Better “CMOS” Produces Clearer Images: Learning Space-Variant Blur Estimation for Blind Image Super-Resolution

Xuhai Chen\(^1\) Jiangning Zhang\(^2\)* Chao Xu\(^1\) Yabiao Wang\(^2\) Chengjie Wang\(^2\) Yong Liu\(^1\)†
\(^1\) APRIL Lab, Zhejiang University \(^2\) YouTu Lab, Tencent
\{22232044, 21832066\}@zju.edu.cn, yongliu@iipc.zju.edu.cn
\{vtzhang, caseywang, jasoncjwtwang\}@tencent.com

Abstract

Most of the existing blind image Super-Resolution (SR) methods assume that the blur kernels are space-invariant. However, the blur involved in real applications are usually space-variant due to object motion, out-of-focus, etc., resulting in severe performance drop of the advanced SR methods. To address this problem, we firstly introduce two new datasets with out-of-focus blur, i.e., NYUv2-BSR and Cityscapes-BSR, to support further researches of blind SR with space-variant blur. Based on the datasets, we design a novel Cross-MOdal faSiOn network (CMOS) that estimate both blur and semantics simultaneously, which leads to improved SR results. It involves a feature Grouping Interactive Attention (GIA) module to make the two modalities interact more effectively and avoid inconsistency. GIA can also be used for the interaction of other features because of the universality of its structure. Qualitative and quantitative experiments compared with state-of-the-art methods on above datasets and real-world images demonstrate the superiority of our method, e.g., obtaining PSNR/SSIM by \(+1.91%/+0.0048%\) on NYUv2-BSR than MANet\(^1\).

1. Introduction

Blind image SR, with the aim of reconstructing High-Resolution (HR) images from Low-Resolution (LR) images with unknown degradations, has attracted great attention due to its significance for practical use [2, 5, 6, 12, 15, 22–24, 29]. Two degradation models, bicubic downsampling [35] and traditional degradation [26, 32], are usually used to generate LR images from HR images. The latter can be modeled by:

\[
y = (x \boxtimes k) \downarrow_s + n. \tag{1}\]

It assumes the LR image \(y\) is obtained by first convolving the HR image \(x\) with a blur kernel \(k\), followed by a down-sampling operation with scale factor \(s\) and an addition of noise \(n\). On top of that, some works [38, 48] propose more complex and realistic degradation models, which also assume that blur is space-invariant. However, in real-world applications, blur usually changes spatially due to factors such as out-of-focus and object motion, so that the mismatches will greatly degrade the performance of existing SR methods. Fig. 1 gives an example when the LR image suffers from space-variant blur. Since both KernelGAN [1] and DCLS [28] estimate only one blur kernel for an image, there are a lot of mismatches. In the first row of Fig. 1, where the kernel estimated by the two methods are sharper than the real one of the patch, SR results are over smoothing and high frequency textures are significantly blurred. In the second row, where the kernels estimated are smoother than the correct one, SR results show ringing artifacts caused by over-enhancing high-frequency edges. This phenomenon illustrates that mismatch of blur will significantly affect SR results, leading to unnatural outputs. In this paper, we focus on the space-variant blur estimation to ensure that the estimated kernel is correct for each pixel in the images.

A few recent works [15, 23, 43] have taken space-variant blur into account. Among them, MANet [23] is the most representative model, which assumes that blur is space-invariant within a small patch. Based on this, MANet uses a moderate receptive field to keep the locality of degradations. However, there are still two critical issues. 1) Because there...
is no available dataset containing space-variant blur in SR field, MANet is trained on space-invariant images, resulting in blur deviation of the training and testing phase. 2) Even limiting the size of the receptive field, the estimation results are still poor at the boundaries of different kernels, leading to mean value prediction of space-variant blur.

To address the aforementioned challenges, we first introduce a new degradation method and propose two corresponding datasets, i.e., NYUv2-BSR and Cityscapes-BSR to support relevant researches of space-variant blur in the SR domain. As a preliminary exploration, out-of-focus blur is studied as an example in this paper and it is generated according to the depth of the objects using the method proposed in [19]. Besides, we also add some space-invariant blur into the datasets so that the models trained on them can cope with both spatially variant and invariant situations.

Furthermore, to improve the performance at the boundaries of different blur regions, we present a novel model named Cross-MODal RaSion network (CMOS). Our intuition is that the sharp semantic edges are usually aligned with out-of-focus blur boundaries and it can help to distinguish different blur amounts. This raises a critical concern that how to effectively introduce semantics into the process. Specifically, we firstly predict blur and semantics simultaneously instead of using the semantics as an extra input, which not only avoids using extra information during test phase, but also enables non-blind SR methods to recover finer textures with the two modalities. Secondly, to enhance accuracy at the blur boundaries, we conduct interaction between the semantic and blur features for complementary information learning inspired by multi-task learning [36, 42]. However, in some cases these two modalities are inconsistent. As shown in Fig. 2, the wall and the picture on it are completely different in the semantic map, with clear boundaries. But the depth of them are almost the same, so the blur amounts depending on depth are also very similar. In this case, not only can the two modalities fail to use common features, but they can also negatively influence each other. Besides, since we add some space-invariant blurred images with uniform blur maps in the datasets, it will also greatly increase the inconsistency.

Motivated by these observations, we propose a feature Grouping Interactive Attention (GIA) module to help the interaction of the two modalities. GIA has two parallel streams: one operating along the spatial dimension and the other along the channel dimension. Both streams employ group interactions to process the input features and make adjustments. Moreover, GIA has an upsampling layer based on the flow field [21] to support inputs of different resolutions. Its universal structure allows it to be used for more than just interactions between the two modalities.

The main contributions of this work are as follows:

- To support researches on space-variant blur in the field of SR, we introduce a new degradation model of out-of-focus blur and propose two new datasets, i.e., NYUv2-BSR and Cityscapes-BSR.
- We design a novel model called CMOS for estimating space-variant blur, which leverages extra semantic information to improve the accuracy of blur prediction. The proposed GIA module is used to make the two modalities interact effectively. Note that GIA is universal and can be used between any two features.
- Combined with existing non-blind SR models, CMOS can estimate both space-variant and space-invariant blur and achieve SOTA SR performance in both cases.

2. Related Work

2.1. Degradation Model

SR methods give rise to poor performance if the assumed degradation deviates from those in reality. Many works [4, 45, 49] use the traditional model (Eq. 1) to generate their training data. Compared to bicubic downsampling [40, 50], although traditional model has taken more factors into account, it is still too simple to simulate real degradation. Consequently, Real-ESRGAN [38] proposes a flexible high-order degradation model by applying traditional model repeatedly, while BSRGAN [48] adjusts the degradation order of the traditional model and use randomly shuffled blur, downsampling and noise. Liang et al. [23] go a step further to simulate space-variant blur by dividing images into patches and applying different kernels. Unfortunately, it cannot well simulate the real situations. As a result, to support relevant researches, we introduce space-variant out-of-focus blur into SR, and propose two corresponding datasets, i.e., NYUv2-BSR and Cityscapes-BSR.

2.2. Kernel Estimation

One of the mainstream methods of blind SR is to estimate degradation first and then use it as prior information for non-blind SR. KernelGAN [1] proposes to learn a kernel from the internal distribution of image patches, while IKC [6] uses an iterative correction scheme to learn the PCA features of kernels. Luo et al. [28] transfer blur estimation into LR space and learn kernel weights instead of kernel itself. However, these methods only estimate a unique kernel,
Figure 3. Original RGB images, the generated out-of-focus images and blur maps. The changes from dark to light in blur maps indicate that the corresponding out-of-focus image changes from clear to blur. The first three columns are images from NYUv2-BSR, and the last three columns are images from Cityscapes-BSR.

thus the performance will be significantly reduced on space-variant situations. Accordingly, KOALAnet [15] proposes to learn specific kernels for each pixel, and MANet [23] designs a network with moderate receptive field to adapt to the locality of degradation. However, they still have limitations, such as the moderate receptive field might limit the capacity of the model. By contrast, with the help of semantic information, our CMOS can predict space-variant blur effectively and accurately.

2.3. Non-blind SR

Non-blind SR aims to restore images with known degradations. Early non-blind SR methods [13, 14, 18, 25] are based on bicubic downsampling, which struggle to generalize to images with more complex degradations. To address this problem, SRMD [49] first proposes to stretch the blur and noise to the size of LR images, and take the concatenated images and degradation maps as input to restore the HR counterparts. Following SRMD, SFTMD [6] uses SFT layer [39] to combine the stretching degradation maps instead of simply concatenation, while UDVD [44] employs per-pixel dynamic convolution to more effectively deal with variational degradations across images. Besides, zero-shot methods [11, 32, 34] have also been investigated in non-blind SR with multiple degradations. What is noteworthy is that our CMOS can be easily combined with most non-blind SR methods to achieve excellent blind SR performance.

3. The Proposed Datasets

To support researches on space-variant blur, we propose two novel datasets, NYUv2-BSR and Cityscapes-BSR, where BSR stands for Blind image SR. To the best of our knowledge, we are the first to introduce out-of-focus, one of the most common space-variant blur in real world, into blind image SR. Out-of-focus is caused by differences in depth. Every point that is not in the plane of focus corre-

<table>
<thead>
<tr>
<th>Dataset</th>
<th>NYUv2-BSR</th>
<th>Cityscapes-BSR</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>VA</td>
<td>IVA</td>
</tr>
<tr>
<td>Train</td>
<td>636</td>
<td>159</td>
</tr>
<tr>
<td>Val</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Test</td>
<td>524</td>
<td>130</td>
</tr>
</tbody>
</table>

Table 1. Details of NYUv2-BSR and Cityscapes-BSR. VA and IVA represents the number of images with space-variant out-of-focus blur and space-invariant blur respectively.

sponds to a Circle Of Confusion (COC) in image plane. The blur can be simulated by isotropic Gaussian kernels with standard deviation \( \sigma \) related to the diameters of COCs [17], which can be calculated using thin lens model [30]. We employ the method proposed in [19] to blur the images and the ground truth blur map is constructed by \( \sigma \) of each pixel.

As mentioned above, we need depth-color image pairs to generate images with out-of-focus blur. Thus, we select NYUv2 [33] and Cityscapes [3] as original datasets. NYUv2 is an indoor dataset. It contains 1449 pairs of RGB and depth images, in which 795 pairs are used for training and the rest 654 for testing. Cityscapes is an outdoor dataset and the fine-annotated part consists of training, validation and test sets containing 2975, 500, and 1525 images, respectively. Since the depth maps in Cityscapes contain invalid measurements, which are not conducive to the generation of out-of-focus images, we use CREStereo [20], a deep learning-based stereo matching method, to generate disparity maps and calculate the corresponding depth maps based on the camera parameters. Fig. 3 shows the original RGB images of NYUv2 and Cityscapes, as well as the generated out-of-focus images and corresponding blur maps.

In terms of parameters of the isotropic Gaussian kernels, the kernel width range is set to \([0.0, 5.0]\) and \([0.0, 15.0]\) for NYUv2 and Cityscapes, respectively. The kernel size is fixed to \(21 \times 21\) and \(61 \times 61\), and the downsampling scale...
factor is set to 4. Besides, 1/4 of the images are blurred by space-invariant kernels, so that the models trained on the datasets are not limited by the space-variant situations. Tab. 1 shows the details. In addition, to ensure the adequacy and fairness of experiments, we created five test groups for each dataset, in which each group had a different 1/4 of the images blurred by space-invariant kernels.

4. Method

As stated before, sharp semantic edges can increase the accuracy of space-variant blur estimation near the boundaries. Motivated by this, we propose a Cross-MOdal fuSion network (CMOS) to predict both blur and semantic maps simultaneously by mutual supervision of them.

4.1. Overview

Inspired by [36], CMOS is a multi-scale network, which consists of three main stages, as shown in Fig. 4. In the first stage, a fully convolutional encoder capable of generating multi-scale features is used to extract deep features \( \{F^0, F^1, \ldots, F^n\} \). In the next stage, for each scale \( i \), we apply two task-specific heads, \( \text{head}_b^i \) and \( \text{head}_s^i \), to predict initial blur and semantic features \( F^i_{\text{blur}} \) and \( F^i_{\text{seg}} \). Then, we use a proposed GIA\(_m^i\) module to achieve effective information interaction between the two modalities to obtain more accurate features \( \hat{F}^i_{\text{blur}} \) and \( \hat{F}^i_{\text{seg}} \) in a mutually supervised manner, formulated as:

\[
F^i_{\text{blur}} = \text{head}_b^i(F^i_b), \quad F^i_{\text{seg}} = \text{head}_s^i(F^i_s), \quad \hat{F}^i_{\text{blur}} \oplus \hat{F}^i_{\text{seg}} = \text{GIA}_m^i(F^i_{\text{blur}}, F^i_{\text{seg}}),
\]

where \( F^i_b \) and \( F^i_s \) denotes the input of the task-specific heads. To make better use of the multi-scale information, we use GIA\(_b^i\) and GIA\(_s^i\) to fuse the adjacent low-scale features, so the input of the heads can be written as:

\[
F^0_b = F^0_s = F^0, \quad F^i_b = \text{Sum}(\text{GIA}_b^i(F^i_b, \hat{F}^{i-1}_{\text{blur}})), \quad F^i_s = \text{Sum}(\text{GIA}_s^i(F^i_s, \hat{F}^{i-1}_{\text{seg}})),
\]

where \( \text{Sum}(\cdot) \) represents for adding outputs of the modules. At the highest resolution \( n \), the task-specific features are fed into two convolution layers to generate auxiliary blur and semantic maps for additional supervision, which is beneficial to further improve the accuracy of the final prediction.

The last stage consists \( n + 1 \) GIA\(_l^i\) modules to get the final features \( \hat{F}^i_B \) and \( \hat{F}^i_S \) of each scale as:

\[
\hat{F}^i_B, \hat{F}^i_S = \text{GIA}_l^i(F^i_{\text{blur}}, F^i_{\text{seg}}).
\]

These features are then concatenated and convolved to obtain the prediction of blur and semantic maps. In this way, we can build a shorter way for each scale to the supervision and further facilitate the interaction between blur and semantics. Besides, since blur is done in the HR space, we upsample the outputs using bi-linear interpolation by scale factor \( s \).

4.2. Grouping Interactive Attention Module

GIA is designed to help blur and semantics interact more effectively and avoid inconsistency. Besides, it can also be used for other features because of the universal structure. GIA involves two parallel streams operating on spatial and channel dimensions, and it can handle inputs of different resolutions by using a flow-based upsample module [21].

Spatial Grouping Feature Interaction. The input features may be similar on most patches, but different on some. As shown in Fig. 2, the picture hanging on the wall brings difference between the blur and semantic maps. As a result,
we propose to adjust the spatial weight map in the general spatial attention [7, 46, 47] mechanism to take advantage of similar information and avoid inconsistencies.

In the top half of Fig. 4 (b), each input is first passed through a convolution layer and divided into windows denoted by $F_{w}$. These windows are then further processed by another convolution layer before being fed into the feature interaction module (last part of Sec. 4.2). The spatial adjusting weight map $M_a \in \mathbb{R}^{1 \times H \times W}$ can be obtained by a $1 \times 1$ convolution layer after the interaction. Additionally, each input has its own spatial weight map $M^i_a \in \mathbb{R}^{1 \times H \times W}$ extracted from the windows by another $1 \times 1$ convolution layer directly. Thus, the outputs $F^j$ corresponding to the two inputs can be expressed as:

$$F^j = \text{Mul}(F_{w}^j, \text{Add}(M^o_{a}, \alpha M^j_{a})), j = 1, 2, \quad (9)$$

where $\alpha$ is a learnable parameter. Finally, windows are restored as features and final output is obtained by smoothing out possible seams with a layer of $3 \times 3$ convolution.

**Channel Grouping Feature Interaction.** As spatial feature interaction concentrates on local details, we further introduce channel grouping feature interaction to calibrate global information inspired by [8]. Firstly, we transfer the input $F^j_{in}$ to channel-wised attention vector $A^j_{in} \in \mathbb{R}^{C}$ by applying global average pooling and an MLP layer. Then, the vectors are fed into a feature interaction module, and two adjusting attention vectors $A^j_{a} \in \mathbb{R}^{C}$ integrating the two features are obtained through another MLP layer. Similar to the spatial one, the final outputs can be obtained by:

$$F^j = \text{Mul}(F_{in}^j, \text{Add}(A^j_{in}, \beta A^j_{a})), j = 1, 2 \quad (10)$$

where $\beta$ is a learnable parameter. Since global information is important for both blur [31] and semantic estimation [27], feature interaction of channel dimension is also essential.

**Feature Group Interaction.** This module is designed to interact spatial or channel features in groups. For spatial interaction, the input size is $C \times H \times W$. We regard the features of each pixel as a group, and the size of the grouped features is $N \times D$, where $N = HW$, $D = C$. For channel interaction, the input size is $C$. It will be divided into $N$ groups with length $D$, where $C = ND$. In this way, both spatial and channel inputs can be represented as $G_i \in \mathbb{R}^{N \times D}, i = 1, 2$ after grouping, where $i$ represents two different inputs. Then, we use inner product for feature interaction and get the interactive feature $F_{fuse} \in \mathbb{R}^{N \times N}$,

$$F_{fuse} = G_1 G_2^T. \quad (11)$$

After that, for spatial interaction, one of the output can be obtained by reshaping $F_{fuse}$ to $H \times W \times N$, and the other can be obtained by reshaping $F_{fuse}$ to $N \times H \times W$. For channel interaction, the two final outputs are the same and can be both obtained by simply flatten $F_{fuse}$.

### 4.3. Loss Function

We use the mean absolute error (MAE) for blur estimation and the cross-entropy (CE) loss for semantic segmentation. As shown in Fig. 4 (a), the auxiliary loss $L_1$ and loss $L_3$ are both MAE, while the auxiliary loss $L_2$ and loss $L_4$ are both CE, specifically:

$$L_1 = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \| B_{i,j} - \hat{B}_{i,j} \|_1 \quad (12)$$

$$L_2 = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} \sum_{c=1}^{C} S^c_{i,j} \log(\hat{S}^c_{i,j}) \quad (13)$$

where $B_{i,j}$ and $\hat{B}_{i,j}$ denote the estimated blur map and the corresponding ground-truth at position $(i, j)$. Similarly, $\hat{S}_{i,j}^c$ and $S_{i,j}^c$ represent the estimated semantic map and the ground-truth at position $(i, j)$ of the $c$-th category. $C$ is the number of object categories, and $H, W$ are the height and width of the maps. We do not adopt a particular loss weighing strategy, but simply sum the losses together,

$$L = L_1 + L_2 + L_3 + L_4 \quad (14)$$

### 5. Experiments

#### 5.1. Experimental Setup

**Settings of CMOS.** We select HRNet [37] as our backbone and change the stride of the first two convolutions to 1. This translates to 4 scales of the input LR images (1, 1/4, 1/8, 1/16). The task-specific heads are implemented as two basic residual blocks [9]. As for semantic segmentation, we use the official 40 classes for NYUv2-BSR and 19 classes for Cityscapes-BSR. All our experiments are conducted with the pre-trained ImageNet weights.

**Settings of Non-Blind SR.** For non-blind SR, we use RRDB-SFT proposed in [23]. To feed both blur and semantics into it, we use a GIA module. Finally, we fine-tune RRDB-SFT on blur and semantic maps estimated by CMOS. The loss between SR and HR images is also MAE.

**Implementation Details.** The image sizes are selected as $640 \times 480$ for both NYUv2-BSR and Cityscapes-BSR. We augment the training data by scaling with a randomly selected ratio in $\{1, 1.2, 1.5\}$ and the blur values are divided by the ratio. We also flip the training samples with a possibility of 0.5. Adam optimizer [16] with $\beta_1 = 0.9$ and $\beta_2 = 0.99$ is used to train the model for 700 epochs, with a batch size of 8. The learning rate is initialized as 0.0001 and a cosine learning rate schedule with 10 warm-up epochs is adopted. Implemented with PyTorch, it takes about 28 hours to train CMOS on an RTX 3090 GPU.

**Evaluation Metrics.** For blur estimation, we use PSNR and SSIM [41]. For semantic segmentation, we use mIoU. For the final SR images generated by RRDB-SFT with the
### Table 2. Average PSNR/SSIM of different methods for spatially variant blind SR on NYUv2-BSR. Avg. represents the average results on the 5 test groups. The best and second best results are highlighted in red and blue colors, respectively.

<table>
<thead>
<tr>
<th>Method</th>
<th>Group1</th>
<th>Group2</th>
<th>Group3</th>
<th>Group4</th>
<th>Group5</th>
<th>Avg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>KernelGAN [1]</td>
<td>23.10/0.7430</td>
<td>23.13/0.7439</td>
<td>23.18/0.7449</td>
<td>23.16/0.7449</td>
<td>23.18/0.7461</td>
<td>23.15/0.7446</td>
</tr>
<tr>
<td>KOALAnet [15]</td>
<td>27.69/0.8773</td>
<td>27.73/0.8768</td>
<td>27.60/0.8734</td>
<td>27.73/0.8754</td>
<td>27.74/0.8760</td>
<td>27.70/0.8758</td>
</tr>
<tr>
<td>DCLS [28]</td>
<td>27.89/0.8799</td>
<td>27.94/0.8798</td>
<td>27.82/0.8760</td>
<td>27.91/0.8768</td>
<td>27.89/0.8781</td>
<td>27.89/0.8781</td>
</tr>
<tr>
<td>DAN [10]</td>
<td>27.90/0.8809</td>
<td>27.98/0.8808</td>
<td>27.83/0.8771</td>
<td>27.91/0.8775</td>
<td>27.88/0.8791</td>
<td>27.90/0.8791</td>
</tr>
<tr>
<td>MANet [23]</td>
<td>30.16/0.9117</td>
<td>30.20/0.9111</td>
<td>30.07/0.9095</td>
<td>30.07/0.9099</td>
<td>30.10/0.9107</td>
<td>30.12/0.9106</td>
</tr>
<tr>
<td>CMOS(ours)</td>
<td>32.09/0.9168</td>
<td>32.08/0.9159</td>
<td>31.99/0.9145</td>
<td>31.96/0.9147</td>
<td>32.01/0.9153</td>
<td>32.03/0.9154</td>
</tr>
<tr>
<td>Upper Bound</td>
<td>33.80/0.9309</td>
<td>33.78/0.9305</td>
<td>33.69/0.9290</td>
<td>33.73/0.9301</td>
<td>33.74/0.9298</td>
<td>33.75/0.9300</td>
</tr>
</tbody>
</table>

### Table 3. Average PSNR/SSIM of different methods for spatially variant blind SR on Cityscapes-BSR. Avg. represents the average results on the 5 test groups. Note that, there is no official ground truth semantic maps for the test sets of Cityscapes [3], so the upper bound is not available here. The best and second best results are highlighted in red and blue colors, respectively.

<table>
<thead>
<tr>
<th>Method</th>
<th>Group1</th>
<th>Group2</th>
<th>Group3</th>
<th>Group4</th>
<th>Group5</th>
<th>Avg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>KernelGAN [1]</td>
<td>28.96/0.8461</td>
<td>29.02/0.8475</td>
<td>28.88/0.8464</td>
<td>28.96/0.8468</td>
<td>28.99/0.8477</td>
<td>28.96/0.8469</td>
</tr>
<tr>
<td>KOALAnet [15]</td>
<td>32.40/0.9173</td>
<td>32.45/0.9177</td>
<td>32.28/0.9151</td>
<td>32.44/0.9168</td>
<td>32.38/0.9166</td>
<td>32.39/0.9167</td>
</tr>
<tr>
<td>DCLS [28]</td>
<td>32.41/0.9174</td>
<td>32.46/0.9176</td>
<td>32.28/0.9151</td>
<td>32.44/0.9168</td>
<td>32.38/0.9166</td>
<td>32.39/0.9167</td>
</tr>
<tr>
<td>DAN [10]</td>
<td>32.33/0.9162</td>
<td>32.38/0.9165</td>
<td>32.21/0.9140</td>
<td>32.36/0.9156</td>
<td>32.30/0.9155</td>
<td>32.32/0.9156</td>
</tr>
<tr>
<td>MANet [23]</td>
<td>34.24/0.9293</td>
<td>34.29/0.9294</td>
<td>34.16/0.9273</td>
<td>34.27/0.9288</td>
<td>34.27/0.9285</td>
<td>34.25/0.9287</td>
</tr>
<tr>
<td>CMOS(ours)</td>
<td>35.58/0.9388</td>
<td>35.61/0.9389</td>
<td>35.50/0.9373</td>
<td>35.60/0.9385</td>
<td>35.60/0.9381</td>
<td>35.58/0.9383</td>
</tr>
</tbody>
</table>

### Table 4. Importance of using space-variant blur for training. IVA stands for the NYUv2 dataset with only space-invariant blur.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>PSNR↑</th>
<th>SSIM↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>IVA</td>
<td>19.50</td>
<td>0.6840</td>
</tr>
<tr>
<td>NYUv2-BSR</td>
<td><strong>30.12</strong></td>
<td><strong>0.9106</strong></td>
</tr>
</tbody>
</table>

### 5.2. Comparison with the State-of-the-Arts

We compare CMOS with existing blind SR models: KernelGAN [1], KOALAnet [15], DCLS [28], DAN [10], MANet [23] and the upper bound model (RRDB-SFT given ground-truth blur and semantic maps). We retrained all the comparison methods on NYUv2-BSR and Cityscapes-BSR using their official implementations and settings. KernelGAN is an unsupervised method which trained solely on the LR image at test time. DCLS and DAN are end-to-end methods for space-invariant blur, while KOALAnet and MANet are two-stage methods for space-variant blur. Since we use the non-blind SR model proposed in MANet (i.e., RRDB-SFT), we apply same settings to ensure the fairness. **Quantitative comparison.** As shown in Tab. 2 and Tab. 3, CMOS leads to the best performance for different test groups in both the two proposed datasets. Notably, methods that estimate only one blur kernel for an image (i.e., KernelGAN, DCLS, and DAN) all suffer from severe performance drop when the real kernels are spatially variant. Although KOALAnet estimates different kernels for different image pixels, it does not include any special handling for space-variant properties and also produces unfavorable results. MANet takes the locality of blur into account, so it performs relatively better. By contrast, the proposed model CMOS effectively utilizes semantic information to help with spatially variant blur estimation and non-blind SR, outperforming MANet by large margins. **Qualitative comparison.** We present several representative visual samples in Fig. 5. It can be observed that our CMOS outperforms previous approaches in both removing blur and avoiding artifacts. Other methods may either produce ringing artifacts (especially KernelGAN), or fail to restore texture details, leading the patches still blurry.

### 5.3. Ablation Study

All the experiments in this section use NYUv2-BSR for training, and the metrics (i.e., PSNR, SSIM and mIoU) refer to the mean value across the 5 test sets (Sec. 3).

**Importance of Using Space-Variant Blur for Training.** According to [23], because of the moderate receptive field, MANet can handle spatially variant cases even if it is trained on spatially invariant blurred images. But we believe that it is necessary to use the images containing space-variant blur for training. To prove it, we trained two MANet...
Figure 5. Qualitative comparisons between different SR methods on spatially variant blur (out-of-focus). The first two pictures are from NYUv2-BSR and the last two are from Cityscapes-BSR. (Please zoom in for better view.)

<table>
<thead>
<tr>
<th>Method</th>
<th>PSNR ↑</th>
<th>SSIM ↑</th>
<th>mIoU ↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ours w/o GIA</td>
<td>23.21</td>
<td>0.8312</td>
<td>32.15</td>
</tr>
<tr>
<td>Ours w/ F</td>
<td>23.42</td>
<td>0.8314</td>
<td>33.04</td>
</tr>
<tr>
<td>Ours w/ F+C</td>
<td>24.24</td>
<td>0.8336</td>
<td>36.25</td>
</tr>
<tr>
<td>Ours w/ F+C+S (GIA)</td>
<td>24.52</td>
<td>0.8340</td>
<td>35.61</td>
</tr>
</tbody>
</table>

Table 5. Effectiveness of GIA. Note that these are the intermediate results, and PSNR/SSIM refer to the blur maps rather than the final SR images. mIoU evaluates the effect of semantic estimation.

Effectiveness of GIA Module. We take out the components, i.e., flow-based upsampling (F), channel interaction (C) and spatial interaction (S), of GIA to verify validity. We record the best PSNR and mIoU models individually. As shown in Tab. 5, using only flow-based upsampling improves the results slightly, and when combined with channel interaction, the performance can be significantly enhanced. Furthermore, utilizing all three components, i.e., the complete GIA module, can yield even greater improvements.

Effectiveness of Semantic Information in SR. In order to illustrate that the semantic information is conducive to SR, we ablate it and only input blur maps into RRDB-SFT. It is worth noting that we use the ground truth blur and semantic maps here. As shown in Tab. 6, adding semantic maps improves the PSNR (+0.34 dB ↑) and SSIM (+0.0022 ↑) of the final SR results. We hold the opinion that semantic information may allow the network to take advantage of textural features of related objects it has learned about, and sharp semantic edges may also be helpful in SR.

Effectiveness of Multi-task Learning (MTL). To demonstrate the effectiveness of MTL, firstly, we make separate predictions for blur and semantic maps and compared them
Figure 6. Visual results on real-world images for scale factor 4. The first picture of the indoor scene uses the model trained on NYUv2-BSR, and the second picture of the outdoor scene uses the model trained on Cityscapes-BSR. (Please zoom in for better view.)

<table>
<thead>
<tr>
<th>Method</th>
<th>PSNR ↑</th>
<th>SSIM ↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>RRDB-SFT w/o semseg</td>
<td>33.41</td>
<td>0.9278</td>
</tr>
<tr>
<td>RRDB-SFT w/ semseg</td>
<td>33.75</td>
<td>0.9300</td>
</tr>
</tbody>
</table>

Table 6. Importance of using semantic information in SR. The ground-truth blur and semantic maps are used in this experiment.

<table>
<thead>
<tr>
<th>Method</th>
<th>Intermediate Results</th>
<th>SR Results</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>PSNR/SSIM ↑</td>
<td>mIoU ↑</td>
</tr>
<tr>
<td>Single Task</td>
<td>24.58/0.8393</td>
<td>33.95</td>
</tr>
<tr>
<td>CMOS (Ours)</td>
<td>24.52/0.8340</td>
<td><strong>35.61</strong></td>
</tr>
</tbody>
</table>

Table 7. Effectiveness of MTL. PSNR/SSIM and mIoU of the intermediate results refer to the blur maps and the semantic maps.

5.4. Experiments on Real-World SR

As there is no ground-truth for real images, we only compare visual results of different methods. As shown in Fig. 6, similar to the results on our datasets, KernelGAN still generate ringing artifacts, especially in the outdoor scene. DAN, DCLS and KOALAnet all produce blurry results, while MANet performs slightly better. In comparison, CMOS can produce realistic and natural textures, and the results are the clearest.

6. Conclusion

In this paper, we introduce out-of-focus blur to SR and propose two new datasets: NYUv2-BSR and Cityscapes-BSR. Besides, we further propose a novel model CMOS to estimate the blur and semantic maps simultaneously. By incorporating semantics, we can restore finer SR results. GIA modules is used to achieve effective feature interaction in both spatial and channel dimensions. Extensive experiments on proposed datasets and real-world images demonstrate that our model can achieve SOTA performance in blind SR when integrated with existing non-blind models. **Acknowledgments:** This work is supported by the National Natural Science Foundation of China (61836015).
References


[12] Xiaozhong Ji, Yun Cao, Ying Tai, Chengjie Wang, Jilin Li, and Feiyue Huang. Real-world super-resolution via kernel estimation and noise injection. In proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops, pages 466–467, 2020. 1


38] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-esrgan: Training real-world blind super-resolution with pure synthetic data supplementary material. 1, 2


