Deep Degradation Prior for Real-World Super-Resolution

Kyungdeuk Ko¹
kdko@korea.ac.kr

Bokyeung Lee¹
bksain@korea.ac.kr

Jonghwan Hong¹
jhong2661@korea.ac.kr

David Han²
dkh42@drexel.edu

Hanseok Ko¹
hsko@korea.ac.kr

¹ Korea University
Seoul, Republic of Korea

² Drexel University
Philadelphia, USA

Abstract

Real-world Super-Resolution (SR) is a very challenging task to reconstruct a higher resolution image from a real-world image which generally has unexpected artifacts and distortions. Most of the high performance SR methods rely on availability of LR-HR paired datasets, target domain images, or degradation priors. These information, however, are usually not available in real world, thus these methods are not often practical for real world obtained images. Recent studies related to real-world SR mainly focus on constructing the LR-HR paired dataset. The methods estimate the noises and the blur kernels from real-world images to generate a new training set. However, these methods use the degradation only for dataset construction. In this paper, we propose a novel real-world SR method called Deep Degradation Prior-based SR (DDP-SR). Upon completion of training, denoising network and kernel estimation network within DDP-SR becomes capable of extracting degradation representation of any given input image. Thus, the model works regardless of whether the input image from the same or the different domain of the source images. As such, DDP-SR achieves generalization performance on images from different domain while it also outperforms the state-of-the-art methods qualitatively and quantitatively.

1 Introduction

Super-Resolution (SR) is a low-level image processing task to increase resolution and to enhance the quality of images. In general, it is assumed that Low-Resolution (LR) images are obtained from High-Resolution (HR) images through the following degradation process:

\[ I_L = (I_H \ast k) \downarrow s + n, \]  

(1)
where $I_H$ is an HR image and $I_L$ is an LR image. $k$ and $n$ represent a blur kernel and additive noise, respectively. $s$ denotes a scale factor for downsampling. The goal of SR is to find an inverse operator to convert $I_L$ to $I_H$. With the development of deep learning, many studies using LR-HR image pairs made of Eq. (1) have emerged [5, 6, 20, 24, 25, 41, 45, 51, 52]. However, HR images are generally downsampled by a bicubic kernel in these methods, and the SR models are trained by minimizing the difference between the reconstructed images and the original images as:

$$\text{arg min}_{SR} \| SR(I_L) - I_H \|. \quad (2)$$

Consequently, since these SR models are optimized for the bicubic kernel, they generalize poorly on images degraded by other causes.

Several learning-based methods have attempted to solve this problem. One of these methods applies the idea of the domain transfer. In these efforts, LR images reside in the source domain with unknown degradation while the target domain contains clean HR images. Cycle-in-cycle Generative Adversarial Network (CinCGAN), applied by Yuan et al. [47], Lugmayr et al. [27], and Kim et al. [19], sequentially applied two CycleGANs [54] with the first CycleGAN transferring the noisy LR domain to a clean LR domain, and the second CycleGAN transferring the clean LR domain to a clean HR domain. These SR models need target domain images for training, but the target domain images generally could not be acquired in real world.

Another method is to train by self-supervised learning using only test images without any training images. Zero-Shot Super-Resolution (ZSSR) [38] is trained by the internal recurrence of information within a single image. Therefore, ZSSR can be applied to images degraded from a variety of kernels, including bicubic kernels. Since the ground truth kernel of the test image is not known, ZSSR would exhibit poor performance if the predicted kernel and the actual kernel turned out to be different. To compensate for these shortcomings, a blur kernel corresponding to the test image is obtained by using kernel estimation methods such as KernelGAN [2]. With the difficulty of estimating the kernel, ZSSR implemented with KernelGAN often shows poor performance, as we have observed in our experiments.

SR methods relying on LR-HR paired dataset or prior knowledge on degradation have progressed and now are able to deliver some impressive results. However, when the process associated with transforming HR to LR, or the source of noise added in LR images are not known, SR becomes much more challenging. A typical SR method based on supervised learning by paired images is not possible in such cases. In New Trends in Image Restoration and Enhancement (NTIRE) 2020 challenge [51], track 2 of real-world SR is one of these problems without the target domain and without any prior knowledge on noise and blur kernel in LR images. The task provides real-world images taken by smartphones, not by high-performance cameras such as DSLRs. As such, the images are more noisy and blurry due to the phone’s limited hardware and simpler circuitry. RealSR [16], which is trained by creating a degradation pool composed of the noises and kernels extracted from variance based noise cropping method and KernelGAN, won the NTIRE 2020 real-world SR competition. Since the training dataset is created by reflecting the noise and blur kernel of the source domain, the SR model trained with the generated dataset is optimized for the source domain. However, the degradation pool is only used when constructing the training set, so it is difficult for RealSR to reflect the prior for actual testing. If an image with domains different from the trained source domain is tested, poor performance is achieved.

To solve the aforementioned problem in the SR task, we propose a novel method of deep degradation prior to represent the degradation. We developed a novel structure of a
deep prior generator containing a denoising network and a kernel estimation network. The output of the deep prior generator, termed deep degradation prior, is an informative feature representation that strongly corresponds dominant components of the degradation. The representation composed of noise and kernel estimations allow the SR model to generate realistic and clean SR images by an improved understanding of the degradation in the observed images. The novelty we stress is that once the training is done, the deep prior generator is capable of composing representation of degradation involved in any given image. Thus, the trained deep prior generator is capable of dealing with input image domains different from that of the training images. Our proposed deep prior generator, therefore, would generalize well to a variety of input domains. The deep prior generator is trained with the degradation pool to extract the deep degradation prior from the observation image. Our proposed network is implemented for the SR task through an adaptive method based on Adaptive Instance Normalization (AdaIN) [12, 35, 44] and channel attention [11]. We term the overall SR architecture proposed here "Deep Degradation Prior-based SR (DDP-SR)."

Our overall contributions can be summarized as follows:

- We propose a novel real-world SR method called DDP-SR to improve the performance by using the deep degradation prior corresponding to noise and kernel of observation image.
- We propose a novel deep prior generator capable of extracting degradation components of any image on the fly once its training is completed. Our design is rooted on enabling the model to generalize to any given input domain.
- We developed a novel learning architecture for the deep prior generator composed of the denoising network and the kernel estimation network.
- DDP-SR achieves higher performance both quantitatively and qualitatively than the state-of-the-art real-world SR methods in experiments using public datasets.

2 Related Work

2.1 Noise Estimation

Recently, many studies on Convolutional Neural Network (CNN)-based denoising has been conducted. Convolutional Blind Denoising Network (CBDNet) [8] is made up of a noise estimation network and a non-blind denoising network. The noise estimation network is simply composed of five layer convolutional layers without pooling layer and normalization operation and extracts noise level map from the input image. The non-blind denoising network has residual learning [10] and U-Net [37] architecture which takes both the original noisy image and the noise level map extracted by the noise estimation network. Consequently, the non-blind denoising network outputs the clean image. Adaptively Tuned Denoising Network (ATDNet) [21] has an overall similar architecture to the CBDNet, but the noise level map is applied to the denoising subnetwork differently from the CBDNet. Since the noise level map is used as input in the middle of the denoising network by gate-residual block, the gate residual block makes the denoising network adaptive to the change of the noise level. Adaptive Instance Normalization Denoising Network (AINDNet) [22] also has an architecture similar to the the ATDNet. Instead of the gate residual block to apply the noise level map, the AINDNet uses adaptive instance normalization residual block based on AdaIN. The
denoising networks mentioned above have in common that they have residual learning [48]. However, these denoising models are not suitable for real-world image noise because they learn Gaussian noise-based noise levels. Based on these previous studies, we design the deep prior generator that learns the noise information of the real-world image and apply the deep degradation prior to the SR model.

2.2 Blind Super-Resolution

Blind SR estimates the blur kernels from the dataset, and enhance the SR performance by using the blur kernels. Since the blur kernel is an important factor in SR performance, there have been some efforts focused on finding an accurate kernel. Michaeli et al. [30] first introduced the concept of the blind SR. They estimate the blur kernels based on Maximum a Posteriori (MAP). Xu et al. [46] first proposed the blind SR used deep learning. They uses GAN for the blind SR of face image and text image without the kernel estimation. Kernel Modeling Super-Resolution (KMSR) [53] uses blur kernel estimation algorithm based on dark channel prior [9, 34]. But this algorithm is very computationally heavy, and target domain images are required to construct the LR-HR pairs. Iterative Kernel Correction (IKC) [7] estimates the blur kernel using predictor network, and updates the estimated kernel using corrector network. However, IKC needs a prior to predict the accurate blur kernel. Kernel-GAN based on Internal-GAN (InGAN) [39] estimates the blur kernel. The generator simply consists of 6 convolutional layers without a non-linear activation function. The discriminator compares the patch of the input image with the patch of the image downscaled by the generator.

2.3 Real-World Super-Resolution

Real-world SR is a task that makes a blurry and noisy image taken with a real camera or smartphone into a clean HR image. Therefore, there is no LR-HR paired dataset. Furthermore, super-resolution without target domain images is a very challenging task. Many studies have been conducted to solve this problem. Super-Resolution network for Multiple Degradations (SRMD) [49] reduces the dimension of the blur kernel and then concatenates it with the noise level. The degradation representation is stretched and concatenated with the input image. SRMD uses a grid search strategy to estimate the blur kernel and noise level of the real image. However, the strategy is inefficient because it requires a lot of computation to calculate, and it is also inaccurate because it is estimating among the pre-set blur kernel and noise. Ren et al. [36] uses various downsampling, blur kernel, and noise to make synthetic LR images similar to real-world LR images. However, in order to construct these LR images, the target domain dataset is required. On the other hand, RealSR composes the LR-HR paired dataset using only the source domain dataset. However, the degradation information is used only to compose the data set, and the information is not used as an important prior. We proposed DDP-SR that reflects the deep degradation prior to the input real-world image to the SR model. Section 3 describes DDP-SR in detail.

3 Proposed Method

Figure 1 displays our proposed method called DDP-SR. The method is divided into two main parts in dealing with a situation where target domain images don’t exist. The first is
to construct the training dataset using only source images. The other is to train the network based on the constructed data.

### 3.1 Dataset Construction

In the dataset construction, the LR-HR pairs are obtained by using only source images $I_s$. First, the HR image $I_H$ is produced by using bicubic downsampling that removes noise and makes the source image sharper. The noise $n$ and the kernel $k$ of the source image can be estimated by variance based noise cropping [16] and KernelGAN. In order to generate the LR image $I_L$ with a similar domain to the source image, the noises and the blur kernels collected from the source image are randomly selected, and $I_L$ is obtained by Eq.(1) The constructed LR-HR pairs and the degradation information estimated from source images are used to train the SR model and the deep degradation prior generator.

### 3.2 Deep Degradation Prior Generator

The deep degradation prior generator consists of two modules and delivers prior information into Residual Dense Block (RDB) [52], as shown in Figure 2. One is a deep noise prior generator and the other is a deep kernel prior generator.

#### 3.2.1 Deep Noise Prior

The deep noise prior generator is designed as the CNN based on residual learning. The CNN is composed of five convolutional layers, Rectified Linear Unit (ReLU) activation function [33], and the skip connection for the residual learning. The input of the deep noise prior generator is the LR image, and the label $I_{clean} = (I_H \otimes k) \downarrow s$ is the corresponding image before applying the noise prior to the dataset construction stage. For training the deep noise prior generator, we define the noise prior loss, $L_{noise} = \|NG(I_L) - I_{clean}\|_2$, where $NG$ denotes the deep noise prior generator. By applying residual structure for denoising, CNN layers are trained to estimate residual noise. The estimated raw noise map is not adequate as prior knowledge because it is too simple information. So we use encoded feature map which contains noise information.

The deep noise prior $F_n \in \mathbb{R}^{W_I \times H_I \times C}$, where $W_I$ and $H_I$ are the width and height of the input image, and $C$ is the number of channels, is applied to the SR model through AdaIN.
injecting the deep noise prior into the SR model via AdaIN can prevent overfitting to the noise on the training set and regularize the network. For the deep noise prior, AdaIN is composed of convolutional layers and instance normalization (IN) with \( h'_j = \gamma \odot \text{IN}(h_j) + \beta \) where \( h_j \) denotes \( j \)-th mid-level features, \( \text{IN} \) means IN operation, and \( \odot \) represents element-wise multiplication. Also, \( \gamma \) and \( \beta \) are obtained through the deep noise prior and the convolutional layers as in Figure 2.

### 3.2.2 Deep Kernel Prior

The deep kernel prior generator is composed of four convolutional layers, Leaky ReLU (LReLU) activation function [29], and a global average pooling. The size of the pooling result is the same as \( W_k \times H_k \) of the blur kernel used in the dataset construction stage. After the pooling result is reshaped to the shape of the blur kernel, we use the kernel prior loss, \( L_{\text{kernel}} = \| KG(I_L) - k \|_2 \), where \( KG \) represents the deep kernel prior generator and \( k \) means the blur kernel that used when generating the degraded image \( I_L \). For deep kernel prior \( F_k \in \mathbb{R}^{1 \times 1 \times (W_k \times H_k)} \), channel attention is similar to AdaIN in the deep noise prior. Channel attention is used to apply the deep kernel prior to global areas of features. In our case, the kernel size set to \( 33 \times 33 \).

### 3.3 Deep Degradation Prior for Super-Resolution

The deep degradation prior extracted from the deep prior generator can be applied to various SR models. In this case, Enhanced Super-Resolution GAN (ESRGAN) [45] is used as the SR model. The performance of ESRGAN has not only been proven in several studies [16, 36], but also empirically showed the best in our case. The generator of ESRGAN is composed of Residual-in-Residual Dense Block (RRDB) which combines multi-level residual network and dense network. The deep degradation previously mentioned in Section 3.2 is applied to each RDB in each RRDB. Based on the inverse operation of Eq.1, the deep noise prior is applied to the beginning of the RDB and the deep kernel prior is applied to the end of RDB. For efficient learning, the pre-trained SR generator and discriminator are used. Pixel loss,
perceptual loss \[17\], and adversarial loss are used to train the generator. The pixel loss is the distance between \(I_H\) and the output of the SR model \(I_{SR}\), and the perceptual loss is the distance in the feature space of CNN. The pixel loss preserves the overall HR image content, and the perceptual loss improves high-frequency components. In this case, both the pixel loss and the perceptual loss use L1 distance, and VGG-19 \[40\] is used for perceptual loss. The adversarial loss makes the generated image more realistic by enhancing texture details. The final loss is defined as the weighted sum of the aforementioned losses as follows:

\[
L_{total} = \lambda_{per}L_{per} + \lambda_{pixel}L_{pixel} + \lambda_{noise}L_{noise} + \lambda_{kernel}L_{kernel} + \lambda_{adv}L_{adv},
\]

where \(\lambda_{per}\) and \(\lambda_{adv}\) are set as 1 and 0.005, and \(\lambda_{pixel}, \lambda_{noise},\) and \(\lambda_{kernel}\) are all set as 0.01.

Patch discriminator \[15, 54\] is used to determine whether each patch is real or fake, instead of the original discriminator of ESRGAN. The patch discriminator makes a final decision through average operation after determining whether it is real or fake for each patch of the image. Consequently, it can work regardless of the size of the image. The SR generator and the discriminator are fine-tuned for 60000 iterations by using Adam optimizer \[23\] with parameters \(\beta_1 = 0.9\) and \(\beta_2 = 0.999\) without weight decay. The learning rate is initially set to 0.0001 and halved at \{5000, 10000, 20000, 30000\} iterations.

## 4 Experiment

### 4.1 Datasets

The DSLR Photo Enhancement Dataset (DPED) \[13\] is used for real-world SR experiments. The DPED is the dataset taken with four different cameras, but only pictures taken with iPhone 3GS are used this time. The training set contains only 5614 source domain images. The test set includes 100 images cropped from the original images. Since there is no Ground Truth (GT), the performance needs to be evaluated based on the no-reference method. Therefore, Natural Image Quality Evaluator (NIQE) \[32\], Blind/Referenceless Image Spatial QUality Evaluator (BRISQUE) \[31\], Perception-based Image Quality Evaluator
Table 1: Quantitative results on the DPED dataset, RealSR-V3 dataset, DIV2K dataset, and CelebA-HQ dataset.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Method</th>
<th>PNSR↑</th>
<th>SSIM↑</th>
<th>NIQE↓</th>
<th>BRISQUE↓</th>
<th>PIQE↓</th>
<th>NRQM↑</th>
<th>PI↓</th>
<th>DISTS↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>DPED</td>
<td>Bicubic</td>
<td>5.5426</td>
<td>53.2307</td>
<td>89.4906</td>
<td>3.1220</td>
<td>6.2103</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SRMD</td>
<td>5.4809</td>
<td>52.2995</td>
<td>75.7359</td>
<td>4.0941</td>
<td>5.6934</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IKC</td>
<td>5.6268</td>
<td>51.2808</td>
<td>76.4175</td>
<td>4.2140</td>
<td>5.7064</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ZSSR</td>
<td>6.7644</td>
<td>42.1918</td>
<td>51.3600</td>
<td>5.2201</td>
<td>5.7722</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>K-ZSSR</td>
<td>10.2671</td>
<td>43.8575</td>
<td>44.2234</td>
<td>4.7748</td>
<td>7.7462</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RealSR</td>
<td>4.1223</td>
<td>23.3344</td>
<td>14.0526</td>
<td>6.5217</td>
<td>3.8003</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td>3.7137</td>
<td>21.4566</td>
<td>13.6320</td>
<td>6.6546</td>
<td>3.5296</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>RealSR-V3</td>
<td>Bicubic</td>
<td>0.8206</td>
<td>6.0751</td>
<td>53.2307</td>
<td>3.1220</td>
<td>6.2103</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SRMD</td>
<td>0.8303</td>
<td>5.1688</td>
<td>50.2295</td>
<td>4.0941</td>
<td>5.6934</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IKC</td>
<td>0.8331</td>
<td>4.9729</td>
<td>49.7129</td>
<td>4.2140</td>
<td>5.7064</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ZSSR</td>
<td>5.5090</td>
<td>6.7244</td>
<td>43.4438</td>
<td>3.1220</td>
<td>6.2103</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>K-ZSSR</td>
<td>5.6268</td>
<td>7.2699</td>
<td>43.4680</td>
<td>4.7748</td>
<td>7.7462</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RealSR</td>
<td>4.1223</td>
<td>7.2699</td>
<td>56.2521</td>
<td>4.8461</td>
<td>6.2119</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td>3.7137</td>
<td>3.3585</td>
<td>23.3344</td>
<td>6.4980</td>
<td>3.8003</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>DIV2K</td>
<td>Bicubic</td>
<td>27.2291</td>
<td>21.4566</td>
<td>13.6320</td>
<td>6.6546</td>
<td>3.5296</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SRMD</td>
<td>27.5849</td>
<td>23.3344</td>
<td>14.0526</td>
<td>6.5217</td>
<td>3.8003</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IKC</td>
<td>27.1788</td>
<td>21.4566</td>
<td>13.6320</td>
<td>6.6546</td>
<td>3.5296</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ZSSR</td>
<td>25.5300</td>
<td>21.4566</td>
<td>13.6320</td>
<td>6.6546</td>
<td>3.5296</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>K-ZSSR</td>
<td>23.1509</td>
<td>21.4566</td>
<td>13.6320</td>
<td>6.6546</td>
<td>3.5296</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RealSR</td>
<td>24.0146</td>
<td>21.4566</td>
<td>13.6320</td>
<td>6.6546</td>
<td>3.5296</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td>24.2664</td>
<td>21.4566</td>
<td>13.6320</td>
<td>6.6546</td>
<td>3.5296</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CelebA-HQ</td>
<td>Bicubic</td>
<td>34.1167</td>
<td>34.1167</td>
<td>34.1167</td>
<td>34.1167</td>
<td>34.1167</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>SRMD</td>
<td>34.4920</td>
<td>34.4920</td>
<td>34.4920</td>
<td>34.4920</td>
<td>34.4920</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>IKC</td>
<td>36.7090</td>
<td>36.7090</td>
<td>36.7090</td>
<td>36.7090</td>
<td>36.7090</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>ZSSR</td>
<td>27.6223</td>
<td>27.6223</td>
<td>27.6223</td>
<td>27.6223</td>
<td>27.6223</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>K-ZSSR</td>
<td>27.8847</td>
<td>27.8847</td>
<td>27.8847</td>
<td>27.8847</td>
<td>27.8847</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>RealSR</td>
<td>33.2764</td>
<td>33.2764</td>
<td>33.2764</td>
<td>33.2764</td>
<td>33.2764</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Ours</td>
<td>33.8797</td>
<td>33.8797</td>
<td>33.8797</td>
<td>33.8797</td>
<td>33.8797</td>
<td>-</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4.2 Experimental Results

To compare DDP-SR performance, we perform baseline experiments such as bicubic, SRMD, IKC, ZSSR, K-ZSSR which means a combination of KernelGAN and ZSSR, and RealSR that is the state-of-the-art method for real-world SR. As shown in Table 1, DDP-SR achieves the best performance in all evaluation metrics. Except for RealSR, the baseline performance is quantitatively very poor. Especially, NIQE values of ZSSR, K-ZSSR, and IKC are worse than the bicubic method. Figure 3 displays the outputs of DPP-SR and the baseline methods.
ZSSR produces the LR-HR pairs without considering the source domain, so the blurred outputs are generated. Also, K-ZSSR makes unwanted textures that look like artifacts. Real-SR shows good performance compared to other baselines, but over-sharpening occurs at edges such as letters or digits. On the other hand, DPP-SR produces relatively clean and sharpened results. In particular, DPP-SR creates brighter results that are similar to the original brightness and expresses a pattern such as a tree branch well. Consequently, it can be seen that DDP-SR shows higher performance than the state-of-the-art real-world SR methods.

To demonstrate generalization performance of our proposed method, we use the RealSR-V3 dataset, DIV2K dataset, and CelebA-HQ dataset with the networks trained with the DPED dataset. In Table 1, the quantitative results are summarized. DDP-SR shows higher performance than other methods in the no-reference evaluation metrics. On the other hand, since the degraded images were generated synthetically, IKC which is supervised learning based method shows higher performance than DDP-SR in all reference evaluation metrics. However, it can be seen that the results of IKC are more blurry than DDP-SR in Figure 4. It can be seen that distortion has occurred in the results of RealSR. In contrast, DDP-SR produces the high-quality images without distortion. Figure 5 shows the experimental results for CelebA-HQ. DDP-SR converts degraded LR face images into clear high-resolution face images. As a result, the experimental results demonstrate that our proposed method shows high generalization performance qualitatively and quantitatively.

Figure 4: Qualitative results on other dataset different from the training dataset domain: (a) RealSR-V3 dataset, (b) DIV2K dataset
4.3 Ablation Study

Several experiments are additionally conducted to analyze how deep degradation prior affects SR performance. In Table 2, the results of the ablation study are summarized. First, we compare when only the deep noise prior is used, when only the deep kernel prior is used, and when both are used. Table 2 shows better NIQE, BRISQUE, NRQM, and PI when using both than when using only one, but PIQE is highest when using only deep noise prior. Next, there is the experiment according to the position where the deep noise prior and the deep kernel prior are applied. In Table 2, front is when both priors are in front, back is when they are in the back, and reverse is when the two positions are swapped. Our proposed position achieves the best NIQE, BRISQUE, NRQM and PI, but PIQE is best at back. The last experiment is the performance evaluation according to the position where the deep noise prior is extracted from the deep noise prior generator. BRISQUE, PIQE, and NRQM are the best when extracted from Conv5. However, the result of Conv3 shows the best NIQE, PI, and visually the best results. As a result, the deep noise prior extracted from Conv3 has the highest performance both in terms of NIQE and PI, and qualitatively. The qualitative results about the ablation study are displayed in the supplementary material.

5 Conclusion

In this paper, a novel real-world SR method called DDP-SR was proposed. In DDP-SR, the deep degradation prior extracted from the deep prior generators is applied to the SR model. The SR model delivered high performance by reflecting the degradation prior. To evaluate the performance of DDP-SR, we compared with state-of-the-art methods. As a result, DDP-SR outperformed the state-of-the-art methods. Additionally, when testing with the datasets different from the domain of DPED used for the training, the DDP-SR outperformed the baseline qualitatively and quantitatively.
Acknowledgements

This research was supported by Deep Machine Lab (Q2109331).

References


