SRFeat: Single Image Super-Resolution with Feature Discrimination

Seong-Jin Park[1], Hyeongseok Son[1], Sunghyun Cho[2], Ki-Sang Hong[1],
Seungyong Lee[1]

[1]POSTECH [2]DGIST
{windray, sonhs, hongks, leesy}@postech.ac.kr scho@dgist.ac.kr

Abstract. Generative adversarial networks (GANs) have recently been adopted to single image super-resolution (SISR) and showed impressive results with realistically synthesized high-frequency textures. However, the results of such GAN-based approaches tend to include less meaningful high-frequency noise that is irrelevant to the input image. In this paper, we propose a novel GAN-based SISR method that overcomes the limitation and produces more realistic results by attaching an additional discriminator that works in the feature domain. Our additional discriminator encourages the generator to produce structural high-frequency features rather than noisy artifacts as it distinguishes synthetic and real images in terms of features. We also design a new generator that utilizes long-range skip connections so that information between distant layers can be transferred more effectively. Experiments show that our method achieves the state-of-the-art performance in terms of both PSNR and perceptual quality compared to recent GAN-based methods.

Keywords: super-resolution, adversarial network, high frequency features, perceptual quality

1 Introduction

Single image super-resolution (SISR) is a task to restore the original high-resolution (HR) image from a single low-resolution (LR) image counterpart. Successful super-resolution (SR) is of great value in that it can be effectively utilized for diverse applications such as surveillance imaging, medical imaging, and ultra high definition contents generation. However, SISR is still a challenging problem despite extensive research for decades because of its inherent ill-posedness, i.e., for a given LR image, there exist a numerous number of HR images that can be downsampled to the same LR image.

Most existing SISR approaches try to minimize pixel-wise mean squared errors (MSEs) between the super-resolved image and the target image. Minimizing pixel-wise errors inherently maximizes peak signal-to-noise ratio (PSNR), which is commonly used to compare different methods. However, it is well-known that measuring pixel-wise difference can hardly capture perceptual differences between images [49, 48, 17], thus higher PSNR does not necessarily lead to a per-

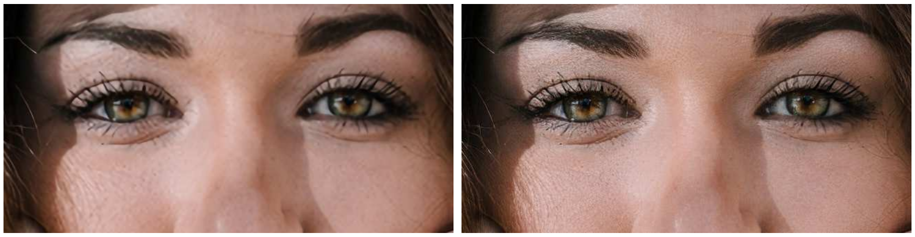


Fig. 1. Our SR results. The final result from our network trained with GAN (right) is much more perceptually realistic than the result obtained by our network trained with MSE only (left).

ceptually better image. Rather, it prefers blurry results without high-frequency details as minimization of the errors regresses to an average of possible solutions.

Recently, Goodfellow *et al.* [14] introduced a novel framework called generative adversarial network (GAN), which consists of two neural networks competing each other: a generator and a discriminator. The generator tries to fool the discriminator by generating a realistic image, while the discriminator tries to distinguish generated fake images from real ones. Joint training of these two networks leads to a generator that is able to produce remarkably realistic fake images. Thanks to its effectiveness in image generation, GAN has been widely applied to various tasks such as image synthesis, style transfer, image inpainting, and object detection [37, 55, 20, 25, 23, 19, 30, 28].

Recently GAN has also been applied to SISR to overcome the aforementioned limitation and produce super-resolved images with synthesized high-frequency details. Ledig *et al.* proposed SRGAN [27] that employs an adversarial loss term with a data term for obtaining visually-pleasing results rather than maximizing PSNR. Sajjadi *et al.* proposed EnhanceNet [40], which is also based on GAN. EnhanceNet additionally adopts a texture matching loss inspired by Gatys *et al.* [13] to encourage super-resolved results to have the same textures as the ground truth HR images.

While GAN-based SISR methods show dramatic improvements over previous approaches in terms of perceptual quality, they often tend to produce less meaningful high-frequency noise in super-resolved images. We argue that this is because the most dominant difference between super-resolved images and real HR images is high-frequency information, where super-resolved images obtained by minimizing pixel-wise errors lack high-frequency details. The simplest way for a discriminator to distinguish super-resolved images from real HR images could be simply inspecting the presence of high-frequency components in a given image, and the simplest way for a generator to fool the discriminator would be to put arbitrary high-frequency noise into result images.

In this paper, we propose a novel GAN-based SISR method that can produce perceptually pleasing images (Fig. 1). To overcome the limitation of previous GAN-based SISR approaches and produce more realistic results, our method adopts two discriminators: an image discriminator and a feature discriminator, differently from previous approaches. The image discriminator takes an image in the pixel domain as input as done in previous approaches. On the other hand, the feature discriminator feeds an image into a VGG network and extracts an intermediate feature map. The feature discriminator then tries to distinguish super-resolved images from real HR images based on the extracted feature map. As the feature map encodes structural information, the feature discriminator distinguishes super-resolved images and real HR images based not simply on high-frequency components but on structural components. Eventually, our generator is trained to synthesize realistic structural features rather than arbitrary high-frequency noise.

To achieve high-quality SR, we also propose a novel generator network with long-range skip connections. Skip connections are first introduced in [18] to enable efficient propagation of information between neural network layers, and have been shown effective in training very deep networks. We further extend the idea of skip connections and introduce long-range skip connections to our generator network so that information in distant layers can be more effectively propagated. Our novel network architecture enables our generator to achieve state-of-the-art PSNRs when it is trained alone without discriminators, as well as perceptually pleasing results when trained with our discriminators.

Our contributions can be summarized as follows.

- We propose a new SISR framework that employs two different discriminators: an image discriminator working in the image domain, and a feature discriminator in the feature domain. Thanks to our feature discriminator, our generator network can produce perceptually realistic SR results. To our knowledge, this is the first attempt to apply GAN to the feature domain for SISR.
- We propose a novel generator with long-range skip connections for SISR. Our generator achieves the state-of-the-art performance in PSNR when compared to existing methods with the same amount of parameters.

2 Related Work

SISR has been intensively studied in computer vision and image processing. Early methods are based on simple interpolation such as bicubic and Lanczos interpolation [11]. While interpolation-based methods perform efficiently, they cannot restore fine textures and structures, producing oversmoothed images. To overcome this limitation, and to enhance edges, edge preserving interpolation [3, 29] and edge prior-based approaches [4, 8, 43] were proposed. However, because of the complexity of natural images, modeling global priors is not sufficient to deal with fine structures of various natural images.

To more effectively restore high-frequency details, a number of methods utilizing external information have been proposed. Freeman *et al.* [12] proposed to collect LR and HR patch pairs from a set of training images, and directly replace patches in an input LR image with collected HR patches. To further improve the quality, several other approaches along this line have been proposed such as neighborhood embedding [7,45,46,36], sparse coding [52,54,51,16], and local mapping function regression [15,50,38]. All these approaches collect pairs of LR and HR patches from a set of training images, and learn a mapping function between LR and HR patches in a low dimensional space. While these approaches show substantial quality improvement, their qualities are still limited due to their less capable mapping models for LR and HR images.

Recent advancement of deep learning has enabled to learn a more powerful mapping function from a LR image to a HR image. Dong *et al.* [10,9] trained a shallow convolutional neural network (CNN) with three layers using pairs of LR and HR image patches, and showed comparable performance to contemporary state-of-the-arts methods. To further improve the accuracy and also speed and memory efficiency, a number of CNN models have been proposed since then [24,31,47,41,6,44]. Specifically, Kim *et al.* [24] proposed very deep neural networks with one long skip-connection and showed that deeper networks can achieve better accuracy. Shi *et al.* [41] proposed a sub-pixel convolution layer that aggregates feature maps from the LR space to the HR space. Their sub-pixel convolution layer makes it possible to directly feed a LR image into a network, instead of a bicubic upsampled LR image, reducing memory usage and processing time. Thanks to the modeling power of CNNs, these methods have achieved high performance in terms of PSNR. However, they are still unable to restore high-frequency information because they rely on minimizing MSE losses, which results in blurry images as the minimization regresses to an average of solutions.

Recently, a few methods have been proposed to overcome the limitation of MSE losses and to produce perceptually more pleasing results. Johnson *et al.* [22] proposed a perceptual loss inspired by the content loss of [13]. A perceptual loss measures the difference between feature maps of two images extracted from image recognition networks such as VGG networks [42]. They showed that minimizing a perceptual loss results in low PSNRs but perceptually more pleasing results. However, their method is not able to restore high-frequency details completely lost in input images. GANs have also been recently employed for SISR [27,40] to synthesize perceptually pleasing high frequency details in super-resolved images. Ledig *et al.* [27] introduced an adversarial loss in addition to a perceptual loss. Sajjadi *et al.* [40] extends Ledig *et al.*'s work by introducing a texture matching loss inspired by a style loss in [13] in order to encourage super-resolved images to have the same texture styles as the ground truth HR images. While these methods are not able to restore high-frequency details completely lost in input images, they instead synthesize high-frequency details so that the results look perceptually pleasing. However, they tend to produce arbitrary high-frequency artifacts as discussed in Sec. 1. In addition, these GAN-based SR methods adopt a perceptual loss that minimizes MSE of VGG features. Similarly to MSE on pixels, simply minimizing MSE of VGG features would not be enough to fully represent the actual characteristics of feature maps. To remedy this, we adopt a feature discriminator to better regress to a real distribution of features and to generate perceptually more pleasing high frequency details.

3 Super-Resolution with Feature Discrimination

Our goal is to generate a HR image I^g from a given LR image I^l that looks as similar to the original HR image I^h as possible, and at the same time, perceptually pleasing. The LR image I^l of size $W' \times H' \times C$ can be obtained by applying various downsampling operations to the HR image I^h of size $W \times H \times C$ where $W = sW'$, $H = sH'$, and s is the scaling factor. In this paper, we assume only bicubic downsampling without loss of generality, i.e., we assume that I^l is obtained by downsampling with bicubic interpolation.

To recover I^h from I^l, we design a new deep CNN (DCNN)-based generator utilizing multiple long-range skip connections. The network generates a HR image I^g from I^l where I^g has the same dimensions as I^h. The network is first trained to reduce the pixel-wise difference between I^g and I^h. Pixel-wise loss well reproduces I^h in terms of PSNR, but generally results in a blurry and visually-unsatisfactory image I^g.

To improve the visual quality of I^g, we employ a perceptual loss and propose additional GAN-based loss functions. These losses enable the network to generate a visually more realistic image by approximating the distributions of natural HR images and their feature maps.

In the following subsections, we first describe the architecture of our generator. Then, we explain training loss functions in detail.

3.1 Architecture

We design a DCNN generator as illustrated in Fig. 2. The network consists of residual blocks [18] and multiple long-range skip connections. Specifically, the network takes I^l as input and first applies a 9×9 convolution layer to extract low-level features. The network then employs multiple residual blocks similarly to previous works [27, 40] to learn higher-level features with more nonlinearities and larger receptive fields. The residual block is successfully applied in various recent architectures [18, 32, 35] as it has been well proven that residual blocks enable efficient training process. Each block has a short-range skip connection as an identity mapping that preserves the signal from the previous layer and lets the network learn residuals only, while allowing back-propagation of gradients through the skip-connection path. Inspired by SRResNet [27], our residual block consists of multiple successive layers: 3×3 convolution, batchnorm, leaky ReLU [33], 3×3 convolution, and batchnorm layers. We use 16 residual blocks in our experiments to extract deep features. All the residual blocks are applied to the features of the LR spatial dimensions for efficient memory usage and fast inference. All the convolution layers in our generator network except the sub-pixel convolution layers have the same number of filters. In our experiments, we tried 64 and 128 filters for each convolution layer to analyze the performance of different network configurations.

We utilize additional long-range skip connections to aggregate features from different residual blocks. Specifically, we connect the output of each residual block to the end of the residual blocks with one 1×1 convolution layer. The

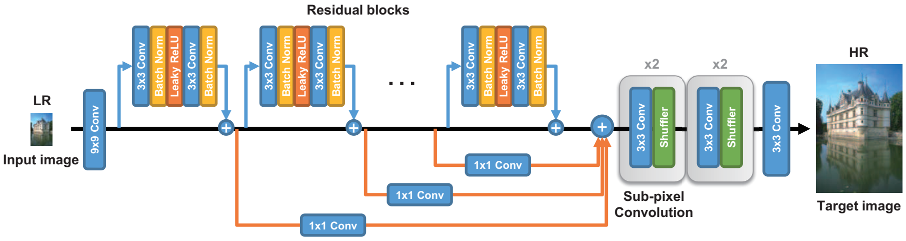


Fig. 2. Architecture of our generator network with short and long-range skip connections. We use 16 residual blocks for our experiments.

purpose of long-range skip connection is to further encourage back-propagation of gradients, and to give potentials to re-use intermediate features to improve the final feature. As the outputs of different residual blocks correspond to different levels of abstraction of image features, we apply 1×1 convolution to each long-range skip connection to adjust the outputs and balance them. The effect of this 1×1 convolution will be discussed in Sec. 4.3.

To upsample the feature map obtained by the residual blocks to the target resolution, we use sub-pixel convolution layers (also known as pixel shuffler layers) proposed in [41]. Specifically, a sub-pixel convolution layer consists of two sub-modules: one convolution layer with $s'^2 N_c$ filters where N_c is the number of input channels, and a shuffling layer that rearranges data from channels into different spatial locations. A sub-pixel convolution layer enlarges an input feature map by the scale factor s' in each spatial dimension. In our experiments, we consider only 4× upsampling, so we use two sub-pixel convolution layers with $s' = 2$ in a row. Finally, the upsampled feature map goes into a 3×3 convolution layer with three filters to obtain a 3-channel color image.

3.2 Pre-training of the Generator Network

We train our generator network through two steps: pre-training, and adversarial training. In the pre-training step, we train the network by minimizing a MSE loss defined as:

$$L_{MSE} = \frac{1}{WHC} \sum_{i}^{W} \sum_{j}^{H} \sum_{k}^{C} (I^h_{i,j,k} - I^g_{i,j,k})^2. \qquad (1)$$

The resulting network obtained from the pre-training step is already able to achieve high PSNRs. However, it cannot produce perceptually pleasing results with desirable high-frequency information.

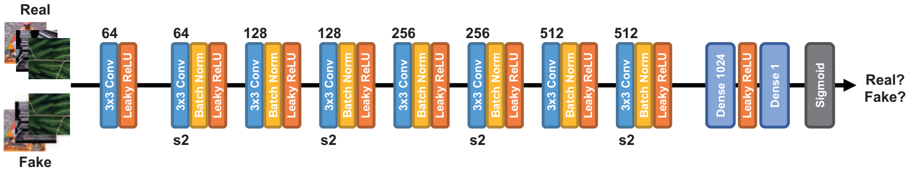


Fig. 3. Architecture of our discriminator network. The number above a convolution layer represents the number of filters, while s2 below represents the stride of 2.

3.3 Adversarial Training with a Feature Discriminator

To improve perceptual quality, we employ the GAN framework [14]. The GAN framework solves a minimax problem defined as:

$$\min_{g}\max_{d}\left(\mathbb{E}_{\boldsymbol{y}\sim p_{data}(\boldsymbol{y})}\left[\log\left(d\left(\boldsymbol{y}\right)\right)\right]+\mathbb{E}_{\boldsymbol{x}\sim p_{\boldsymbol{x}}(\boldsymbol{x})}\left[\log\left(1-d\left(g\left(\boldsymbol{x}\right)\right)\right)\right]\right), \tag{2}$$

where $g(\boldsymbol{x})$ is the output of a generator network for $\boldsymbol{x}$, and d is a discriminator network. $\boldsymbol{y}$ is a sample from a real data distribution and $\boldsymbol{x}$ is random noise.

While the conventional GAN framework consists of a pair of a single generator and a single discriminator, we use two discriminators: an image discriminator d^i and a feature discriminator d^f. Our image discriminator d^i discriminates real HR images and fake SR images by inspecting their pixel values. On the other hand, our feature discriminator d^f discriminates real HR images and fake SR images by inspecting their feature maps so that the generator can be trained to synthesize more meaningful high-frequency details.

To train our pre-trained generator network with discriminators, we minimize a loss function defined as:

$$L_g = L_p + \lambda\left(L_a^i + L_a^f\right), \tag{3}$$

where L_p is a perceptual similarity loss that enforces SR results to look similar to the ground truth HR images in the training set. L_a^i is an image GAN loss for the generator to synthesize high-frequency details in the pixel domain. L_a^f is a feature GAN loss for the generator to synthesize structural details in the feature domain. λ is a weight for the GAN loss terms. While L_g looks similar to the loss functions of previous methods, it has an additional feature GAN loss term L_a^f that makes a significant difference in terms of perceptual quality as shown in our experiments. To train discriminators d^i and d^f, we minimize loss functions L_d^i and L_d^f, each of which corresponds to L_a^i and L_a^f, respectively. The generator and discriminators are trained by alternatingly minimizing L_g, L_d^i and L_d^f. In the following, we will describe each of the loss terms in more detail.

Perceptual Similarity Loss $\boldsymbol{L_p}$ The perceptual similarity loss measures the difference between two images in the feature domain instead of the pixel domain so that minimizing it leads to perceptually consistent results [22]. The perceptual similarity loss L_p between I^h and I^g is defined in the following manner. First, I^h and I^g are fed into a pre-trained recognition network such as a VGG network. Then, the feature maps of the two images at the m-th layer are extracted. The MSE difference between the extracted feature maps is defined as the perceptual similarity loss. Mathematically, L_p is defined as:

$$L_p = \frac{1}{W_m H_m C_m} \sum_{i}^{W_m} \sum_{j}^{H_m} \sum_{k}^{C_m} \left(\phi^m_{i,j,k}(I^h) - \phi^m_{i,j,k}(I^g) \right)^2 , \tag{4}$$

where $W_m, H_m,$ and C_m denote the dimensions of the m-th feature map ϕ^m. In our experiments, we use VGG-19 [42] for the recognition network. Here ϕ^m represents the output of the ReLU layer after the convolution before the m-th pooling.

Image GAN Losses $\boldsymbol{L_a^i}$ and $\boldsymbol{L_d^i}$ The image GAN loss term L_a^i for the generator and the loss function L_d^i for the image discriminator are defined as:

$$L_a^i = -\log\left(d^i\left(I^g\right)\right), \qquad \text{and} \tag{5}$$

$$L_d^i = -\log\left(d^i\left(I^h\right)\right) - \log\left(1 - d^i\left(I^g\right)\right), \tag{6}$$

where $d^i(I)$ is the output of the image discriminator d^i, i.e., the probability that the image I is an image sampled from the distribution of natural HR images. Note that we minimize $-\log(d^i(I^g))$ instead of $\log(1 - d^i(I^g))$ for stable optimization [14]. For the image discriminator d^i, we use the same discriminator network used in [27] following the guidelines proposed by [37] (Fig. 3).

Feature GAN Losses $\boldsymbol{L_a^f}$ and $\boldsymbol{L_d^f}$ The feature GAN loss term L_a^f for the generator and the loss function L_d^f for the feature discriminator are defined as:

$$L_a^f = -\log\left(d^f\left(\phi^m\left(I^g\right)\right)\right), \qquad \text{and} \tag{7}$$

$$L_d^f = -\log\left(d^f\left(\phi^m\left(I^h\right)\right)\right) - \log\left(1 - d^f\left(\phi^m\left(I^g\right)\right)\right), \tag{8}$$

where $d^f(\phi^m)$ is the output of the feature discriminator d^f, i.e., the probability that the feature map ϕ^m is sampled from the distribution of the natural HR image feature maps. As features correspond to abstracted image structures, we can encourage the generator to produce realistic structural high-frequency rather than noisy artifacts. Both the perceptual similarity loss and the feature GAN losses are based on feature maps. However, in contrast to the perceptual similarity loss that promotes perceptual consistency between I^g and I^h, the feature GAN losses L_a^f and L_d^f enable synthesis of perceptually valid image details. We use the network architecture in Fig. 3 for the feature discriminator d^f in our experiments. We also tried variations of the network architecture, but observed no significant performance difference between them, while all the variations showed similar tendency of improvement. We refer the reader to our supplementary material for the results with other variations.

4 Experiments

In this section, we first present details about our dataset and training process. We then analyze the performance of a pre-trained generator network, and a fully trained version with the feature discriminator.

4.1 Dataset

We used ImageNet [39] dataset for pre-training the generator as done in [27]. The dataset contains millions of images in 1000 categories. We randomly sampled about 120 thousands of images that have width and height larger than 400 pixels and then we took a center-cropped version of the sampled images for pre-training. For evaluation, we use three widely used datasets: Set5 [5], Set14 [53], and 100 test images of BSD300 [34].

To train our final GAN-based model, we used DIV2K dataset [2] which consists of 800 HR training images and 100 HR validation images. In our experiments, we observed that training our GAN-based model with DIV2K dataset is faster and more stable than with ImageNet. We conjecture that this is partly because DIV2K images are in lossless PNG format while ImageNet images are in lossy JPEG format. To expand the volume of training data, we applied data augmentation to DIV2K images. Specifically, we applied random flipping, rotation, and cropping to the images to make target HR images. We additionally sampled a small number of training images and included their downscaled versions by 1/2 and 1/4 for data augmentation in order to train the network to be able to deal with contents of different scales.

4.2 Training Details

Here we explain training details in our experiments. We obtained the target HR images by cropping the HR images to 296×296 sub images. We downsampled the images using bicubic interpolation[1] to obtain the 74×74 low-resolution input training images. We normalized the intensity ranges of I^h and I^l to $[-1, 1]$. We set the weight λ in Eq. (3) as 10^{-3}. Regarding ϕ^m in Eqs. (4), (7) and (8), we used Conv5 layer in VGG-19 in our experiments as we found that Conv5 generally produces better results than other layers. To balance different loss terms, we scaled the feature map ϕ^m with a scaling factor 1/12.75 before we computed loss terms.

For both pre-training and adversarial training, we used Adam optimizer [26] with the momentum parameter $\beta_1 = 0.9$. For pre-training, we performed about 280 thousand iterations, which are roughly 20 epochs for our randomly sampled ImageNet dataset. We set the initial learning rate for pre-training as 10^{-4} and decreased it by 1/10 when the training loss stopped decreasing. After the learning rate reached at 10^{-6}, we used the value without further decreasing. We performed adversarial training for about five epochs, which are roughly 100,000

[1] We used MATLAB function 'imresize' for bicubic interpolation with anti-aliasing.

Table 1. Quantitative comparison of SISR methods for ×4 upscaling; A+ [46], SRCNN [10], VDSR [24], Enhance [40], SRDense [47], SRRes [27]. Our network (SRFeat$_{\mathrm{M}}$) obtains the best accuracy in terms of PSNR and SSIM. With a similar number of parameters, our network with 64 feature channels (SRFeat$_{\mathrm{M}}$-64) shows better accuracy than SRResNet.

Set5	Bicubic	A+	SRCNN	VDSR	Enhance	SRDense	SRRes	**SRFeat$_{\mathrm{M}}$-64**	**SRFeat$_{\mathrm{M}}$**
PSNR	28.42	30.28	30.48	31.35	31.74	32.02	32.05	32.14	**32.27**
SSIM	0.8104	0.8603	0.8628	0.8838	0.8869	0.8934	0.8910	0.8918	**0.8938**
Set14	Bicubic	A+	SRCNN	VDSR	Enhance	SRDense	SRRes	**SRFeat$_{\mathrm{M}}$-64**	**SRFeat$_{\mathrm{M}}$**
PSNR	26.00	27.32	27.49	28.01	28.42	28.50	28.53	28.61	**28.71**
SSIM	0.7027	0.7491	0.7503	0.7674	0.7774	0.7782	0.7804	0.7816	**0.7835**
BSD100	Bicubic	A+	SRCNN	VDSR	Enhance	SRDense	SRRes	**SRFeat$_{\mathrm{M}}$-64**	**SRFeat$_{\mathrm{M}}$**
PSNR	25.96	26.82	26.90	27.29	27.50	27.53	27.58	27.59	**27.64**
SSIM	0.6675	0.7087	0.7101	0.7251	0.7326	0.7337	0.7354	0.7357	**0.7378**

iterations. We used 10^{-4} as the learning rate for the first two epochs, 10^{-5} for the next two epochs, and 10^{-6} for the final one epoch of adversarial training. We fixed the parameters in batch-normalization layers during the test phase. All the models were trained on an NVIDIA Titan XP with 12GB memory.

4.3 Evaluation of the Pre-trained Generator

As our pre-trained network is trained using only a MSE loss, it is supposed to maximize PSNRs. To evaluate the performance of the pre-trained network, we measure PSNRs and SSIMs [48] on Y channel and compare them with those of other state-of-the-arts methods. For fair comparison, we excluded four pixels from the image boundaries as most existing SISR methods are not able to restore image boundaries properly. For our network, we tested two different configurations, one with 128 channels, and the other with 64 channels. We denote them as SRFeat$_{\mathrm{M}}$ and SRFeat$_{\mathrm{M}}$-64, respectively. SRFeat$_{\mathrm{M}}$-64 has a similar number of parameters to SRResNet [27]. Specifically, the difference between the model sizes of SRFeat$_{\mathrm{M}}$-64 and SRResNet is less than 0.06MB. Table 1 shows that SRFeat$_{\mathrm{M}}$ achieves the state-of-the-art accuracy and outperforms all the other methods. SRFeat$_{\mathrm{M}}$-64 also achieves higher PSNRs and SSIMs than SRResNet [27], where they have similar numbers of parameters.

In Table 2, we compare variations of our architecture to see the effect of each component. We first verify the necessity of 1×1 convolution in the long-range skip connection. Without 1×1 convolution (w/o Conv), features from different residual blocks equally contribute to the final feature regardless that they are high-level or low-level features. Table 2 shows that long-range skip connections without 1×1 convolution result in worse quality than SRFeat$_{\mathrm{M}}$-64. The table also shows that the network with long-range skip connections with 1×1 convolution achieves higher quality than the network without long-range skip connections (w/o Skip), which verifies the effectiveness of long-range skip connections with 1×1 convolution.

Table 2. Comparison between variations of our generator network.

		Set5	Set14	BSD100
SRFeat_M	PSNR	**32.27**	**28.71**	**27.64**
	SSIM	**0.8938**	**0.7835**	**0.7378**
w/o Conv	PSNR	32.05	28.59	27.56
	SSIM	0.8912	0.7809	0.7353
w/o Skip	PSNR	32.22	28.71	27.63
	SSIM	0.8933	0.7833	0.7373
SRFeat_M-64	PSNR	32.14	28.61	27.59
	SSIM	0.8918	0.7816	0.7357

4.4 Evaluation of the Fully Trained Generator

We evaluate the performance of our GAN-based final generator. Existing quantitative assessment measures such as PSNR and SSIM are not appropriate to measure the perceptual quality of images. To provide a measure reasonably correlated with human perception, Sajjadi *et al.* [40] used object recognition performance. They first downsample original HR images and perform SISR on those images. Then, they apply a state-of-the-art object recognition model to the SR results as well as the original HR images. They assume that the gap between the object recognition accuracies from those results implies degradation of perceptual qualities. We also adopt the approach to validate the perceptual quality of our method.

We used the official Caffe model of ResNet-50 [18] for the recognition model, which obtained the state-of-the-art classification accuracy. For evaluation, we used the first 1000 images from the validation set of ILSVRC2016 CLS-LOC dataset as done in [40]. To compute the baseline accuracy, we resized the images to have 256 pixels along the shorter side and cropped the center of 224×224 pixels as done in [18]. Then, we made four different degraded versions of the dataset by downsampling the images to 56×56 and applying four different versions of our generator network: SRFeat_M trained with MSE, SRFeat_I trained with the perceptual loss and the image GAN loss but without the feature GAN loss, and SRFeat_IF-64 and SRFeat_IF trained with all loss terms. All the networks use 128 filters in their convolution layers except SRFeat_IF-64 that uses 64 filters. We also report the error rates of [40] taken from their paper although the baseline error rates reported in the paper using the same ResNet-50 network is slightly different from ours (*e.g.* Top-5 error rate: 7.1 % in ours and 7.2 % in [40]). We suspect that the gap comes from the differences in deep learning platforms such as Caffe [21] and Tensorflow [1].

The results are shown in Table 3. Obviously, our SRFeat_M without GAN shows much worse accuracy than the baseline obtained using the original images as it generates blurry images without high-frequency details. However, our SRFeat_I with the image GAN loss considerably improves the accuracy by restoring textures lost in downsampling. With our feature GAN loss (SRFeat_IF), the gap between the baseline and ours reduces up to 3.9 % in the case of Top-5

Table 3. Performances of classification tests using images from the validation dataset of ILSVRC 2016. The baseline error rate was calculated from the inference results of ResNet-50 for the original 224 × 224 cropped images. SRFeat$_{\mathrm{I}}$ and SRFeat$_{\mathrm{IF}}$ denote our networks trained using GAN-based perceptual losses without and with the feature GAN loss, respectively.

ResNet-50	Bicubic	SRFeat$_{\mathrm{M}}$	Enhance [40]	SRFeat$_{\mathrm{I}}$	SRFeat$_{\mathrm{IF}}$-64	SRFeat$_{\mathrm{IF}}$	Baseline
Top-1 error (%)	47.9	41.4	39.9	31.1	33.0	**30.9**	25.4
Top-5 error (%)	23.0	20.1	17.1	11.9	11.8	**11.0**	7.1

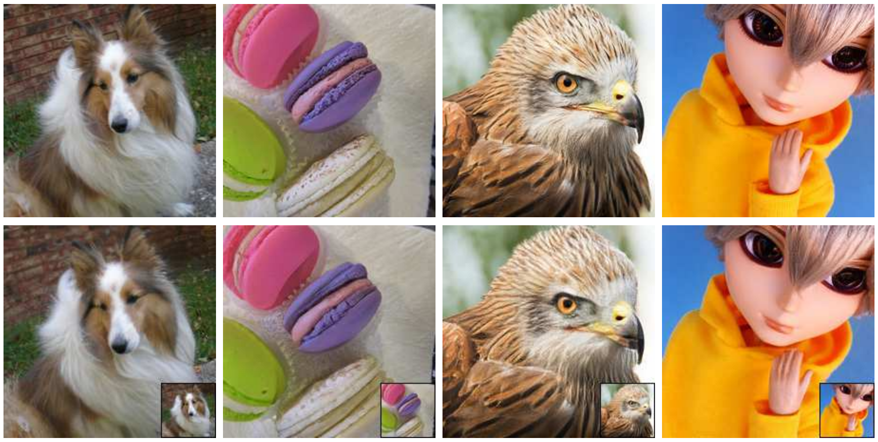
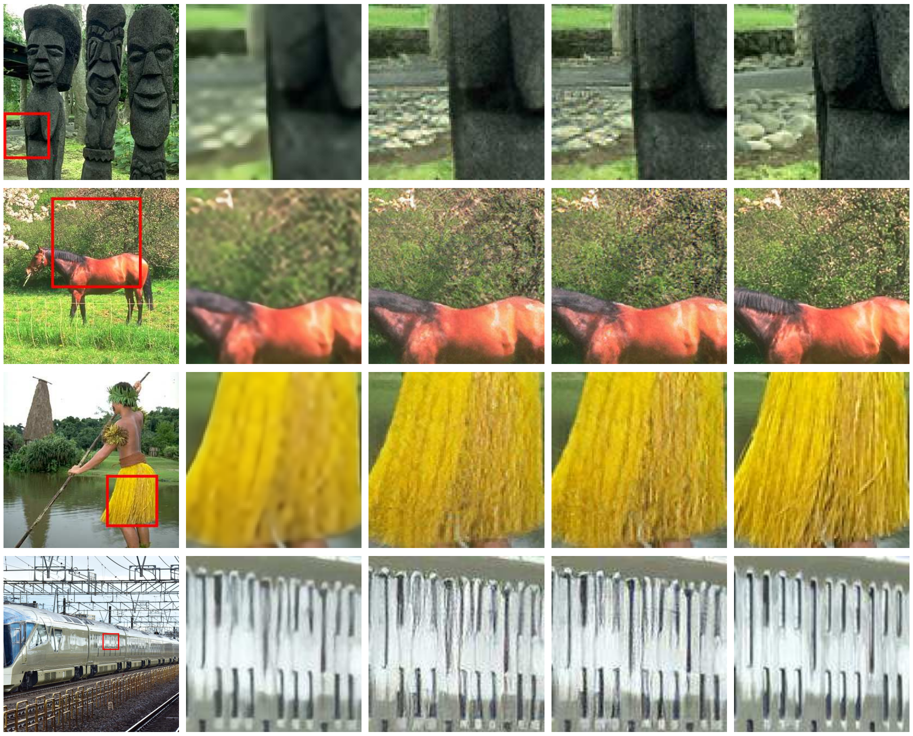


Fig. 4. Samples of original input and SR images used in the classification test. Top row: original images (224 × 224). Bottom row: SR images (224 × 224) and the LR images (56 × 56) at the lower right corners.

error. Fig. 4 shows some samples drawn from the validation dataset. From the samples, we can see that the accuracy is reasonable as the perceptual quality difference between the original images and our results is not significant. The gap between SRFeat$_{\mathrm{I}}$ and SRFeat$_{\mathrm{IF}}$ in Top-5 error (0.9) is larger than the gap between SRFeat$_{\mathrm{IF}}$-64 and SRFeat$_{\mathrm{IF}}$ in Top-5 error (0.8), which implies the effectiveness of our feature GAN loss. There is also a large gap between EnhanceNet [40] and all our networks except SRFeat$_{\mathrm{M}}$, which clearly shows the effectiveness of our method.

We also qualitatively exhibit the improvement in perceptual quality obtained by employing the feature GAN loss. As shown in Fig. 5, our feature GAN loss suppresses noisy high frequencies, while generating perceptually plausible structured textures. Fig. 6 shows a qualitative comparison of GAN-based SR methods. EnhanceNet results have high-frequency artifacts around edges, and SRGAN results have blurry structural textures. On the other hand, our results have naturally synthesized sharp details without blurriness or high-frequency artifacts thanks to our feature GAN loss. We refer the readers to the supplementary material for more results including a user study.



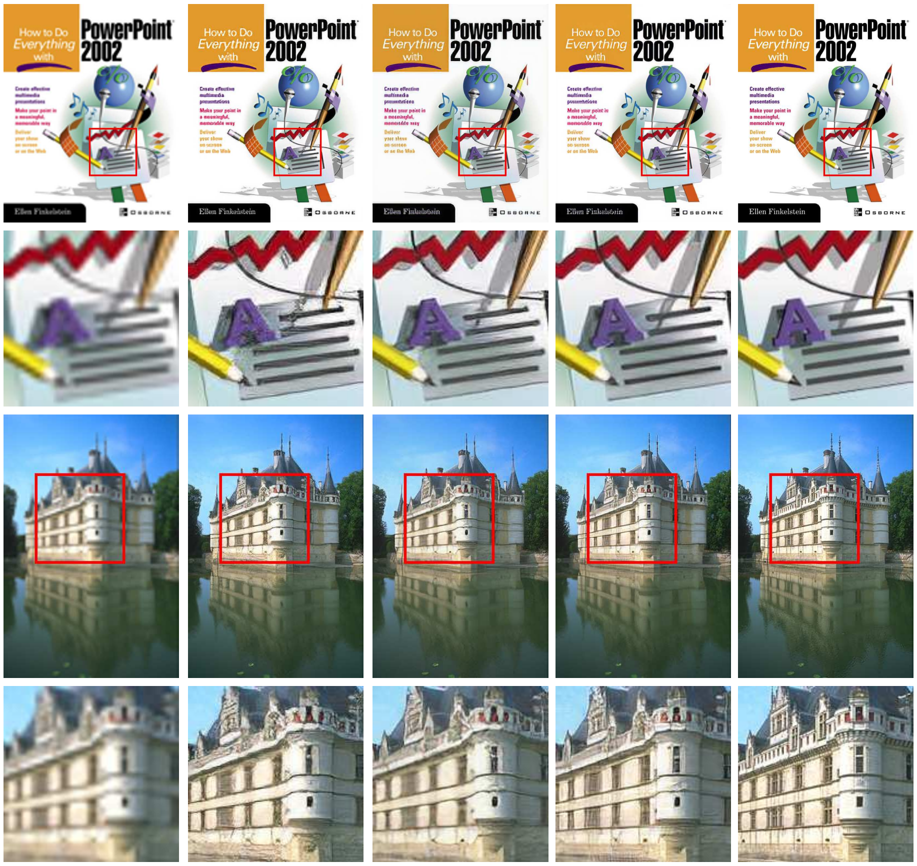
original images (a) MSE (b) SRFeat$_\mathrm{I}$ (c) SRFeat$_\mathrm{IF}$ (d) GT

Fig. 5. Qualitative comparison between our models without the feature GAN loss (SRFeat$_\mathrm{I}$) and with the feature GAN loss (SRFeat$_\mathrm{IF}$). In all examples, SRFeat$_\mathrm{IF}$ generates more realistic textures than SRFeat$_\mathrm{I}$ while suppressing arbitrary high-frequency artifacts.

5 Discussion and Conclusion

We proposed a novel SISR method that can produce perceptually pleasing images by employing two discriminators: an image discriminator and a feature discriminator. Especially, our feature discriminator encourages the generator to make more structural high-frequency details rather than noisy artifacts. We also proposed a novel generator network architecture employing long-range skip connections for more effective propagation of information between distant layers. Experiments showed that our results achieve the state-of-the-art performance quantitatively and qualitatively.

For the feature GAN loss and perceptual similarity loss, our network uses features of only one fixed layer. However, we found that the optimal layer for the feature GAN loss and perceptual similarity loss depends on image contents. Therefore, we may further improve perceptual quality if we can adaptively choose a layer according to image contents. We leave this content-dependent SR as our future work. Applying the GAN framework to feature maps may also be



(a) Bicubic (b) Enhance [40] (c) SRGAN [27] (d) SRFeat$_{\mathrm{IF}}$ (e) GT

Fig. 6. Qualitative comparison of GAN-based SR methods with our results at scaling factor 4. Result images of the other methods are taken from their websites.

beneficial to other problems besides SR. Exploring other applications can be another interesting future work.

Acknowledgements We appreciate the constructive comments from the reviewers. This work was supported by the Ministry of Science and ICT, Korea, through IITP grant (IITP-2015-0-00174), Giga Korea grant (GK18P0300), and NRF grant (NRF-2017M3C4A7066317). It was also supported by the DGIST Start-up Fund Program (2018010071).

References

1. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., Kudlur, M., Levenberg, J., Monga, R., Moore,

S., Murray, D.G., Steiner, B., Tucker, P., Vasudevan, V., Warden, P., Wicke, M., Yu, Y., Zheng, X.: Tensorflow: A system for large-scale machine learning. In: Proc. OSDI (2016)

2. Agustsson, E., Timofte, R.: Ntire 2017 challenge on single image super-resolution: Dataset and study. In: CVPR Workshops (2017)
3. Allebach, J., Wong, P.W.: Edge-directed interpolation. In: Proc. ICIP (1996)
4. Baker, S., Kanade, T.: Limits on super-resolution and how to break them. IEEE Transactions on Pattern Analysis and Machine Intelligence **24**(9), 1167–1183 (2002)
5. Bevilacqua, M., Roumy, A., Guillemot, C., Alberi-Morel, M.L.: Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In: Proc. BMVC (2012)
6. Bruna, J., Sprechmann, P., LeCun, Y.: Super-resolution with deep convolutional sufficient statistics. Proc. ICLR (2016)
7. Chang, H., Yeung, D.Y., Xiong, Y.: Super-resolution through neighbor embedding. In: Proc. CVPR (2004)
8. Dai, S., Han, M., Xu, W., Wu, Y., Gong, Y., Katsaggelos, A.K.: Softcuts: A soft edge smoothness prior for color image super-resolution. IEEE Transactions on Image Processing **18**(5), 969–981 (2009)
9. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. IEEE Transactions on Pattern Analysis and Machine Intelligence **38**(2), 295–307 (2016)
10. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: Proc. ECCV (2014)
11. Duchon, C.E.: Lanczos filtering in one and two dimensions. Journal of Applied Meteorology **18**(8), 1016–1022 (1979)
12. Freeman, W.T., Pasztor, E.C., Carmichael, O.T.: Learning low-level vision. International Journal of Computer Vision **40**(1), 25–47 (2000)
13. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Proc. CVPR (2016)
14. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Proc. NIPS (2014)
15. Gu, S., Sang, N., Ma, F.: Fast image super resolution via local regression. In: Proc. ICPR (2012)
16. Gu, S., Zuo, W., Xie, Q., Meng, D., Feng, X., Zhang, L.: Convolutional sparse coding for image super-resolution. In: Proc. ICCV (2015)
17. Gupta, P., Srivastava, P., Bhardwaj, S., Bhateja, V.: A modified psnr metric based on hvs for quality assessment of color images. In: Proc. ICCIA (2011)
18. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proc. CVPR (2016)
19. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Globally and locally consistent image completion. ACM Transactions on Graphics **36**(4), 107 (2017)
20. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: Proc. CVPR (2017)
21. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: Convolutional architecture for fast feature embedding. In: Proc. ACM MM (2014)
22. Johnson, J., Alahi, A., fei Li, F.: Perceptual losses for real-time style transfer and super-resolution. In: Proc. ECCV (2016)
23. Karras, T., Aila, T., Laine, S., Lehtinen, J.: Progressive growing of gans for improved quality, stability, and variation. In: Proc. ICLR (2018)

24. Kim, J., Lee, J.K., Lee, K.M.: Accurate image super-resolution using very deep convolutional networks. In: Proc. CVPR (2016)
25. Kim, T., Cha, M., Kim, H., Lee, J., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. In: Proc. ICML (2017)
26. Kingma, D., Ba, J.: Adam: A method for stochastic optimization. In: Proc. ICLR (2015)
27. Ledig, C., Theis, L., Huszar, F., Caballero, J., Aitken, A.P., Tejani, A., Totz, J., Wang, Z., Shi, W.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proc. CVPR (2017)
28. Li, J., Liang, X., Wei, Y., Xu, T., Feng, J., Yan, S.: Perceptual generative adversarial networks for small object detection. In: Proc. CVPR (2017)
29. Li, X., Orchard, M.T.: New edge-directed interpolation. IEEE Transactions on Image Processing **10**(10), 1521–1527 (2001)
30. Li, Y., Liu, S., Yang, J., Yang, M.H.: Generative face completion. In: Proc. CVPR (2017)
31. Lim, B., Son, S., Kim, H., Nah, S., Lee, K.M.: Enhanced deep residual networks for single image super-resolution. In: CVPR Workshops (2017)
32. Lin, G., Milan, A., Shen, C., Reid, I.: Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In: Proc. CVPR (2017)
33. Maas, A.L., Hannun, A.Y., Ng, A.Y.: Rectifier nonlinearities improve neural network acoustic models. In: Proc. ICML (2013)
34. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proc. ICCV (2001)
35. Park, S.J., Hong, K.S., Lee, S.: Rdfnet: Rgb-d multi-level residual feature fusion for indoor semantic segmentation. In: Proc. ICCV (2017)
36. Perez-Pellitero, E., Salvador, J., Ruiz-Hidalgo, J., Rosenhahn, B.: Psyco: Manifold span reduction for super resolution. In: Proc. CVPR (2016)
37. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. In: Proc. ICLR (2016)
38. Romano, Y., Isidoro, J., Milanfar, P.: Raisr: Rapid and accurate image super resolution. IEEE Transactions on Computational Imaging **3**, 110–125 (2017)
39. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: Imagenet large scale visual recognition challenge. International Journal of Computer Vision **115**(3), 211–252 (2015)
40. Sajjadi, M., Schölkopf, B., Hirsch, M.: Enhancenet: Single image super-resolution through automated texture synthesis. In: Proc. ICCV (2017)
41. Shi, W., Caballero, J., Huszar, F., Totz, J., Aitken, A.P., Bishop, R., Rueckert, D., Wang, Z.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proc. CVPR (2016)
42. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: Proc. ICLR (2015)
43. Sun, J., Sun, J., Xu, Z., Shum, H.Y.: Gradient profile prior and its applications in image super-resolution and enhancement. IEEE Transactions on Image Processing **20**(6), 1529–1542 (2011)
44. Tai, Y., Yang, J., Liu, X.: Image super-resolution via deep recursive residual network. In: Proc. CVPR (2017)
45. Timofte, R., De, V., Gool, L.V.: Anchored neighborhood regression for fast example-based super-resolution. In: Proc. ICCV (2013)

46. Timofte, R., De, V., Gool, L.V.: A+: Adjusted anchored neighborhood regression for fast super-resolution. In: Proc. ACCV (2014)
47. Tong, T., Li, G., Liu, X., Gao, Q.: Image super-resolution using dense skip connections. In: Proc. ICCV (2017)
48. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: From error visibility to structural similarity. IEEE Transactions on Image Processing **13**(4), 600–612 (2004)
49. Wang, Z., Simoncelli, E.P., Bovik, A.C.: Multiscale structural similarity for image quality assessment. In: Proc. ACSSC (2003)
50. Yang, C.Y., Yang, M.H.: Fast direct super-resolution by simple functions. In: Proc. ICCV (2013)
51. Yang, J., Wang, Z., Lin, Z., Cohen, S., Huang, T.: Coupled dictionary training for image super-resolution. IEEE Transactions on Image Processing **21**(8) (2012)
52. Yang, J., Wright, J., Huang, T.S., Ma, Y.: Image super-resolution via sparse representation. IEEE Transactions on Image Processing **19**(11), 2861–2873 (2010)
53. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: Proc. Curves and Surfaces (2012)
54. Zhang, K., Gao, X., Tao, D., Li, X.: Multi-scale dictionary for single image super-resolution. In: Proc. CVPR (2012)
55. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: Proc. ICCV (2017)