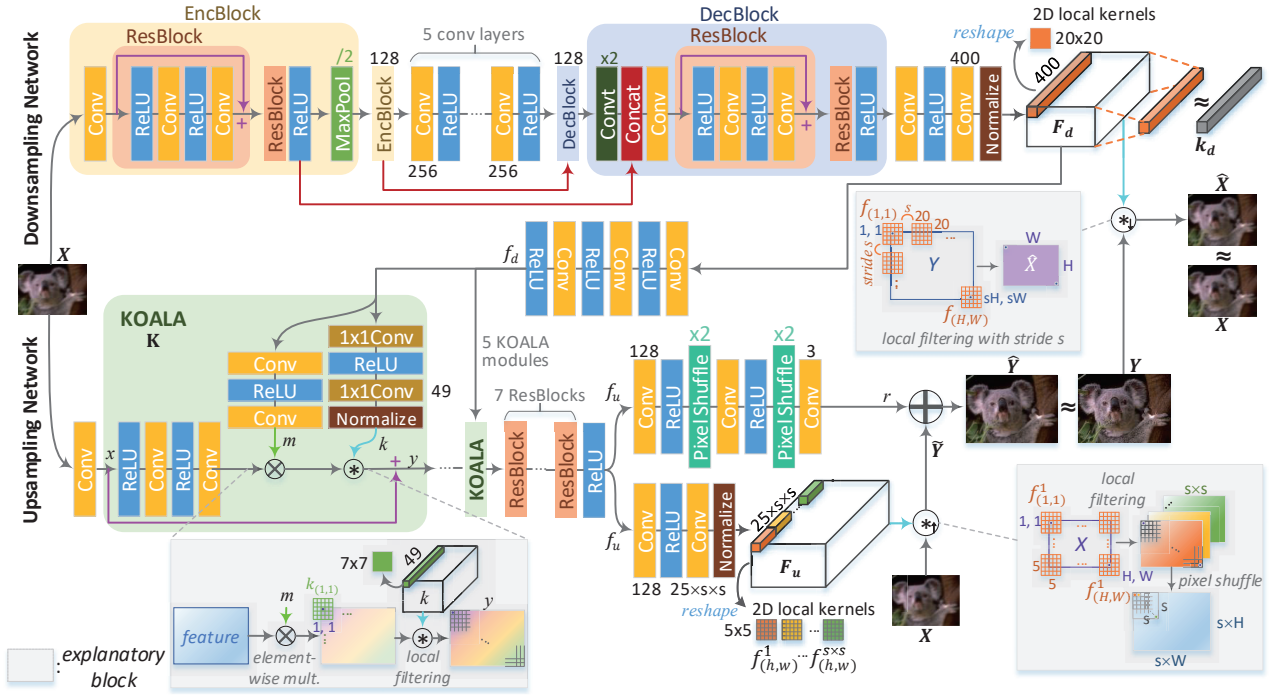


Figure 2: Our proposed blind SR framework, KOALAnet, for  $\times 4$ . The downsampling network predicts spatially-variant kernels, which are fed into the KOALA modules that in turn produce degradation-specific multiplicative ( $m$ ) and local filter parameters ( $k$ ) used to modulate the features in the upsampling network. The upsampling network generates an SR result with spatially-variant upsampling kernels.

non-blind SR [29] methods. There also exist methods that employ GANs to generate realistic kernels for data augmentation [31], or learn to synthesize LR images along with the SR image [4, 9]. In comparison, our downsampling network predicts the underlying *spatially-variant* blur kernels that are used to modulate and locally filter the upsampling features.

**Dynamic filter generation.** Jia *et al.* [11] first proposed *dynamic filter networks* that generate image- and location-specific *dynamic filters* that filter the input images in a locally adaptive manner to better handle the non-stationary property of natural images, in contrast to the conventional convolution layers with spatially-equivariant filters. Application-wise, Niklaus *et al.* [19, 20] and Jo *et al.* [12] successfully employed the dynamic filtering networks for video frame interpolation, and video SR, respectively. The recent non-blind SR method by Xu *et al.* [27] also employed a two-branch dynamic upsampling architecture [12]. However, the provided ground truth degradation kernel is still restricted to spatially *uniform* kernels and are entered naively by simple concatenation, unlike our proposed KOALAnet that estimates the underlying *non-uniform* blur kernels from input LR images and effectively integrates this information for SR.

### 3. Proposed Method

We propose a blind SR framework with (i) a downsampling network that predicts spatially-variant degradation ker-

nels, and (ii) an upsampling network that contains KOALA modules, which adaptively fuses the degradation kernel information for enhanced blind SR.

#### 3.1. Downsampling Network

**Data generation.** During training, an LR image,  $X$ , is generated by applying a random anisotropic Gaussian blur kernel,  $k_g$ , on an HR image,  $Y$ , and downsampling it with the bicubic kernel,  $k_b$ , similar to [8, 27, 29], given as,

$$X = (Y * k_g) * k_b \downarrow_s = (Y * k_d) \downarrow_s, \quad (1)$$

where  $\downarrow_s$  denotes downsampling by scale factor  $s$ . Hence, the downsampling kernel  $k_d$  can be obtained as  $k_d = k_g * k_b$ , and the degradation process can be implemented by an  $s$ -stride convolution of  $Y$  by  $k_d$ . We believe that anisotropic Gaussian kernels are a more suitable choice than isotropic Gaussian kernels for blind SR, as anisotropic kernels are the more generalized superset. We do not apply any additional anti-aliasing measures (like in the default Matlab `imresize` function), since  $Y$  is already low-pass filtered by  $k_g$ .

The downsampling network, shown in the upper part of Fig. 2, takes a degraded LR RGB image,  $X$ , as input, and aims to predict its underlying degradation kernel that is assumed to have been used to obtain  $X$  from its HR counterpart,  $Y$ , through a U-Net-based [21] architecture with ResBlocks. The output,  $F_d$ , is a 3D tensor of size  $H \times W \times 400$ ,



composed of  $20 \times 20$  local filters at every  $(h, w)$  pixel location. The local filters are normalized to have a sum of 1 (denoted as *Normalize* in Fig. 2) by subtracting each of their mean values and adding a bias of  $1/400$ . With  $F_d$ , the LR image,  $\hat{X}$ , can be reconstructed by,

$$\hat{X} = (Y \circledast F_d) \downarrow_s, \quad (2)$$

where  $\circledast \downarrow_s$  represents  $20 \times 20$  local filtering [11] at each pixel location with stride  $s$ , as illustrated in Fig. 2.

For training, we propose to use an LR reconstruction loss,  $L_r = l_1(\hat{X}, X)$ , which indirectly enforces the downsampling network to predict a *spatially-variant degradation kernel* at each pixel location based on image prior. To bring flexibility in the spatially-variant kernel estimation, the loss with the ground truth kernel is only given to the spatial-wise mean of  $F_d$ . Then, the total loss for the downsampling network is given as,

$$L_d = l_1(\hat{X}, X) + l_1(E_{hw}[F_d], k_d), \quad (3)$$

where  $E_{hw}[\cdot]$  denotes a spatial-wise mean over  $(h, w)$ , and  $k_d$  is reshaped to  $1 \times 1 \times 400$  from the original size of  $20 \times 20$ .

Estimating the blur kernel for a smooth region in an LR image is difficult since dissimilar blur kernels may produce similar smooth pixel values. Consequently, if the network aims to directly predict the true blur kernel, the gradient of a kernel matching loss may not back-propagate a desirable signal. Meanwhile, for highly textured regions of HR images, the induced LR images are largely influenced by the blur kernels, which enables the downsampling network to find inherent degradation cues from the LR images. In this case, the degradation information can be highly helpful in reconstructing the SR image as well, since most of the SR reconstruction error tends to occur in these regions.

### 3.2. Upsampling Network

We consider the upsampling process to be the inverse of the downsampling process, and thus, design an upsampling network in correspondance with the downsampling network as shown in Fig. 2. The upsampling network takes in the degraded LR input,  $X$ , of size  $H \times W \times 3$ , and generates an SR output,  $\hat{Y}$ , of size  $sH \times sW \times 3$ , where  $s$  is a scale factor. In the early convolution layers of the upsampling network, the SR feature maps are adjusted by five cascaded KOALA modules,  $K$ , which are explained in detail in the next section. Then, after seven cascaded residual blocks,  $R$ , the resulting feature map,  $f_u$ , is given by,  $f_u = (RL \circ R^7 \circ K^5 \circ Conv)(X)$ , where  $RL$  is ReLU activation [7].  $f_u$  is fed separately into a residual branch and a filter generation branch similar to [12], where the residual map,  $r$ , and local upsampling filters,  $F_u$ , are obtained as,

$$r = (Conv \circ PS \circ RL \circ Conv \circ PS \circ RL \circ Conv)(f_u), \quad (4)$$

$$F_u = (Normalize \circ Conv \circ RL \circ Conv)(f_u), \quad (5)$$

for  $s = 4$ , where  $PS$  is a pixel shuffler [22] of  $s = 2$  and *Normalize* denotes normalizing by subtracting the mean and adding a bias of  $1/25$  for each  $5 \times 5$  local filter. The second  $PS$  and its preceding convolution layer are removed when generating  $r$  for  $s = 2$ .

When applying the generated  $F_u$  of size  $H \times W \times (25 \times s \times s)$  on the input  $X$ ,  $F_u$  is split into  $s \times s$  tensors in the channel direction, and each chunk of  $H \times W \times 25$  tensor is interpreted as a  $5 \times 5$  local filter at every  $(h, w)$  pixel location. They are applied on  $X$  (same filters for RGB channels) by computing the local inner product at the corresponding grid position  $(h, w)$ . After filtering all of the  $s \times s$  chunks, the produced  $H \times W \times (s \times s \times 3)$  tensor is pixel-shuffled with scale  $s$  to generate the enlarged  $\hat{Y}$  of size  $sH \times sW \times 3$  similar to [12]. Finally,  $\hat{Y}$  is computed as  $\hat{Y} = \hat{Y} + r$ , and the upsampling network is trained with  $l_1(\hat{Y}, Y)$ .

**Kernel-oriented adaptive local adjustment.** We propose a novel feature transformation module, KOALA, that adaptively adjusts the intermediate features in the upsampling network based on the degradation kernels predicted by the downsampling network. The KOALA modules are placed at the earlier stage of feature extraction in order to calibrate the anisotropically degraded LR features before the reconstruction phase.

Specifically, when the input feature,  $x$ , is entered into a KOALA module,  $K$ , it goes through 2 convolution layers, and is adjusted by a set of multiplicative parameters,  $m$ , followed by a set of local kernels,  $k$ , generated based on the predicted degradation kernels,  $F_d$ . Instead of directly feeding  $F_d$  into  $K$ , the kernel features,  $f_d$ , extracted after 3 convolution layers are entered. After a local residual connection, the output,  $y$ , of the KOALA module is given by,

$$y = K(x \mid f_d) = \{(\text{Conv} \circ \text{RL} \circ \text{Conv} \circ \text{RL})(x) \otimes m\} \circledast k + x, \quad (6)$$

where,

$$m = (\text{Conv} \circ \text{RL} \circ \text{Conv})(f_d), \quad \text{and} \quad (7)$$

$$k = (\text{Normalize} \circ \text{Conv}_{1 \times 1} \circ \text{RL} \circ \text{Conv}_{1 \times 1})(f_d). \quad (8)$$

In Eq. 6,  $\otimes$  and  $\circledast$  denote element-wise multiplication and local feature filtering, respectively. For generating  $k$ ,  $1 \times 1$  convolutions are employed so that spatially adjacent values of kernel features,  $f_d$ , are not mixed by convolution operations. The kernel values of  $k$  are constrained to have a sum of 1 (*Normalize*), like for  $F_d$  and  $F_u$ .

The local feature filtering operation,  $\circledast$ , is first applied by reshaping a  $1 \times 1 \times 49$  vector at each grid position  $(h, w)$  to a  $7 \times 7$  2D local kernel, and computing the local inner product at each  $(h, w)$  position of the input feature. Since the same  $7 \times 7$  kernels are applied channel-wise, the multiplicative parameter,  $m$ , introduces element-wise scaling for the features over the channel depth. This is also efficient in terms of the

Method $\times 2$	Set5 PSNR/SSIM	Set14 PSNR/SSIM	BSD100 PSNR/SSIM	Urban100 PSNR/SSIM	Manga109 PSNR/SSIM	DIV2K-val PSNR/SSIM	DIV2KRRK[2] PSNR/SSIM	Complexity Time (s)/GFLOPs
Bicubic	27.11/0.7850	26.00/0.7222	26.09/0.6838	22.82/0.6537	24.87/0.7911	28.27/0.7835	28.73/0.8040	- / -
ZSSR [23]	27.30/0.7952	26.55/0.7402	26.46/0.7020	23.13/0.6706	25.43/0.8041	28.69/0.7958	29.10/0.8215	24.69/6,238
KernelGAN [2] +ZSSR [23]	27.35/0.7839	24.57/0.7061	25.56/0.6990	23.12/0.6907	25.99/0.8270	27.66/0.7892	30.36/0.8669	230.66/10,219
BlindSR [5]	<u>28.61/0.8371</u>	<u>26.63/0.7686</u>	<u>26.86/0.7381</u>	<u>24.11/0.7396</u>	<u>26.19/0.8499</u>	<u>28.90/0.8227</u>	29.44/0.8464	<u>9.21/13,910</u>
<b>Ours</b>	<b>33.08/0.9137</b>	<b>30.35/0.8568</b>	<b>29.70/0.8248</b>	<b>27.19/0.8318</b>	<b>32.61/0.9369</b>	<b>32.55/0.8902</b>	<b>31.89/0.8852</b>	<b>0.71/201</b>

Method $\times 4$	Set5	Set14	BSD100	Urban100	Manga109	DIV2K-val	DIV2KRRK[2]	Complexity
Bicubic	26.41/0.7511	24.73/0.6641	25.12/0.6321	22.04/0.6061	23.60/0.7482	27.04/0.7417	25.33/0.6795	- / -
ZSSR [23]	26.49/0.7530	24.93/0.6812	25.36/0.6526	22.39/0.6327	24.43/0.7813	27.39/0.7590	25.61/0.6911	16.91/6,091
KernelGAN [2] +ZSSR [23]	22.12/0.5989	19.73/0.5194	21.02/0.5377	20.12/0.5743	22.61/0.7345	23.75/0.6830	26.81/0.7316	357.70/11,908
IKC [8] <i>last</i>	27.73/0.8024	25.38/0.7162	25.68/0.6844	23.03/0.6852	25.44/0.8273	27.61/0.7843	<u>27.39/0.7639</u>	<u>0.67/575</u>
IKC [8] <i>max</i>	28.04/0.8079	<u>25.85/0.7261</u>	<u>26.01/0.6951</u>	<u>23.21/0.6943</u>	<u>25.82/0.8361</u>	<u>27.98/0.7912</u>	<b>27.70/0.7684</b>	<u>0.67/575</u>
<b>Ours</b>	<b>30.28/0.8658</b>	<b>27.20/0.7541</b>	<b>26.97/0.7172</b>	<b>24.71/0.7427</b>	<b>28.48/0.8814</b>	<b>29.44/0.8156</b>	<b>27.77/0.7637</b>	<b>0.59/57</b>

Table 1: Quantitative comparison on various datasets. We also provide a comparison on computational complexity in terms of the average inference time on Set5, and GFLOPs on “baby” in Set5. **Bold** and underline indicate the best and the second best performance, respectively.

number of parameters, compared to predicting the per-pixel local kernels for every channel ( $49 + 64$  vs.  $49 \times 64$  filter parameters). By placing the residual connection after the feature transformations (Eq. 6), the adjustment parameters can be considered as removing the unwanted feature residuals related to degradation from the original input features.

### 3.3. Training Strategy

We employ a 3-stage training strategy: (i) the downsampling network is pre-trained with  $l_1(\hat{X}, X)$ ; (ii) the upsampling network is pre-trained with  $l_1(\hat{Y}, Y)$  by replacing all KOALA modules with ResBlocks; (iii) the whole framework (KOALANet) including the KOALA modules (with convolution layers needed for generating  $f_d$ ,  $m$  and  $k$  inserted on the pre-trained ResBlocks) is jointly optimized based on  $l_1(\hat{X}, X) + l_1(\hat{Y}, Y)$ . With this strategy, the KOALA modules can be effectively trained with already meaningful features obtained from the early training phases, and focus on utilizing the degradation kernel cues for SR.

## 4. Experiment Results

**Data generation.** In our implementations,  $k_d$  of size  $20 \times 20$  is computed by convolving  $k_b$  with a random anisotropic Gaussian kernel ( $k_g$ ) of size  $15 \times 15$ , following Eq. 1. It should be noted that  $k_b$  is originally a bicubic downscaling kernel of size  $4 \times 4$  same as in the *imresize* function of Matlab *without* anti-aliasing, but is zero-padded to be  $20 \times 20$  to align with the size of  $k_d$  as well as to avoid image shift. The Gaussian kernels for degradation are generated by randomly rotating a bivariate Gaussian kernel by  $\theta \sim \text{Uniform}(0, \pi/2)$ , and by randomly selecting its kernel width that is determined by a diagonal covariance matrix with  $\sigma_{11}$  and  $\sigma_{22} \sim \text{Uniform}(0.2, 4.0)$ . With  $k_d$ , we build our training data on the DIV2K [1] dataset according to Eq.

1. Testsets are generated using Set5 [3], Set14 [28], BSD100 [17], Urban100 [10], Manga109 [18] and DIV2K-val [1] for comparison with other methods. When generating the testsets, we ensure that different parameters are selected for different images by assigning different random seed values. We additionally compare to DIV2KRRK proposed in [2], which contains DIV2K images that are randomly degraded.

**Training parameters.** All convolution filters in the KOALANet are of size  $3 \times 3$  with 64 output channels following [13], unless otherwise noted as  $1 \times 1$ Conv or with the output channel noted next to an operation block in Fig. 2. All CNN-based networks used in our experiments are trained with LR patches of size  $64 \times 64$  normalized to  $[-1, 1]$ , where each patch is randomly cropped, and randomly degraded with  $k_d$  during training. The mini-batch size is 8, and the initial learning rate of  $10^{-4}$  is decreased by 1/10 at 80% and 90% of 200K iterations for each training stage. We consider  $s = 2$  and  $s = 4$  for SR in our experiments.

### 4.1. Comparison to Existing Blind SR Methods

We compare our method with recent state-of-the-art blind SR methods, BlindSR [5] and IKC [8]. For [5], we use the pre-trained model in an independent implementation by an author with only  $s = 2$  model. For [8], we use the official pre-trained model by the authors with only  $s = 4$  model. We also compare against ZSSR [23] with default degradation as well as by incorporating KernelGAN [2] to provide degradation information, both with the official codes.

**Quantitative comparison.** We compare the Y-channel PSNR and SSIM for various methods on the six random anisotropic degradation testsets as well as DIV2KRRK [2] in Table 1. We also provide a comparison on the average inference time and GFLOPs in the rightmost column, where the inference time in seconds is measured on Set5 with



Figure 3: Qualitative comparison to other methods. The estimated or ground truth degradation kernels are placed on the top left.

an NVIDIA Titan RTX excluding file I/O times, and the GFLOPs is computed on the “baby” image in Set5 that is of  $512 \times 512$  resolution in terms of the HR ground truth. The inference time includes zero-shot training for ZSSR [23] and optimization for [2, 5]. The inherent limitation of zero-shot models is that they cannot leverage the abundant training data that is utilized by other methods as image-specific CNNs are trained at test time. For IKC [8], we report the results of the last iteration (IKC *last*) as well as those producing the maximum PSNR (IKC *max*) from total 7 iteration loops. Since IKC is trained under an isotropic setting, its data modeling capability tends to be limited under a more randomized superset of anisotropic blur. Note that BlindSR [5] had been trained under an anisotropic setting like ours. On DIV2K [2], where artificial noise is injected to the synthetic degradation kernels, internal-learning-based methods ZSSR [23] and KernelGAN [2] are advantageous over the other methods as they can adapt to the unorthodox kernels. Nevertheless, our method outperforms all compared methods on DIV2K even though it was not trained with kernel noise, demonstrating good generalization ability. On other testsets, our KOALANet outperforms the other methods by a large margin of over 1 dB in most cases.

**Qualitative comparison.** We compare the visual results on the randomized anisotropic testset in Fig. 3. We have also visualized the mean of the predicted spatially-variant kernels along with the ground truth kernels. Our method is able to restore sharp edges and high frequency details. Most importantly, in Fig. 4, we also compare our method under real conditions on old historic images [14] without ground truth labels. In this case, the results are generated using the configuration for real images for ZSSR, and we show the results generated at the last iteration for IKC. Our method performs well even on these real images with unknown degradations, demonstrating good generalization ability.

## 4.2. Results on Aesthetic Images

We collected several shallow DoF images from the web containing intentional spatially-variant blur for aesthetic purposes, to compare the SR results of existing blind SR methods [2, 8, 23] to ours. Before applying SR, these images are bicubic-downsampled so that we can consider the original images as ground truth, in order to gauge the intended blur characteristics in the original images. As shown in Fig. 1, IKC [8] and ZSSR [23] with KernelGAN [2] tend to over-sharpen even the intentionally blurry areas that should





(a) Real images upscaled by 2.

(left: input, top: ZSSR, middle: BlindSR, bottom: Ours)



(b) Real images upscaled by 4.

(left: input, top: ZSSR, middle: IKC, bottom: Ours)

Figure 4: Qualitative results on old historic images [14].

be left blurry. ZSSR [23] produces blurry results overall, even in the foreground (in-focus) regions. In contrast, our KOALANet leaves the originally out-of-focus region blurry and appropriately upscales the overall image, yielding results that are closest to the original images. For further analysis, we also compare our KOALANet with a Baseline that only has the upsampling network in our framework. As shown in Fig. 5, the regions with strong blur far away from the in-focus area remain blurry for both methods. However, the Baseline cannot correctly disentangle the intentional blur from the degradation blur in the boundary areas between the in-focus and completely out-of-focus areas, where it can be ambiguous whether the blur should be sharpened or left blurry. With shallow DoF images where only a narrow band of regions are in focus in Fig. 5, the Baseline tends to produce results with a *deeper DoF* than the original image due to over-sharpening of the boundary areas.

### 4.3. Ablation Study

In this Section, we analyze the effect of the different components in our framework with various ablation studies, and provide visualizations of estimated blur kernels, local upsampling filters, and local filters in the KOALA modules.

**Effect of KOALA modules.** We analyzed the effect of the proposed KOALA modules by retraining the following

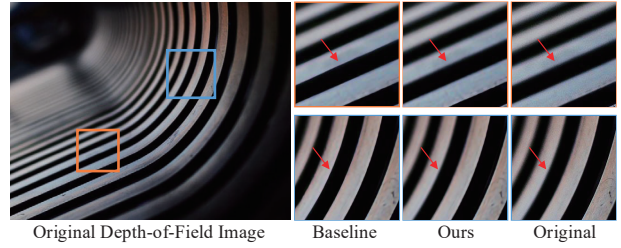


Figure 5: Comparison with the Baseline on a shallow DoF image for  $\times 4$ . The Baseline with only the upsampling network tends to generate a deeper DoF image compared to the original image.

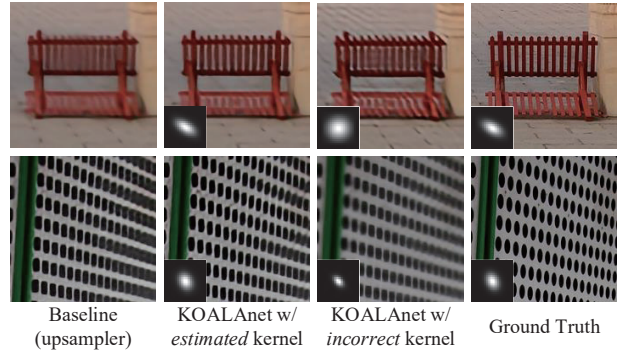


Figure 6: Effect of the degradation kernels on KOALANet for  $\times 4$ .

Model	Baseline (upsampler)	KOALA only $k$	KOALANet	KOALA +GT kernel
PSNR	29.20	29.40	29.44	<b>29.67</b>
SSIM	0.8110	0.8150	0.8156	<b>0.8212</b>

Table 2: Ablation study on the KOALA module for  $\times 4$ .

SR models: (i) Baseline with only the upsampling network without using any degradation kernel information (no down-sampling network, nor KOALA modules), (ii) a model that only has the parameters  $k$  (not  $m$ ) in the KOALA modules, (iii) a model to which ground truth kernels are given instead of the estimated degradation kernels (KOALA+GT kernel). From the Baseline, adding KOALA modules with only  $k$  parameters improves PSNR performance by 0.2 dB, and adding  $m$  further improves the PSNR gain by 0.04 dB, showing the effectiveness of our proposed KOALA modules in incorporating degradation kernel information. The SR performance of KOALA modules with ground truth kernels can be considered as an upper bound, with 0.23 dB higher PSNR than the KOALANet. Fig. 6 compares the  $\times 4$  SR results of the Baseline and the KOALANet. With the predicted degradation kernel information, the KOALA modules help to effectively remove the blur induced by the random degradations, while revealing fine edges (2<sup>nd</sup> column). The images in the 3<sup>rd</sup> column show the SR results when incorrect larger or smaller blur kernels are deliberately provided to

Models	(a) ResNet	(b) U-Net	(c) Uniform	(d) No Norm.	Full Model
PSNR	47.79	47.85	45.78	49.15	<b>49.50</b>
SSIM	0.9967	0.9967	0.9945	0.9976	<b>0.9980</b>

Table 3: Experiment on downsampling network architecture for  $\times 1/4$ . All networks contain 27 convolution layers.

$l_2$ error	KernelGAN[2]	BlindSR[5]	KOALAnet
DIV2K-val	0.0230	0.0152	<b>0.0010</b>
DIV2KRRK[2]	0.0084	0.0081	<b>0.0044</b>

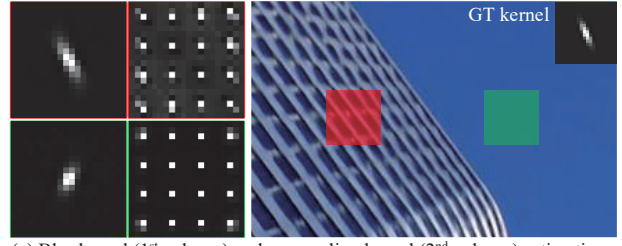
Table 4: Kernel accuracy (average  $l_2$  error) measured on the two random anisotropic datasets, DIV2K-val and DIV2KRRK [2].

the KOALA modules. In these cases, the wrong kernels cause the upsampling network to produce over-sharpened (for larger kernels) or blurry (for smaller kernels) results. All models were tested on the DIV2K-val testset.

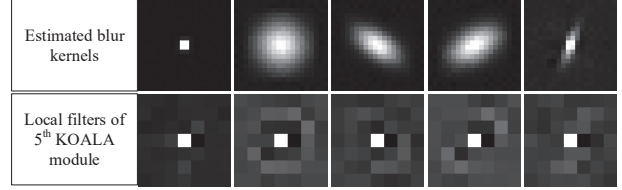
**Downsampling network architecture.** To analyze the downsampling network, we retrained its four variants: (a) a ResNet model – a ResNet-style architecture trained instead of a U-Net, (b) a model with residual connection removed from ResBlocks – thus a common U-Net, not a ResU-Net, (c) a model that estimates uniform kernels, (d) a model without normalization (sum to 1) employed for  $F_d$  (*No Norm*). Table 3 shows PSNR and SSIM values measured between the original degraded LR image and the LR image reconstructed using the kernels produced from the different models, on the DIV2K-val testset. *Full Model* in Table 3 denotes our final downsampling network (a ResU-Net), where there is a large drop in PSNR if any of the components are ablated. Especially, if spatially-equivariant kernels are estimated as in (c), PSNR performance drops drastically by 4.72 dB, showing the importance of using spatially-variant kernels.

**Degradation kernel estimation accuracy.** In order to evaluate the accuracy of degradation kernel estimation, we measured the average  $l_2$  distance between the ground truth kernels and the kernels estimated by KernelGAN [2], BlindSR [5] and the downsampler of KOALAnet, on the random anisotropic degradation testset DIV2K-val and DIV2KRRK [2] in Table 4. We make sure that the centers of the estimated kernels are aligned to the center of each ground truth kernel through manual shifting. As shown in Table 4, KOALAnet predicts more accurate degradation kernels with lower  $l_2$  error compared to other kernel estimators.

**Various kernel visualizations.** In Fig. 7 (a), we visualized the estimated blur kernels and upsampling kernels of our KOALAnet on two different locations in the same image. The 1<sup>st</sup> column shows the spatially-variant degradation kernels that are predicted by the downsampling network. As discussed in Section 3.1, the predicted blur kernel is close to



(a) Blur kernel (1<sup>st</sup> column) and upsampling kernel (2<sup>nd</sup> column) estimation.



(b) Examples of estimated blur kernels  $F_d$  and the corresponding  $7 \times 7$  local filters  $k$  of the 5<sup>th</sup> KOALA module.

Figure 7: Various kernel visualizations of KOALAnet.

the true kernel in the complex area (*red box*), while a non-directional kernel is obtained in the homogeneous region (*green box*). In the 2<sup>nd</sup> column, the  $s \times s$  2D upsampling kernels of size  $5 \times 5$  are also shown to be non-uniform depending on the location. We have also visualized some examples of local filters,  $k$ , of the KOALA modules in Fig. 7 (b). The top row shows the degradation kernels estimated by the downsampling network, and the bottom row shows the  $7 \times 7$  local filters ( $k$ ) of the 5<sup>th</sup> KOALA module. Even without any explicit enforcement on the shape of  $k$ , they are learned to be related to the orientations and shapes of the blur kernel, and thus able to adjust the SR features accordingly.

## 5. Conclusion

Blind SR is an important step towards generalizing learning-based SR models for diverse types of degradations and content of LR data. In order to achieve this goal, we designed a downsampling network that predicts spatially-variant kernels and an upsampling network that leverages this information effectively, by applying these kernels as local filtering operations to modulate the early SR features based on the degradation information. As a result, our proposed KOALAnet accurately predicts the HR images under a randomized synthetic setting as well as for historic data. Furthermore, we first analyzed the SR results on real aesthetic photographs, for which our KOALAnet appropriately handles the intentional blur unlike other methods or the Baseline. Our code and data are publicly available on the web.

**Acknowledgement.** This work was supported by Institute for Information & communications Technology Promotion (IITP) grant funded by the Korea government (MSIT) (No. 2017-0-00419, Intelligent High Realistic Visual Processing for Smart Broadcasting Media).



## References

- [1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1122–1131, 2017.
- [2] Sefi Bell-Kligler, Assaf Shocher, and Michal Irani. Blind super-resolution kernel estimation using an internal-gan. In *Advances in Neural Information Processing Systems*, pages 284–293, 2019.
- [3] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie-Line Morel. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In *Proceedings of the British Machine Vision Conference*, pages 135.1–135.10, 2012.
- [4] Xi Cheng, Zhenyong Fu, and Jian Yang. Zero-shot image super-resolution with depth guided internal degradation learning. In *Proceedings of the European Conference on Computer Vision*, pages 265–280, 2020.
- [5] Victor Cornillère, Abdelaziz Djelouah, Wang Yifan, Olga Sorkine-Hornung, and Christopher Schroers. Blind image super-resolution with spatially variant degradations. *ACM Transactions on Graphics*, 38(6):166.1–166.13, 2019.
- [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*, pages 184–199. Springer, 2014.
- [7] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 315–323, 2011.
- [8] Jinjin Gu, Hannan Lu, Wangmeng Zuo, and Chao Dong. Blind super-resolution with iterative kernel correction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1604–1613, 2019.
- [9] Yong Guo, Jian Chen, Jingdong Wang, Qi Chen, Jiezhong Cao, Zeshuai Deng, Yanwu Xu, and Mingkui Tan. Closed-loop matters: dual regression networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5407–5416, 2020.
- [10] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5197–5206, 2015.
- [11] Xu Jia, Bert De Brabandere, Tinne Tuytelaars, and Luc V Gool. Dynamic filter networks. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, pages 667–675, 2016.
- [12] Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, and Seon Joo Kim. Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3224–3232, 2018.
- [13] 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*, pages 1646–1654, 2016.
- [14] 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 Conference on Computer Vision and Pattern Recognition*, pages 624–632, 2017.
- [15] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, 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*, pages 4681–4690, 2017.
- [16] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 136–144, 2017.
- [17] David Martin, Charles Fowlkes, Doron Tal, and Jitendra Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 416–423, 2001.
- [18] 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, 2015.
- [19] Simon Niklaus, Long Mai, and Feng Liu. Video frame interpolation via adaptive convolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 670–679, 2017.
- [20] Simon Niklaus, Long Mai, and Feng Liu. Video frame interpolation via adaptive separable convolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 261–270, 2017.
- [21] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Proceedings of the Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, 2015.
- [22] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1874–1883, 2016.
- [23] Assaf Shocher, Nadav Cohen, and Michal Irani. “zero-shot” super-resolution using deep internal learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3118–3126, June 2018.
- [24] Neal Wadhwa, Rahul Garg, David E. Jacobs, Bryan E. Feldman, Nori Kanazawa, Robert Carroll, Yair Movshovitz-Attias, Jonathan T. Barron, Yael Pritch Knaan, and Marc Levoy. Synthetic depth-of-field with a single-camera mobile phone. *ACM Transactions on Graphics*, 37(4):64.1–64.13, 2018.
- [25] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep

- spatial feature transform. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 606–615, June 2018.
- [26] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *Proceedings of the European Conference on Computer Vision*, pages 1–15, 2018.
  - [27] Yu-Syuan Xu, Shou-Yao Roy Tseng, Yu Tseng, Hsien-Kai Kuo, and Yi-Min Tsai. Unified dynamic convolutional network for super-resolution with variational degradations. *arXiv preprint arXiv:2004.06965*, 2020.
  - [28] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In *Proceedings of the International Conference on Curves and Surfaces*, pages 711–730, 2010.
  - [29] Kai Zhang, Wangmeng Zuo, and Lei Zhang. Learning a single convolutional super-resolution network for multiple degradations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3262–3271, 2018.
  - [30] 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*, pages 286–301, 2018.
  - [31] Ruofan Zhou and Sabine Susstrunk. Kernel modeling super-resolution on real low-resolution images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2433–2443, 2019.