SIR-SRGAN: Super-Resolution Generative Adversarial Network with Self-Interpolation Ranker

Jun-Hong Huang[1]
xx1217522411@outlook.com
Hai-Kun Wang[1]
wanghaikun@gmail.com
Yong Yu[2]
pzhyuyong@126.com
Zhi-Wu Liao[1]
liaozhiwu@163.com

[1]The college of computer science
Sichuan Normal University
Chengdu, China
[2]The college of mathematics and computers
Panzhihua University
Panzhihua, China

Abstract

Super-Resolution Generative Adversarial Networks (SRGAN) and follow-up perceptual single image super-resolution(SISR) method has shown us impressive texture generation capability. However, these models do not fully exploit the difference between the reconstructed image and the original image. In this paper, we propose a Self-Interpolation Ranker(SI-Ranker) to take advantage of the difference between the reconstructed image and the original image. SI-Ranker sorts the interpolated image of the reconstruction image and the original image and guides the image reconstruction during training. This method allows the generator to focus on the difference between the reconstruction image and the original image to improve the quality of the reconstructed image while obtaining a reconstruction image closer to the original image. In addition, we propose Patch Distance Loss (PDL) to reduce the artifacts in the reconstructed image. PDL reduces artifacts by calculating the cosine similarity between the reconstructed image and HR in all patches. Experiments show that SIR-SRGAN improves consistency with the original at both pixel and feature levels, allowing it to be compared to the state-of-the-art. Our code at http://github.com/huang-junhong/SIRSRGAN.

1. Introduction

The purpose of the single image super-resolution (SISR) is to restore the input low-resolution (LR) image to its corresponding high-resolution image (HR). SISR is an ill-posed problem because different HRs may get the same LR by downsampling. Many SISR models[2][3][4][5][6] use mean-absolute-error(MAE) or mean-square-error(MSE) to get higher PSNR. However, people discovered that these methods' optimization of PSNR was too smooth to match the human visual perception when the super-resolution factor 4(SRF4) or bigger. [7][8] was the first to introduce the Generation Adversarial Network (GAN)[9] into SISR, which improved the realism of the reconstructed image. However, with GAN, the reconstruction images produced many artifacts. Subsequently, ESRGAN[11], NatSR[12], SFT-SRGAN[44], Rank-SRGAN[13], SPSR[14], made progress in suppressing artifacts and improving the quality of reconstruction.

One of the problems with the perceptual SISR method is that using GAN enhances the realism of textures while producing many artifacts and noise, reducing the consistency of the reconstructed image with the original image. In this paper, we propose Self-Interpolation

Ranker(SI-Ranker) and patch distance loss (PDL) to improve the consistency of the reconstructed image with the original image. Specifically, SI-Ranker discovers the intrinsic differences by sorting the interpolation image between reconstructing images and HR. The interpolation factor determines the quality of interpolated images without using IQA; this also eliminates the need for an SI-Ranker to prepare external data and pay more attention to the intrinsic characteristics of the reconstructed image. PDL is a loss function used to constrain high frequencies and reduces the artifacts of the reconstructed image by cutting the reconstructed image and HR into patches and calculating the cosine similarity between SR patches and all HR patches.

In summary, our contributions include:

1. We propose a framework SIR-SRGAN for supervised SISR, which uses SI-Ranker to dynamically explore the intrinsic differences between the reconstructed image and the original image to improve the reconstructed image quality.
2. We propose PDL, which reduces the noise and artifacts of reconstructing images by cutting the reconstructed image and HR into patches and calculating the cosine similarity between SR patches and all HR patches.

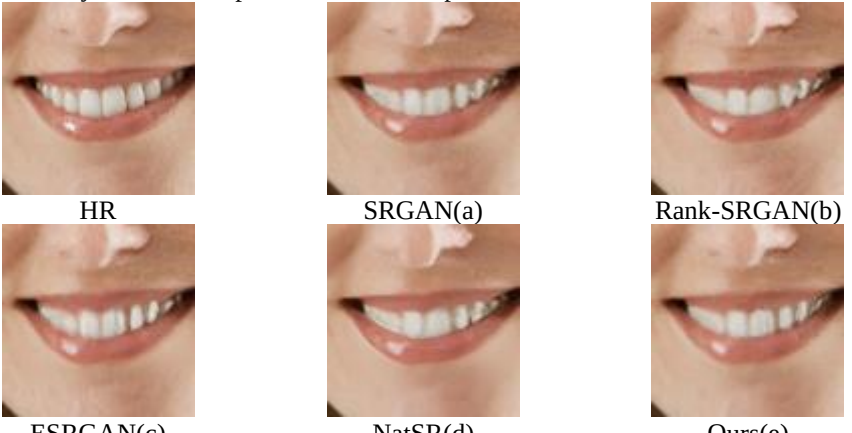


Figure 1. SR results of different methods. That model is the perceptual SISR method. SIR-SRGAN uses SRRes-Net as its generator, which same as (a), (b).

2. Related Works

Let us start with SISR methods, which can broadly divide into PSNR-Oriented and Perceptual-Oriented.

PSNR-Oriented SISR: Dong et al. first proposed [2] using CNN in SISR. Soon many CNN-based SISR models were proposed. These methods[3],[4],[5],[6][15][16][17] use MSE or MAE as their loss function to get higher PSNR, and the main differences are the structure and depth of the network. These methods make a significant contribution to improve PSNR. However, these PSNR-Oriented models' results are too smooth in larger SRF.

Perceptual-Oriented SISR: [7],[8] introduce GAN into SISR for the first time. It makes the reconstructed image have a rich texture and improves realism. The perceptual-Oriented method is often tied to Perceptual Loss[18] because GAN eliminates the board artifacts that Perceptual loss brings while enhancing its effects. ESRGAN[11] replaces the generator with RRDN, while the discriminator uses Ra-GAN to improve reconstruction quality. NatSR[12] proposes to add a natural manifold discriminator based on ESRGAN to improve the quality

of reconstructed images. SPSR[14] adds gradient branches and gradient loss to the generator to protect the structure of the reconstruction image. SFT-SRGAN[44] integrates semantic information into the generator to improve the texture of the reconstructed image. The closest method to our work is the recently proposed Rank-SRGAN[13]. Rank-SRGAN uses the reconstructed images from other SISR models to train rankers to learn image quality ranking. The selected IQA determines the quality and ranking of the image. Rank-SRGAN will only improve the parts that IQA is interested in. No IQA can comprehensively evaluate the image quality, and such improvement is not comprehensive. Different IQA focuses on different image information, making it difficult for these IQAs to combine[52].

Learn to rank: Learn to rank was first proposed by [20], and its original purpose was to control the strength of a property in the generated image because a property is not only true or false. [21] then first used CNN to learn rank. [22] merges rank with GAN to produce more realistic images. For the first time, [23] uses Ranker for IQA that is called Rank-IQA. Recently, Rank-SRGAN[13] combined different SISR models to simulate the selected IQA and guide image reconstruction.

Unlike Rank-SRGAN, which wants to optimize IQA, we pay attention to the internal information of reconstructed images. Meanwhile, simply eliminating noise and artifacts will reduce the visual quality of the reconstructed image. The image interpolation method can apply to the pixel and image features without conflict. To our knowledge, no one has tried to sort interpolated images and their features before.

3. Method

3.1 Overview

Figure 2 shows the overall structure of SIR-SRGAN. SIR-SRGAN uses the same generator(G) and discriminator(D) with SRGAN. So, the G wants to reconstruct an image that cannot be judged by D, while D hopes to determine whether the input image is an HR or a reconstructed image. Based on SRGAN, we added a pixel ranker and feature ranker. The input of Pixel Ranker is images. (The structure of the two rankers is in the supplementary file). The input of the feature ranker is the high-level feature extracted by the VGG19[29] model. PDL is an additional loss function to reduce artifacts and improve texture.

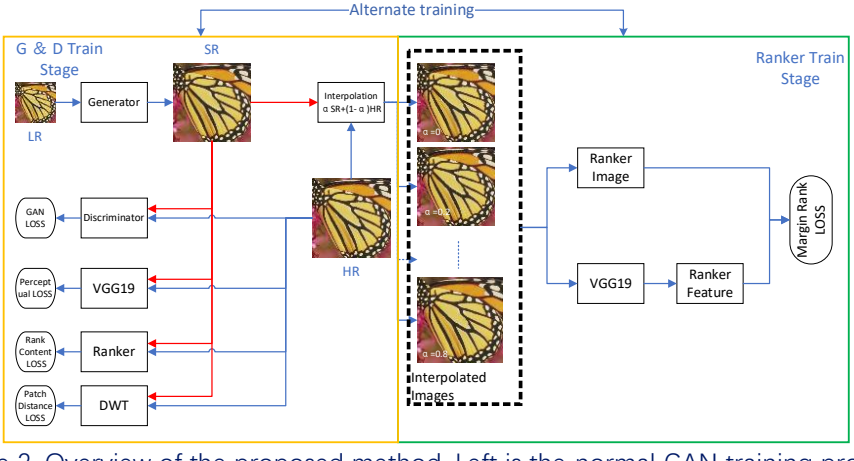


Figure 2. Overview of the proposed method. Left is the normal GAN training process, and right is how SI-Ranker works.

3.2 SI-Ranker

The input during SI-Ranker training is interpolation images between HRs and reconstructed images. It represents as:

$$\begin{cases} Itp\_Img_{\alpha} = SR * \alpha + (1-\alpha) * HR \\ \alpha \in [0,1) \end{cases} \tag{1}$$

Where α is the interpolation factor, SR is the reconstructed image; HR corresponds to a high-resolution image, Itp_Img_{α} is the interpolated image. Different interpolation factors can directly determine the quality of interpolated images without any IQA. Further, controlling the number of interpolated images, the difference between the interpolated images and the original images can be smoother. The lower the score for the input image, the better. That is

$$\begin{cases} S_{\alpha} = R(\theta, Itp\_Img_{\alpha}) \\ S_{\alpha} < S_{\beta} \; if \; \alpha < \beta \end{cases} \tag{2}$$

θ is the parameter that needs to optimize in the network R. S_{α} is the score of Itp_Img_{α} in R, α, β are interpolation factors. The smaller interpolation factors, the better quality. To optimize ranker, we used Margin Ranking Loss as Ranker's optimization function like [13,23]. It represents as:

$$\begin{cases} L(S_{\alpha}, S_{\beta}, \gamma, \epsilon) = MAX((S_{\alpha} - S_{\beta}) * \gamma + \epsilon, 0) \\ \gamma = 1 \quad if \; \alpha < \beta \\ \gamma = -1 \; if \; \alpha > \beta \end{cases} \tag{3}$$

(3) describes sorting two images. SI-Ranker needs to sort multiple images simultaneously, so we extend it to sorting multiple images. It represents as:

$$L_{train\_ranker} = \mathrm{L}(\mathrm{S}, \mathrm{F}, \epsilon) = \frac{1}{n^2} \sum_{1}^{n} Max(0, (S - S^T) * F + \epsilon) \tag{4}$$

$$(S - S^T) * F = \begin{array}{c} \\ S_1 \\ S_2 \\ \vdots \\ S_n \end{array} \begin{array}{c} \begin{array}{cccc} S_1 & S_2 & \cdots & S_n \end{array} \\ \begin{pmatrix} 0 & S_1 - S_2 & \cdots & S_1 - S_n \\ S_2 - S_1 & 0 & \cdots & S_2 - S_n \\ \vdots & \vdots & \ddots & \vdots \\ S_n - S_1 & S_n - S_2 & \cdots & 0 \end{pmatrix} \end{array} * \begin{pmatrix} 0 & \cdots & 1 \\ \vdots & \ddots & \vdots \\ -1 & \cdots & 0 \end{pmatrix} \tag{5}$$

Sorting n images, we will obtain n scores. These scores are connected according to the image quality from good to bad to obtain S in (4). S^T is the transpose of S. We arrange the interpolation images by interpolation factor from low to high input R, and the coefficient matrix F corresponds to γ in (3), the upper half of the coefficient matrix F is 1, and the lower half is -1, and the main verb is 0. SI-Ranker trains alternately with G and D. R update multi(mt) times, then G and D are updated once.

Different from Rank-SRGAN: SIR-SRGAN and Rank-SRGAN add ranker to the SRGAN to constrain the reconstruction of images, but both are entirely differentexcept that they both use Learn to Rank.

1. *Sort in a different way*. Rank-SRGAN sorts the results of different SISR models based on the IQA selected; this allows the ranker to optimize the selected IQA while combining other model results indirectly. In contrast, SI-Ranker sorts the interpolated image between the original and reconstructed images without external data and IQA. This sorting makes SI-Ranker more concerned with reconstructing the difference between the image and the original image.
2. *Different training processes*. Rank-SRGAN requires separate training for ranker, and ranker needs to be frozen in adversarial training, not update. SI-Ranker trains with the

generator and the discriminator, learning the intrinsic difference between the reconstructed and original images.

3.3 Patch Distance Loss

CX loss[48][49] cuts image features into patches and regards each patch as a vector. In the patch of the original image, for the patch of the reconstructed image, find the patch with the minor cosine similarity to guide the feature reconstruction. TV-TV loss [50],[51] obtains better edges by doing total variation(TV) on the reconstructed image and the residual between the reconstructed image and the original image.

Patch Distance Loss (PDL) works in the high frequency of the image. PDL cuts the high-frequency features into 4*4 patches and regards each patch as a vector. Because the image's high frequency is sparse, the response of image texture and structure in high frequency is usually continuous (the surrounding pixels are correlate). Therefore, we use cosine similarity to measure the distance between each patch to obtain more similar vectors.

$$CS(A,B) = \frac{A * B}{||A|| * ||B||} = \frac{\sum_{i=1}^{n} A_i * B_i}{\sqrt{\sum_{i=1}^{n}(A_i)^2} * \sqrt{\sum_{i=1}^{n}(B_i)^2}} \tag{6}$$

$CS(A,B)$ is the cosine similarity of calculating vector A and B, N is the length of the vector, A_i note the ith element in vector A.

It is impossible to reconstruct an image perfectly, the distance between the reconstructed image patch and the HR patch can not be zero. When the distance between two vectors is stable, there are still many values for reconstructing the image feature vector. PDL compresses the value space by calculating the cosine similarity between the reconstructed image patch and multiple HR image patches. PDL express as:

$$PDL(CP,HP) = \frac{1}{k^2}\sum_{i=1}^{k-1}\sum_{j=i+1}^{k} |CS(CP_i,CP_j) - CS(HP_i,HP_j)| \tag{7}$$

CP is the patches from reconstructing images, and HP is the patches from the original image, k is the number of patches.

3.4 Loss function

Adversarial Loss: SIR-SRGAN base on SRGAN. G and D are consistent with SRGAN. Therefore, its adversarial loss function is the same as SRGAN.

$$\min_{\theta_G}\ \max_{\theta_D}\ \mathbb{E}_{I^{HR}\sim P(HR)}\left[\log D_{\theta_D}(I^{HR})\right] + \mathbb{E}_{I^{LR}\sim P(LR)}[\log(1 - D_{\theta_D}(G_{\theta_G}(I^{LR})))] \tag{8}$$

for the G, its loss function is:

$$L_{adv} = \log\left(D(G(LR))\right) \tag{9}$$

Perceptual Loss: Perceptual Loss often appears with GAN, and we follow [11] to calculate its loss using a feature graph from the last convolution layer of VGG19 before active. Represented as:

$$L_{perceptual} = |VGG19_{5\_4}(HR) - VGG19_{5\_4}(SR)|_1 \tag{10}$$

Rank Content Loss: We want the reconstructed image to score lower in R, so it represents as:

$$L_{ranker} = \sigma(R(G(LR))) + \sigma(FR(VGG19_{5\_4}(G(LR)))) \tag{11}$$

Where σ is the sigmoid activation function, which allows the monotony of the score to remain the same while compressing the score range to make it easier to optimize. R is the

image ranker, and FR is the feature ranker. $VGG19_{5\_4}$ represents the feature map after activation of the last convolution of VGG19.

Patch Distance Loss: PDL will act on both RGB images and high frequency.

$$L_{PDL} = PDL(G(LR), HR) + PDL(DWPT(G(LR)), DWPT(HR)) \quad (12)$$

DWPT is the Haar discrete wavelet packet transform; while discarding the separated image low frequency. Finally, we can get the G's loss function is:

$$L_G = \vartheta L_{adv} + \beta L_{perceptual} + \gamma L_{rank\_pixel} + \delta L_{rank\_feature} + \tau L_{PDL} \quad (13)$$

$\vartheta, \beta, \gamma, \delta, \tau$ are trade-off parameters.

4. Experiment

4.1 Train Details

Datasets and Evaluation Metrics: We use DIV2K[32] and Flicker2K[33] as our training set. In training, our batch size is 8, and LR's patch size is 64x64; the corresponding high-resolution image size is 256x256. We used set5[34], set14[35], BSD100[36],G100[47], Urban100[37] and PIRM2018[38] for the test set. LR is obtained by HR using bicubic downsampling in MATLAB. We use PSNR, SSIM[39], PI[40], LPIPS[46]as the metric. The higher the PSNR and SSIM, the better, and the lower the PI and LPIPS.

Train Details: During training, it is an epoch every 1000 iterations. The trade-off parameters are $\vartheta = 0.003, \beta = 1, \gamma = 0.02, \delta = 0.02, \tau = 1$. R update five times(mt=5), then G and D are updated once. and α are [0, 0.2, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]. So, this will cost three times than SRGAN. However, the generator did not make any changes. The learning rates of Discriminator and Generator initialized with 10^{-4}, the two Rankers' learning rate initialized with 10^{-3}. All those learning rates will be half of the previous at [50,125,200,300] epoch. A total of 800 epochs were trained. Same with the previous approach, we use PSNR-Orient trained networks to initialize the generator. Generators, discriminators, and ranker optimize by Adam[43], and $\beta = 0.9$. A total of 20 days of training on an NVIDIA V100.

4.2 Comparison with the-state-of-the-arts

Quantitative Analysis: We compare our methods quantitatively with SRGAN[8], SFT-SRGAN[44], Rank-SRGAN[13], ESRGAN[11], NatSR[12]. The generators of ESRGAN and NatSR are much larger than other models. The results of PSNR, SSIM, PI, and LPIPS are showing in Table 1. The best metrics for each row mark in red, the second-best indicator mark in blue, and the third in green.

We can find that SIR-SRGAN achieved the best PSNR and SSIM in all models with SRRes-Net and weaker than NatSR in comparisons of larger generator models; this shows that SIR-SRGAN can improve the consistency of the reconstructed image with the original image.

There are differences in performance between PI and LPIPS, which emphasize the visual quality of images. On PI, SIR-SRGAN is better than SRGAN but weaker than Rank-SRGAN and SFTGAN. On LPIPS, SIR-SRGAN performs best with ESRGAN. Rank-SRGAN uses NIQE to sort Images, Which makes an advantage in PI. SIR-SRGAN focuses on the difference between the reconstructing image and the original image, so it performs better on the reference metrics like LPIPS, PSNR and SSIM. Further, SIR-SRGAN is superior to NatSR on both PI and LPIPS. In summary, SIR-SRGAN improves the consistency of the reconstructed image with the original image in pixels and features to improve image quality.

Visual Quality Comparison: We also compared the visual quality of the reconstructed image, some of the results shown in Figure 3.

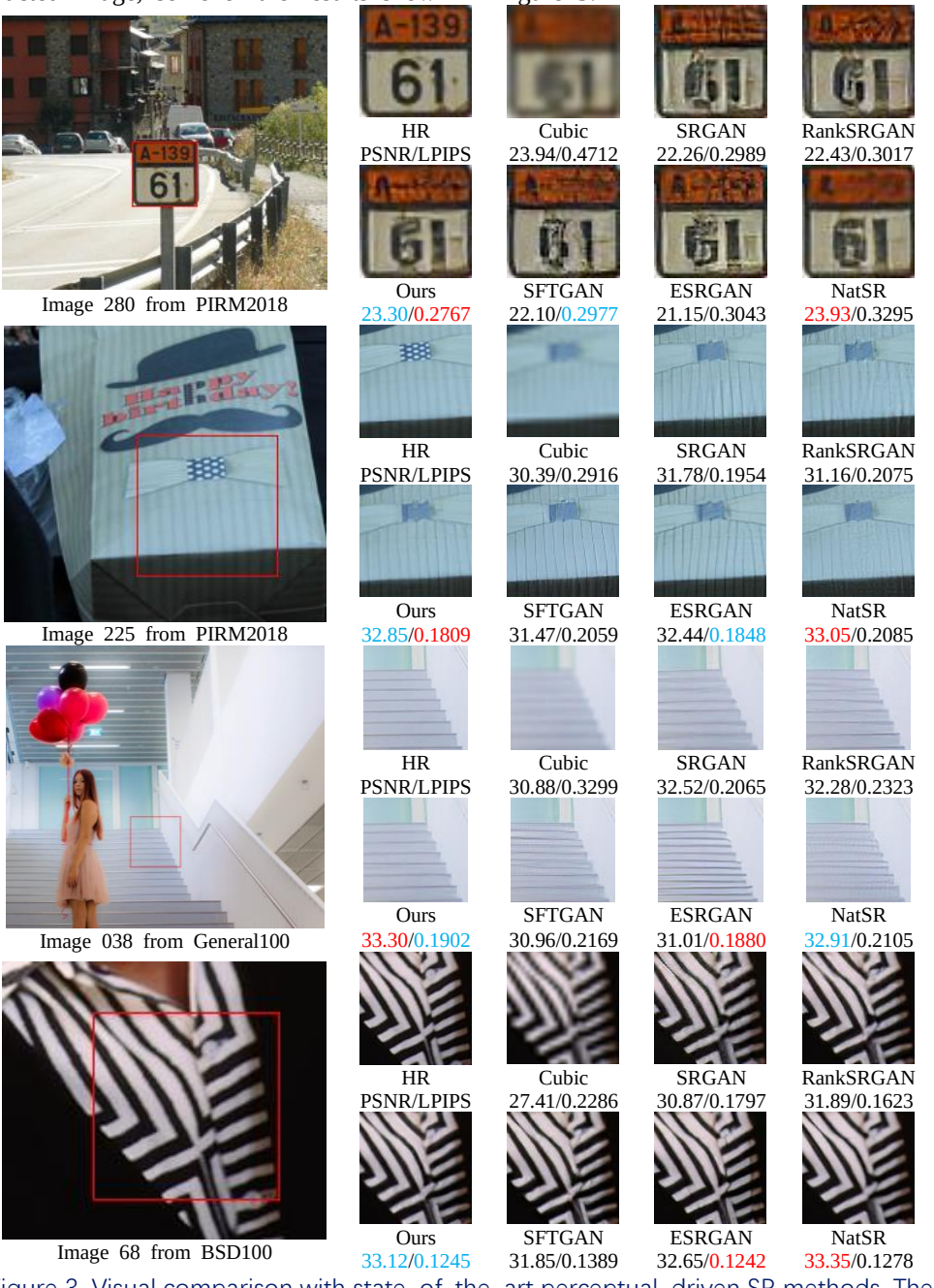


Figure 3. Visual comparison with state-of-the-art perceptual-driven SR methods. The result shows that SIR-SRGAN improves the visual quality while the model has the same generator.

Easy to find that the reconstruction results of SIR-SRGAN are closer to the original, which has the same generator. Its reconstructed images have fewer artifacts and noise, which is why they have higher PSNR. Further, SIR-SRGAN has better performance in the image structure, which we think is why SIR-SRGAN performs better on LPIPS. Compared with ESRGAN and NatSR, SIR-SRGAN is better than NatSR, the same as LPIPS. The visual quality comparison proves that the method we propose is feasible; it improves the quality of the reconstructed image without changing the generator.

Dataset	Metric	Cubic	SRGAN	SFTGAN	Rank	Ours	Esrgan	NatSR
BSD100	PSNR	25.96	25.19	25.52	25.51	26.39	25.34	26.39
	SSIM	0.6699	0.6408	0.6557	0.6487	0.6875	0.6511	0.6844
	PI	7.00	2.55	2.38	2.17	2.49	2.48	2.78
	LPIPS	0.5249	0.2998	0.2729	0.284	0.2644	0.2612	0.2987
U100	PSNR	23.13	24.38	24.01	24.52	25.14	24.35	25.45
	SSIM	0.6596	0.7312	0.7158	0.7281	0.7536	0.7335	0.7614
	PI	7.9377	3.70	3.61	3.33	3.63	3.77	3.65
	LPIPS	0.3528	0.2441	0.2566	0.2510	0.2280	0.2200	0.2512
G100	PSNR	28.02	29.26	29.01	29.13	30.09	29.39	30.35
	SSIM	0.8284	0.8102	0.8056	0.8002	0.8292	0.8094	0.833
	PI	7.9368	4.43	4.29	3.85	4.35	4.32	4.63
	LPIPS	0.3528	0.1964	0.2328	0.2061	0.1812	0.1814	0.209
Set14	PSNR	26.10	25.83	25.91	26.15	26.98	26.03	26.96
	SSIM	0.7850	0.6933	0.6977	0.7003	0.733	0.6977	0.73
	PI	7.03	3.09	2.91	2.61	3.02	2.93	3.11
	LPIPS	0.4393	0.2712	0.2745	0.253	0.2383	0.2372	0.2751
PRIM 2018	PSNR	26.47	25.43	25.00	25.47	26.50	25.04	26.95
	SSIM	0.6898	0.6598	0.6361	0.6561	0.6966	0.6452	0.7088
	PI	6.79	2.18	2.58	1.95	2.16	2.44	2.3773
	LPIPS	0.3899	0.2698	0.2987	0.2671	0.2483	0.258	0.2881

Table 1. Average PSNR, SSIM, PI, LPIPS on BSD100, Urban100, General100, Set14, PIRM2018Test.

User Study: In addition, we used User Study to compare the differences in perception between SRGAN, Rank-SRGAN, and SIR-SRGAN in the human eye. Participants in the test saw both reconstructed images from the three models and the original images, sorted by the visually closest images to the original, a total of 20 participants, each of whom observed 20 randomly selected images. The results show in Table 2:

	SIR-SRGAN	Rank-SRGAN	SRGAN
Rank-1	51.25%	45.75%	3%
Rank-2	45.75%	46.75%	7.75%
Rank-3	3%	7.5%	89.25%

Table 2. User study in SRGAN, Rank-SRGAN, and SIR-SRGAN on BSD100.

According to feedback, Rank-SRGAN's images were sharper in the background and performed better on hair. On the other hand, SIR-SRGAN's reconstructed image's feature is closer to the original image without over-sharpening and has minor artifact and noise. This result also shows that Rank-SRGAN has different characteristics from SIR-SRGAN.

4.3 Ablation study

PDL: Figure 4 shows the visual comparison of SRGAN, CX, and PDL with different patch sizes. PDL reduces the artifacts of the reconstructed image and improves the visual quality. Table 3 shows the performance of CX and PDL on BSD100. Although CX performs better on PI, the reconstructed image of CX has poor consistency with HR in pixel(PSNR) and features(LPIPS). Although the performance of using PDL is similar to that of SGRAN on PI, the visual feeling of the reconstructed image is better by suppressing artifacts and

improving the similarity of high frequency between reconstructed image and HR. Meanwhile, the consistency between reconstructed images and HRs is better by suppression o artifacts and improve the similarity of high-frequency features.

	SRGAN	SRGAN_CX	SRGAN-PDL	SIR-SRGAN(MAE)	SIR-SRGAN	SIR-SRGAN Random Rank
PSNR	25.19	24.59	26.29	26.07	26.39	26.06
SSIM	0.6408	0.6439	0.6834	0.6770	0.6875	0.6747
PI	2.55	2.36	2.52	2.48	2.49	2.72
LPIPS	0.2998	0.3105	0.2739	0.2627	0.2644	0.2985

Table 3. PDL and SI-Ranker impact on the results of the BSD100 dataset reconstruction.

The effect of patch size on PDL is significant. As analyzed above, a larger patch size will increase the dimension of the vector, thus increasing uncertainty and reducing the performance of PDL.

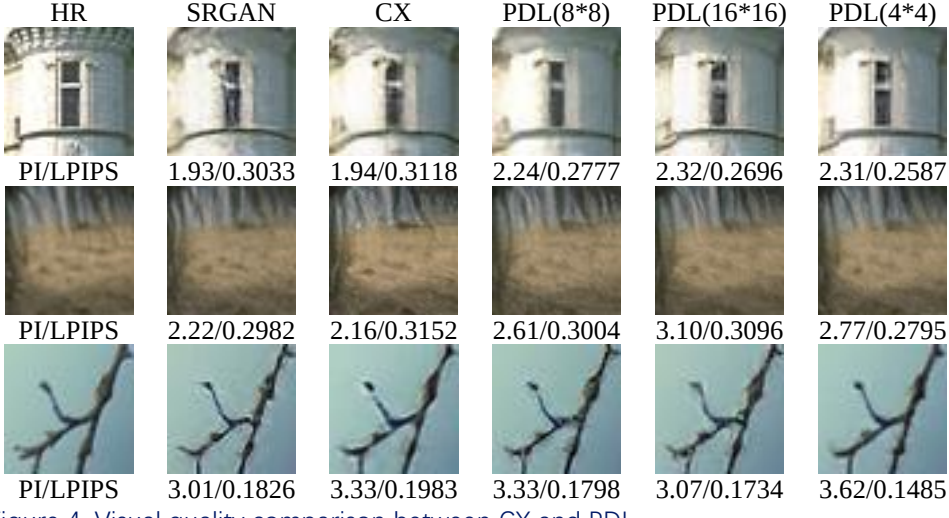
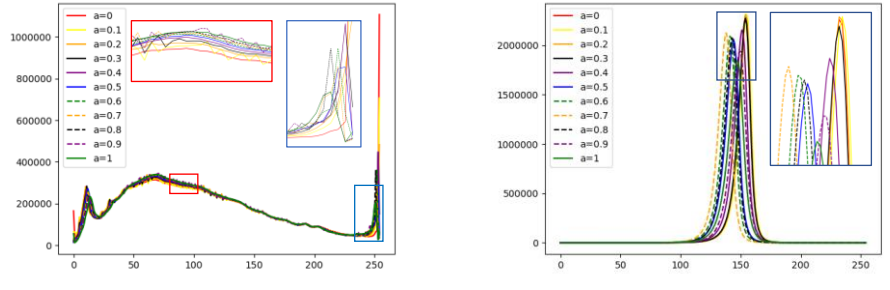


Figure 4. Visual quality comparison between CX and PDL.

SI-Ranker: The choice of interpolation factor will affect the quality of reconstructed images. The interpolation factor we propose to use is a fixed value rather than random selection in [0,1).



a. The distribution of interpolated images with different interpolation factors.

b. The distribution of interpolated images' features with different interpolation factors.

Figure 5. The distribution of pixels and features of interpolated images with different interpolation factors.

We found that the distribution difference between the interpolated image and the original image will not decrease with the decrease of the interpolation factor when $0.6 \leq \alpha \leq 1$. Figure 5.a shows the distribution of interpolated images, and Figure 5.b shows the distribution of interpolated image features extract by VGG19.

In addition, the number of times(mt) R needs to update before updating G and D also affect ranker's performance. Figure 6 shows the effect of different mt on the reconstructed image. At the same time, we also tested the effect of training ranker with incorrect ranking on the reconstructed image with mt=5.

Figure 6. Overall visual comparisons for showing the effects of each component in SIR-SRGAN.

5. Conclusion

For supervised, perceptive single image super-resolution, we propose a SIR-SRGAN independent of the generator's structure. It is using a ranker to sort the interpolated images of the reconstructed image from the original image. The ranker we proposed does not require any IQA but focused directly on reconstructing the intrinsic differences between the image and the original image and improving the consistency of the reconstructed image with the original image. Further, we propose PDL, which constrains the reconstruction of an image by cutting the image and calculating the cosine distance between two. Extensive experiments have proved that SIR-SRGAN is a practical framework; when using the same generator, SIR-SRGAN can effectively improve the consistency of the reconstructed image with the original image on pixels and features.

Acknowledgement

This study was supported by the grant of Chengdu Research Base of Giant Panda Bree ding (Nos: 2018CPB-C01 and 2020CPB-C09).

References

[1] Lecun Y , Bottou L . Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11):2278-2324.

[2] Dong C , Loy C C , He K , et al. Learning a Deep Convolutional Network for Image Super-Resolution[J]. 2014.

[3] Kim J , Lee J K , Lee K M . Accurate Image Super-Resolution Using Very Deep Convolutional Networks[C] IEEE Conference on Computer Vision & Pattern Recognition. IEEE, 2016.

[4] Shi W , Caballero J , Ferenc Huszár, et al. Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network[J]. 2016.

[5] Li J , Fang F , Mei K , et al. Multi-scale Residual Network for Image Super-Resolution: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VIII[C] European Conference on Computer Vision. Springer, Cham, 2018.

[6] Haris M , Shakhnarovich G , Ukita N . Deep Back-Projection Networks For Super-Resolution[J]. 2018.

[7] Sajjadi M S M , Schlkopf B , Hirsch M . EnhanceNet: Single Image Super-Resolution Through Automated Texture Synthesis[J]. 2016.

[8] Ledig C , Theis L , Huszar F , et al. Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network[J]. 2016.

[9] I.Goodfellow,J.Pouget-Abadie,M.Mirza,B.Xu,D.Warde-Farley,S.Ozair,A.Courville,andY.Bengio.Generative adversarial nets. In Advances in Neural Information Processing Systems (NIPS), pages 2672–2680, 2014. 3, 4, 6

[10] Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks[J]. arXiv preprint arXiv:1511.06434, 2015.

[11] Wang X, Yu K, Wu S, et al. Esrgan: Enhanced super-resolution generative adversarial networks[C]Proceedings of the European Conference on Computer Vision (ECCV). 2018: 0-0.

[12] Soh J W, Park G Y, Jo J, et al. Natural and realistic single image super-resolution with explicit natural manifold discrimination[C] Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2019: 8122-8131.

[13] Zhang W, Liu Y, Dong C, et al. Ranksrgan: Generative adversarial networks with ranker for image super-resolution[C] Proceedings of the IEEE International Conference on Computer Vision. 2019: 3096-3105.

[14] Ma C, Rao Y, Cheng Y, et al. Structure-Preserving Super Resolution with Gradient Guidance[C] Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 7769-7778.

[15] Lim B, Son S, Kim H, et al. Enhanced deep residual networks for single image super-resolution[C] Proceedings of the IEEE conference on computer vision and pattern recognition workshops. 2017: 136-144.

[16] Zhang Y, Tian Y, Kong Y, et al. Residual dense network for image super-resolution[C] Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 2472-2481.

[17] Zhang Y, Li K, Li K, et al. Image super-resolution using very deep residual channel attention networks[C] Proceedings of the European Conference on Computer Vision (ECCV). 2018: 286-301.

[18] Johnson J, Alahi A, Fei-Fei L. Perceptual losses for real-time style transfer and super-resolution[C] European conference on computer vision. Springer, Cham, 2016: 694-711.

[19] Jolicoeur-Martineau A. The relativistic discriminator: a key element missing from standard GAN[J]. arXiv preprint arXiv:1807.00734, 2018.

[20] Parikh D, Grauman K. Relative attributes[C] 2011 International Conference on Computer Vision. IEEE, 2011: 503-510.

[21] Souri Y, Noury E, Adeli E. Deep relative attributes[C] Asian conference on computer vision. Springer, Cham, 2016: 118-133.

[22] Saquil Y, Kim K I, Hall P. Ranking cgans: Subjective control over semantic image attri butes[J]. arXiv preprint arXiv:1804.04082, 2018.
[23] Liu X, van de Weijer J, Bagdanov A D. Rankiqa: Learning from rankings for no-refere nce image quality assessment[C] Proceedings of the IEEE International Conference on C omputer Vision. 2017: 1040-1049.
[24] Ji H, Fermüller C. Robust wavelet-based super-resolution reconstruction: theory and algor ithm[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2008, 31(4): 6 49-660.
[25] Aydin V A, Foroosh H. Super-Resolution of Wavelet-Encoded Images[J]. arXiv preprint arXiv:1705.01258, 2017.
[26] Zhong Z, Shen T, Yang Y, et al. Joint sub-bands learning with clique structures for wa velet domain super-resolution[C] Advances in Neural Information Processing Systems. 20 18: 165-175.
[27] Demirel H, Anbarjafari G. Image resolution enhancement by using discrete and stationar y wavelet decomposition[J]. IEEE transactions on image processing, 2010, 20(5): 1458-1 460.
[28] Zhao Y, Wu A. Super-Resolution Image Reconstruction Based on Wavelet Transform an d Edge-Directed Interpolation[J]. Journal of Communication and Computer, 2015, 12: 73-78.
[29] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image reco gnition[J]. arXiv preprint arXiv:1409.1556, 2014.
[30] Lin M, Chen Q, Yan S. Network in network[J]. arXiv preprint arXiv:1312.4400, 2013.
[31] Geirhos R, Rubisch P, Michaelis C, et al. ImageNet-trained CNNs are biased towards te xture; increasing shape bias improves accuracy and robustness[J]. arXiv preprint arXiv:18 11.12231, 2018.
[32] Agustsson E, Timofte R. Ntire 2017 challenge on single image super-resolution: Dataset and study[C] Proceedings of the IEEE Conference on Computer Vision and Pattern Rec ognition Workshops. 2017: 126-135.
[33] Timofte, R., Agustsson, E., Van Gool, L., Yang, M.H., Zhang, L., Lim, B., Son, S., Ki m, H., Nah, S., Lee, K.M., et al.: Ntire 2017 challenge on single image super-resolutio n: Methods and results. In: CVPRW. (2017)
[34] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie-Line Alberi-Morel. Lo w-complexity single-image super-resolution based on nonnegative neighbor embedding. In BMVC, 2012.
[35] Roman Zeyde, Michael Elad, and Matan Protter. On single image scale-up using sparse-representations. In ICCS, pages 711–730. Springer, 2010.
[36] dra Malik. A database of human segmented natural images and its application to evaluat ing segmentation algorithms and measuring ecological statistics. In ICCV, pages 416–42 5, 2001.
[37] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In CVPR, pages 5197–5206, 2015.
[38] Yochai Blau, Roey Mechrez, Radu Timofte, Tomer Michaeli, and Lihi Zelnik-Manor. 20 18 pirm challenge on perceptual image super-resolution. arXiv preprint arXiv:1809.07517, 2018.Wang Z, Bovik A C, Sheikh H R, et al. Image quality assessment: from error vi sibility to structural similarity[J]. IEEE transactions on image processing, 2004, 13(4): 60 0-612.
[39] Yochai Blau, Roey Mechrez, Radu Timofte, Tomer Michaeli, and Lihi Zelnik-Manor. Th e 2018 pirm challenge on perceptual image super-resolution. In ECCV, pages 334–355. Springer, 2018.
[40] Ma C, Yang C Y, Yang X, et al. Learning a no-reference quality metric for single-ima ge super-resolution[J]. Computer Vision and Image Understanding, 2017, 158: 1-16.
[41] Mittal A, Soundararajan R, Bovik A C. Making a “completely blind” image quality anal yzer[J]. IEEE Signal processing letters, 2012, 20(3): 209-212.
[42] Kingma D P, Ba J. Adam: A method for stochastic optimization[J]. arXiv preprint arXi v:1412.6980, 2014.

[43] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture i n image super-resolution by deep spatial feature transform. In CVPR, pages 606–615, 20 18.
[44] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C] Proceedi ngs of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
[45] Ulyanov D, Vedaldi A, Lempitsky V. Instance normalization: The missing ingredient for fast stylization[J]. arXiv preprint arXiv:1607.08022, 2016.
[46] Zhang R , Isola P , Efros A A , et al. The Unreasonable Effectiveness of Deep Featur es as a Perceptual Metric[C] IEEE/CVF Conference on Computer Vision & Pattern Rec ognition. IEEE, 2018.
[47] Chao D , Chen C L , Tang X . Accelerating the Super-Resolution Convolutional Neural Network[C] European Conference on Computer Vision. Springer, Cham, 2016.
[48] Mechrez R , Talmi I , Shama F , et al. Maintaining Natural Image Statistics with the Contextual Loss[J]. Springer, Cham, 2018.
[49] Mechrez R , Talmi I , Zelnik-Manor L . The Contextual Loss for Image Transformati on with Non-Aligned Data[C]// European Conference on Computer Vision. Springer, Cha m, 2018.
[50] Vella M , Mota J . Single Image Super-Resolution via CNN Architectures and TV-TV Minimization[J]. 2019.
[51] Vella M , Mota J . Robust Single-Image Super-Resolution via CNNs and TV-TV Mini mization[J]. 2020.
[52] Blau, Y., Michaeli, T.: The perception-distortion tradeoff. In: CVPR. (2017)