Investigating Loss Functions for Extreme Super-Resolution

Younghyun Jo[1] Sejong Yang[1] Seon Joo Kim[1,2]
[1]Yonsei University [2]Facebook

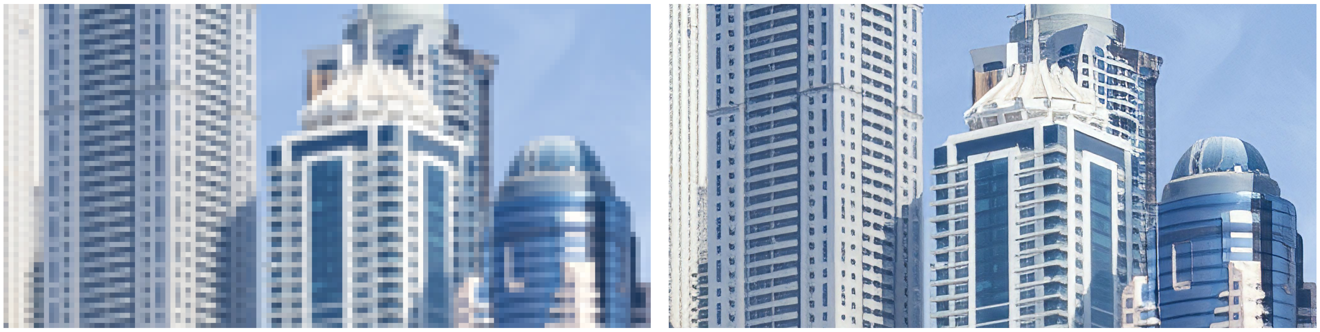

Figure 1. We train deep networks using a new loss function for perceptual ×16 super-resolution, and fine details are successfully restored. Left image is input and right one is output. Please zoom-in for details.

Abstract

The performance of image super-resolution (SR) has been greatly improved by using convolutional neural networks. Most of the previous SR methods have been studied up to ×4 upsampling, and few were studied for ×16 upsampling. The general approach for perceptual ×4 SR is using GAN with VGG based perceptual loss, however, we found that it creates inconsistent details for perceptual ×16 SR. To this end, we have investigated loss functions and we propose to use GAN with LPIPS [23] loss for perceptual extreme SR. In addition, we use U-net structure discriminator [14] together to consider both the global and local context of an input image. Experimental results show that our method outperforms the conventional perceptual loss, and we achieved second and first place in the LPIPS and PI measures respectively for NTIRE 2020 perceptual extreme SR challenge.

1. Introduction

Super-resolution (SR) is the task of generating a high-resolution (HR) image from a given low-resolution (LR) image. SR has been used for many applications like surveillance, satellite, medical, microscopy imaging, and so on. Recently, compression and SR could be a solution for high-quality multimedia streaming services to reduce network bandwidth usage.

Now in the era of deep learning, the performance of image SR has been greatly improved by using convolutional neural networks [2]. However, most of the methods have been studied up to ×4 upsampling, and few were studied for ×16 upsampling [1, 17]. There are three aspects to consider for the new perceptual ×16 upscaling method: datasets, network designs, and loss functions.

Several studies have focused on developing effective deep network structures using datasets containing high-quality images, while the loss function remains unchanged. In general, the adversarial training [3] and the VGG [15] based perceptual loss [7] have been used for perceptual SR. However, we empirically found that they give insufficient performance due to inconsistently hallucinated details for the perceptual ×16 SR.

To this end, we have focused on investigating new loss functions and we found that learned perceptual similarity (LPIPS) [23] is a better choice for the loss function. Also, we adopt U-net structure discriminator to fully utilize the global and local context for better details [14]. We will show that our method generates visually pleasing results with consistent details in the experiments section. Especially in NTIRE 2020 perceptual extreme SR challenge [22], our method ranked second and first in the LPIPS and PI [1] measures respectively for the results.[1]

[1] Codes are available at https://github.com/kingsj0405/ciplab-NTIRE-2020

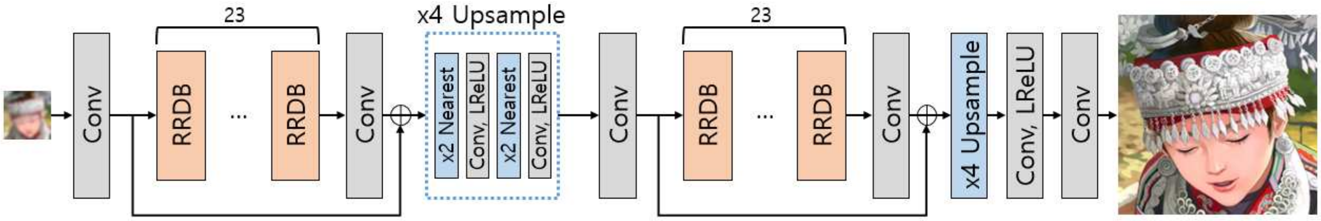

Figure 2. Our generator structure for ×16 SR. We adopt the generator structure of ESRGAN [19], and double the backbone layers. First half of the whole network performs ×4 upsampling and the other half performs remaining ×4 upsampling.

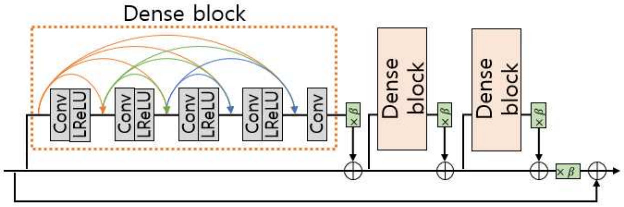

Figure 3. Structure of RRDB. β is the scaling parameter.

2. Related Work

Perceptual Super-Resolution In the early days of deep SR, mean squared error (MSE) has been used to train the model for achieving higher peak signal noise to the ratio (PSNR) – a common image quality assessment metric. However, PSNR is known to be not correlated well to the human visual perception [7, 23], and it suffers from blurry results as it does not consider to create high-frequency details.

To create realistic details, SRGAN [9] used generative adversarial networks (GAN) with the perceptual loss based on VGG features [7]. Since then, many new deep network architectures have been devised to improve the performance. SFTGAN [18] proposed a spatial feature transform layer to efficiently incorporate the categorical conditions information. ESRGAN [19] introduced a residual-in-residual dense block (RRDB) to effectively train a deeper model showing superior performance.

In other directions, EnhanceNet [13] used texture loss to enhance detailed textures, SRFeat [11] suggested an additional discriminator in feature domain, NatSR [16] introduced natural manifold for maintaining the naturalness of results, SROBB [12] exploited texture segmentation labels for region adaptive SR, and RankSRGAN [24] adopted a ranker to force the results get higher ranking.

They are mostly based on the GAN with the VGG perceptual loss, and there were few considerations about the loss functions. In this paper, we have investigated loss functions for perceptual extreme SR and replace the VGG perceptual loss with the LPIPS perceptual loss.

Extreme Super-Resolution A number of methods have emerged for ×4 SR, however, there were few studies for ×8 SR and over. For ×8 SR, LapSRN [8] proposed a progressive upsampling approach using Laplacian pyramid, DBPN [6] proposed an iterative up and downsampling approach to exploit HR features, and RCAN [25] proposed a residual channel attention network to adaptively weight across channels.

The 2018 PIRM challenge on perceptual image super-resolution [1] and NTIRE 2018 challenge on single image super-resolution [17] were held for perceptual ×4 SR. Recently, in AIM 2019 challenge on image extreme super-resolution [4], DIV8K dataset [5] for extreme SR was released and several methods were proposed for ×16 SR [10, 20]. They focused on the network architectures for the performance, however, less on the loss function.

3. Method

For ×16 SR, using MSE hardly restores high-frequency details. Therefore, we adopt the GAN [3] framework which is widely known to be able to hallucinate details for SR [9]. Our method consists of a generator and a discriminator, and we focus on investigating the loss functions for the perceptual performance.

3.1. Generator

There are few studies for the network structures for ×16 SR. Several methods appeared in the last year AIM challenge [4], and most of them cascade two ×4 SR networks for ×16 SR task. Similarly, we adopt one of the state-of-the-art ×4 SR network ESRGAN [19] as our generator network, and double the backbone layers for having enough network capacity for ×16 SR (Fig. 2). ESRGAN removes batch normalization layers from SRGAN [9] to avoid unpleasant artifacts and replaces the original residual block with the RRDB to boost performance (Fig. 3). Specifically, we double the existing 23 RRDBs to 48 RRDBs. The first half of the network performs ×4 upsampling and the other half performs the remaining ×4 upsampling. Formally, the generator produces ×16 super-resolved output image I^{Gen} from an input image I^{In}:

$$I^{Gen} = G(I^{In}). \tag{1}$$

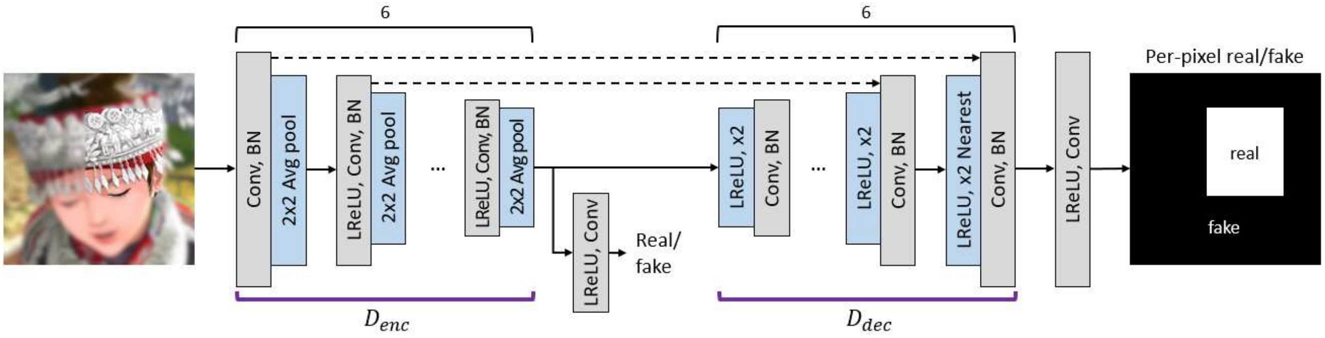

Figure 4. Our discriminator network. To provide per-pixel feedback to the generator, U-net structure [14] is adopted. There are 6 downsampling and 6 upsampling stages, with skip-connections between them.

Note that our method focuses on investigating loss function combinations for better performance.

3.2. Discriminator

Most GAN based SR methods used to encoder structure discriminator [9, 19, 24]. Compressed representation from the discriminator determines whether an input image is real or fake. Recently, there has been a study to improve the performance of image generation using an U-net structure discriminator [14]. On top of the normal encoder structure D_{enc}, they successively attached a decoder structure D_{dec} for providing per-pixel feedback to the generator while maintaining global context (Fig. 4). We empirically found that this gives more details for ×16 SR rather than normal encoder structure discriminator.

In practice, discriminator loss $\mathcal{L}_D$ is computed at both the encoder head $\mathcal{L}_{D_{enc}}$ and the decoder head $\mathcal{L}_{D_{dec}}$. We can formulate the discriminator loss as hinge loss.

$$\begin{aligned}\mathcal{L}_{D_{enc}} = &-\mathbb{E}\Big[\sum_{i,j}\min(0,-1+[D_{enc}(I^{GT})]_{i,j})\Big] \\ &-\mathbb{E}\Big[\sum_{i,j}\min(0,-1-[D_{enc}(I^{Gen})]_{i,j})\Big],\end{aligned} \tag{2}$$

$$\begin{aligned}\mathcal{L}_{D_{dec}} = &-\mathbb{E}\Big[\sum_{i,j}\min(0,-1+[D_{dec}(I^{GT})]_{i,j})\Big] \\ &-\mathbb{E}\Big[\sum_{i,j}\min(0,-1-[D_{dec}(I^{Gen})]_{i,j})\Big],\end{aligned} \tag{3}$$

where I^{GT} is the ground truth image, and $[D(I)]_{i,j}$ is the discriminator decision at pixel (i,j). The corresponding adversarial loss for the generator is as follows:

$$\mathcal{L}_{adv} = -\mathbb{E}\Big[\sum_{i,j}[D_{enc}(I^{Gen})]_{i,j} + \sum_{i,j}[D_{dec}(I^{Gen})]_{i,j}\Big]. \tag{4}$$

To explicitly encourage the discriminator to focus more on semantic and structural changes, a technique called consistency regularization was additionally used in [14]. It synthesizes a new training sample by using CutMix transformation [21], and minimizes the loss $\mathcal{L}_{D_{cons}}$. We apply it to our SR problem, and the total discriminator loss is:

$$\mathcal{L}_D = \mathcal{L}_{D_{enc}} + \mathcal{L}_{D_{dec}} + \mathcal{L}_{D_{cons}}. \tag{5}$$

3.3. Loss Functions

General choice of the loss functions for the perceptual SR methods is the adversarial loss $\mathcal{L}_{adv}$ [3] with the VGG [15] based perceptual loss $\mathcal{L}_{vgg}$ [7]. This loss combination has worked well for ×4 SR [9, 19], however, we empirically found that it does not output satisfactory results for ×16 SR due to highly hallucinated noise and less precise details. Because VGG network is trained for image classification, it may not the best choice for the SR task.

To this end, we use LPIPS [23] for the perceptual loss:

$$\mathcal{L}_{lpips} = \sum_k \tau^k(\phi^k(I^{Gen}) - \phi^k(I^{GT})), \tag{6}$$

where ϕ is a feature extractor, τ transforms deep embedding to scalar LPIPS score, and the score is computed and averaged from k layers (Fig. 5). Assume there is a reference image, and we transform the image in two different ways – small translation and blurring. Traditional image quality metrics like PSNR and SSIM prefer the blurred image, but humans are more likely to prefer the translated one. LPIPS is trained with a dataset of human perceptual similarity judgments and more appropriately reflects the human perception preferences than the VGG perceptual loss.

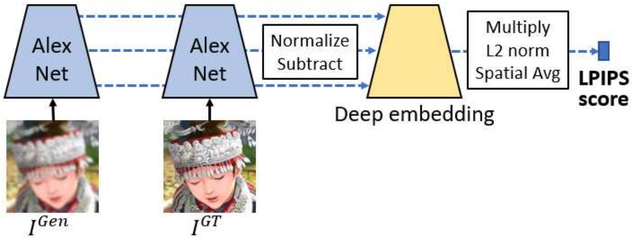

Figure 5. LPIPS [14] is computed from deep feature embeddings.

We additionally use the discriminator's feature matching loss $\mathcal{L}_{fm}$ to alleviate the undesirable noise from the adversarial loss:

$$\mathcal{L}_{fm} = \sum_{l} \mathcal{H}(D^l(I^{Gen}), D^l(I^{GT})), \tag{7}$$

where D^l denotes the activations from the l-th layer of the discriminator D, and $\mathcal{H}$ is the Huber loss (smooth L1 loss). We also use a loss on the pixel space $\mathcal{L}_{pix}$ for preventing color permutation:

$$\mathcal{L}_{pix} = \mathcal{H}(I^{Gen}, I^{GT}). \tag{8}$$

To sum up, the total loss for our generator is:

$$\mathcal{L}_G = \lambda_{adv}\mathcal{L}_{adv} + \lambda_{fm}\mathcal{L}_{fm} + \lambda_{lpips}\mathcal{L}_{lpips} + \mathcal{L}_{pix}, \tag{9}$$

where λs are scaling parameters.

4. Experiments

Dataset For training, we use DIV8K [5] dataset which contains 1500 training images with the resolution up to 8K. It highly covers diverse scene contents and is designed for ×16 SR and further. We randomly crop 384×384 size patches from the training images and synthesize corresponding input patches by bicubic downsampling to 24×24 size. To augment the training data, rotation and left-right flip are randomly applied. For the validation dataset, we uniformly select 10 images from the training images (*i.e.* 0150.png, 0300.png, ..., 1500.png).

Implementation details Our method is implemented in PyTorch 1.2.0 and trained on a single NVIDIA TITAN XP GPU (12G). The negative slopes of leaky relu are 0.2 and 0.1 for our generator and discriminator respectively. We first pretrain the generator using $\mathcal{L}_{pix}$ only for 50K iterations with mini-batch size of 3, and it takes 12 hours (`Pretrained`). We further train the generator using our full loss functions for about 60K iterations with mini-batch size of 2, and it takes about 15 hours (`Ours`). We empirically set $\lambda_{adv} = 1E{-}3$, $\lambda_{fm} = 1$, and $\lambda_{lpips} = 1E{-}6$. We use Adam optimizer and learning rate is set to 0.00001 for training both the generator and the discriminator networks. The number of parameters for our generator and discriminator is 33M and 13M respectively.

4.1. Comparisons with other methods

There were no public codes for other ×16 SR methods. Instead, we compare with bicubic upsampling and our network trained using the VGG perceptual loss (`Adv+VGG`, $\mathcal{L}_G = 1E-3 \cdot \mathcal{L}_{adv} + 1E-5 \cdot \mathcal{L}_{vgg} + \mathcal{L}_{pix}$).

Quantitative result on our validation set is shown in Table 1. We use two conventional image quality assessment metrics PSNR and SSIM (the higher the better). However, they may not reflect the human visual perception, we additionally measure LPIPS for the perceptual quality (the lower the better). In perceptual SR, LPIPS metric is more considered than PSNR and SSIM, and our method achieves the best score. Note that we crop border 16px to avoid boundary artifacts when measuring.

Qualitative results on the DIV8K testset are shown in Fig. 6. In general, as in hair and lines, our results look qualitatively improved than Adv+VGG. This is the effect of using LPIPS loss as it provides better feature space than VGG for improving the perceptual quality. This is also the effect of U-net discriminator as it considers both the global and local context and gives effective feedback to the generator.

More results on general image super-resolution test sets (Set5, Set14, BSDS100, Urban100, and Manga109) are also shown in Table 2 and Fig. 7. On those test sets, our method increases sharpness with consistent details and achieves the lower LPIPS score at the same time (we will describe `Ours-New` model in the following section).

NTIRE challenge Using the results of Ours model, we obtained 22.77 (15th), 0.5251 (16th), 0.352 (2nd), and 3.76 (1st) for PSNR, SSIM, LPIPS, and PI respectively in the results of NTIRE 2020 perceptual extreme SR challenge. They are calculated on the center 1000×1000 subimages of the DIV8K testset.

4.2. Ablation study

To investigate the effect of each loss term, we run the ablation study on our validation set. Detailed loss configurations and the corresponding quantitative and qualitative results are shown in Table 3 and Fig. 8 respectively. When only $\mathcal{L}_{lpips}$ is used without GAN framework (Ours w/o $\mathcal{L}_{adv}$), the LPIPS value is the lowest, however, the actual visual results are not the best and repetitive pattern artifacts occur to create inconsistent details. Even if LPIPS value increases, better visual results are generated by using our proposed full losses. We can see that better LPIPS value does not always guarantee better visual quality.

We further adjust λ_{lpips} from $1E-6$ to $1E-3$ and

Method	PSNR↑	SSIM↑	LPIPS↓
Bicubic	25.05	0.1211	0.6620
Pretrained	**26.22**	**0.1704**	0.5559
Adv+VGG	24.14	0.1178	0.4115
Ours	23.42	0.1183	**0.3357**

Table 1. Quantitative results with other methods on our validation set. Our proposed method shows better LPIPS value than the VGG based perceptual loss, and it is also appeared qualitatively in Fig. 6. Best values are shown in bold and second best values are underlined.

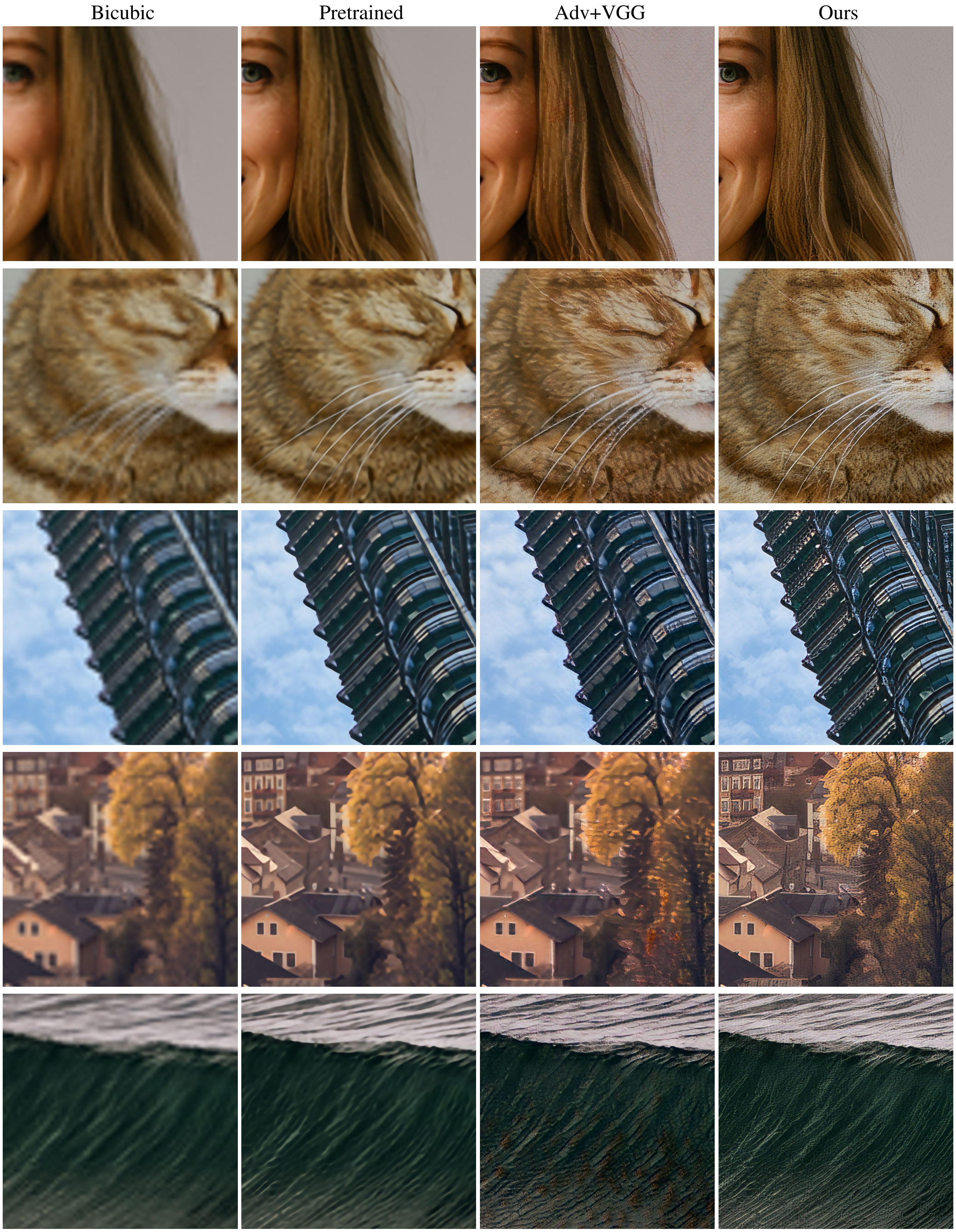

Figure 6. Qualitative results with other methods on our validation set (images are 1601, 1608, 1624, 1638, and 1658 from top to bottom). Pretrained model has limitations in restoring sharpness. Adv+VGG model produces realistic results while it sometimes produces color artifacts and inconsistent details at the same time. However, Ours model generates more visually pleasing results. Please zoom for better comparison.

Method	Set5			Set14			B100			Urban100			Manga109		
	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
Bicubic	21.37	0.1730	0.6836	20.73	0.1389	0.7536	21.74	0.1086	0.8347	18.92	0.0892	0.8474	19.12	0.1349	0.6979
Pretrained	**23.27**	**0.2602**	0.4726	**22.18**	**0.1925**	0.5818	**22.75**	**0.1491**	0.6989	**20.32**	**0.1630**	0.6086	**20.97**	**0.1877**	0.4564
Adv+VGG	21.53	0.1842	0.3554	20.95	0.1455	0.4363	21.29	0.1035	0.5016	19.22	0.1138	0.4816	20.04	0.1388	0.4023
Ours	20.97	0.1837	0.3001	20.14	0.1359	0.3888	20.21	0.0877	0.4233	18.68	0.1224	0.4056	19.50	0.1657	0.3096
Ours-New	21.85	0.1891	**0.2768**	20.75	0.1362	**0.3601**	21.24	0.0946	**0.3986**	19.24	0.1221	**0.3865**	19.94	0.1431	**0.2976**

Table 2. Quantitative results with other methods on general image super-resolution test sets. Our methods show better LPIPS values among them, and we further improve the performance in the Ours-New model.

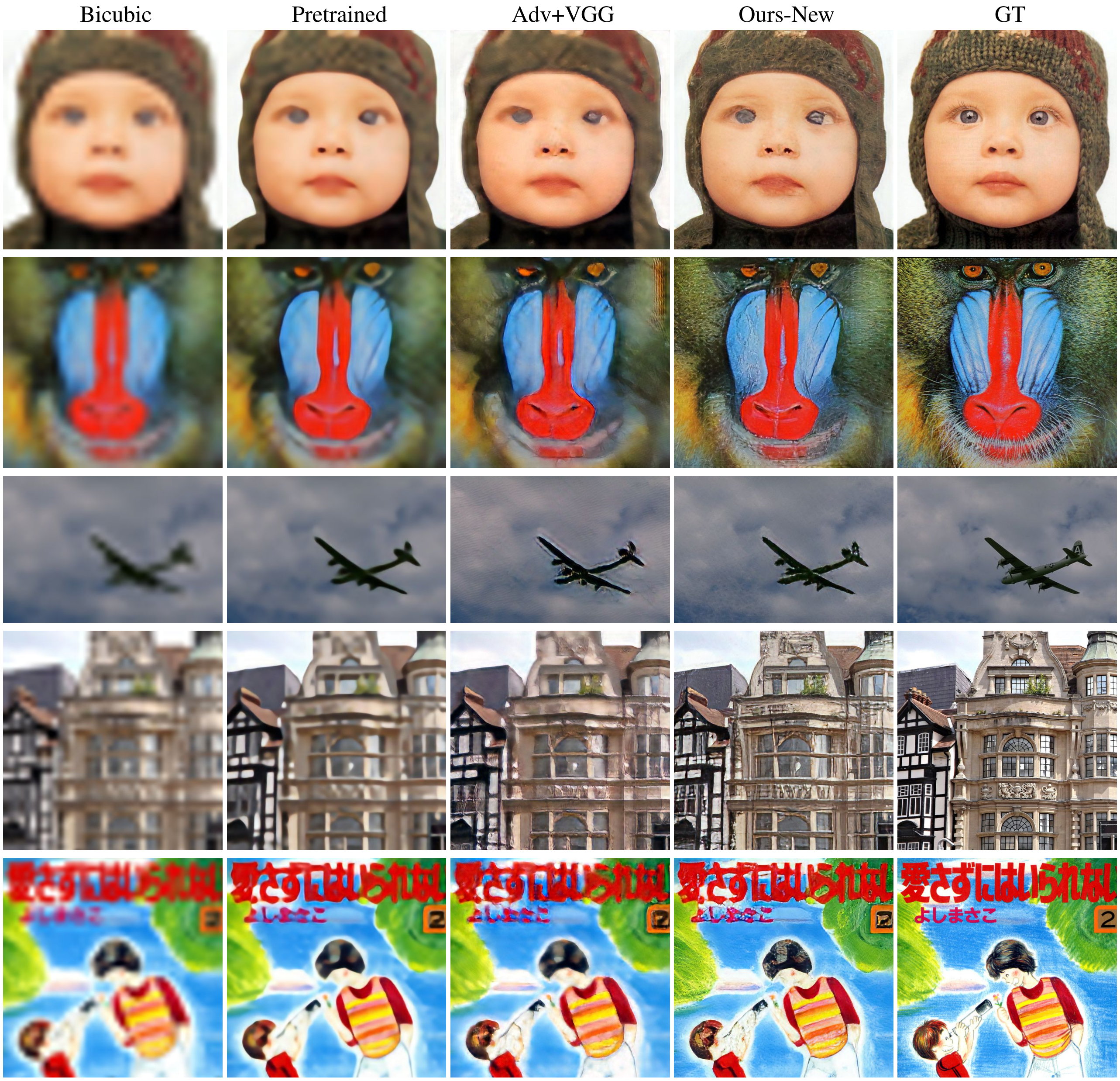

Figure 7. Qualitative results with other methods on the general image super-resolution test sets (images are baby, baboon, 3096, img_053, and AisazuNihaIrarenai from top to bottom). The Adv+VGG model improves the sharpness from the pretrained model. However, our method further enhances the results with consistent details. Please zoom for better comparison.

Method	D	λ_{adv}	λ_{fm}	λ_{lpips}	PSNR↑	SSIM↑	LPIPS↓
Ours w/o $\mathcal{L}_{adv}$	-	0	0	1	24.27	0.1155	**0.2608**
Ours w/o $\mathcal{L}_{fm}$ and $\mathcal{L}_{lpips}$	U	1E-3	0	0	22.82	0.0950	0.4202
Ours w/o $\mathcal{L}_{lpips}$	U	1E-3	1	0	24.39	0.1160	0.3437
Ours	U	1E-3	1	1E-6	23.42	0.1183	0.3357
Ours-New	U	1E-3	1	1E-3	**24.54**	**0.1225**	0.2860
Ours-New w/o D_{dec}	E	1E-3	1	1E-3	23.79	0.1126	0.2933

Table 3. Various loss configurations for ablation study on our validation set. U and E denote U-net and encoder structure for the discriminator. Using only $\mathcal{L}_{lpips}$ without GAN framework (first row) gives the best LPIPS value, but this is not reflected to visually pleasing images (see Fig. 8). Even if LPIPS value increases, the better visual quality is obtained by using all suggested losses. From Ours model, we further improve the performance with the optimized weight balance (`Ours-New`).

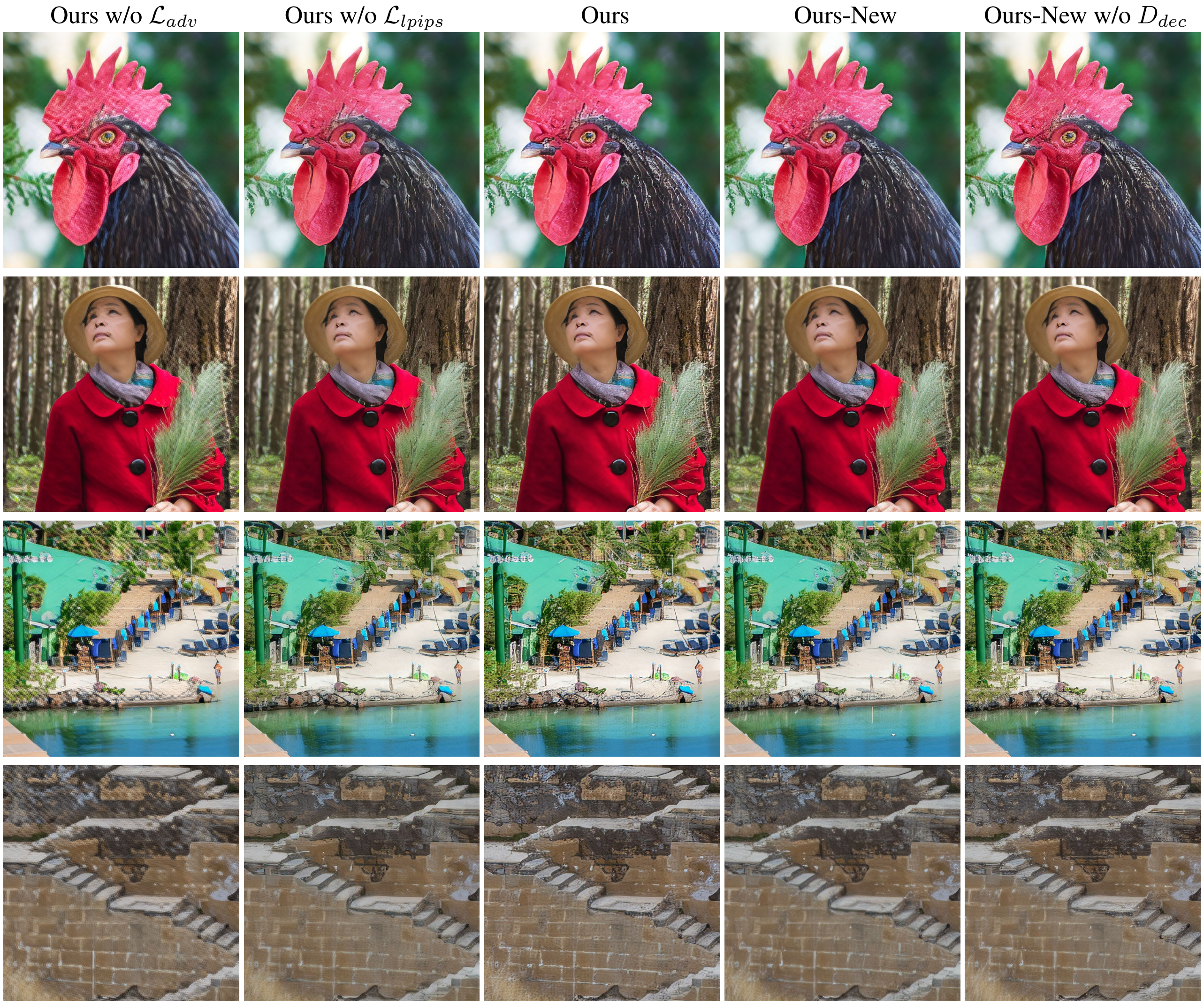

Figure 8. Qualitative results of the ablation study (images are 0300, 0600, 0900, and 1350 from top to bottom). Using only $\mathcal{L}_{lpips}$ without GAN framework (first column) shows repetitive pattern artifacts to create details. Ours results look oversharpened while Ours-New results show moderate details. In addition, Ours-New suppresses undesirable noise compared to Ours-New w/o D_{dec}. Please zoom for better comparison.

obtain better results (`Ours-New`). Ours-New results have moderate details which look more pleasing than Ours results. Also, we verify that the U-net discriminator boosts the performance by suppressing noise (Ours-New versus Ours-New w/o D_{dec}). Note that our NTIRE submission is Ours and Ours-New is the further improved version.

5. Conclusion

We investigate the loss functions for perceptual ×16 SR. Through the experiments, we verify that LPIPS is the better choice than VGG as a perceptual loss for the extreme SR. Our method achieved second place in the perceptual measure in NTIRE 2020 perceptual extreme SR challenge. Moreover, we further improve our results by appropriately balancing loss weights. In the future, more performance gain is expected by combining the proposed loss functions with a more effective generator architecture.

References

[1] Yochai Blau, Roey Mechrez, Radu Timofte, Tomer Michaeli, and Lihi Zelnik-Manor. The 2018 pirm challenge on perceptual image super-resolution. In *ECCVW*, pages 0–0, 2018. 1, 2

[2] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE TPAMI*, 38(2):295–307, 2015. 1

[3] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, pages 2672–2680, 2014. 1, 2, 3

[4] Shuhang Gu, Martin Danelljan, Radu Timofte, Muhammad Haris, Kazutoshi Akita, Greg Shakhnarovic, Norimichi Ukita, Pablo Navarrete Michelini, Wenbin Chen, Hanwen Liu, et al. Aim 2019 challenge on image extreme super-resolution: Methods and results. In *ICCVW*, pages 3556–3564. IEEE, 2019. 2

[5] Shuhang Gu, Andreas Lugmayr, Martin Danelljan, Manuel Fritsche, Julien Lamour, and Radu Timofte. Div8k: Diverse 8k resolution image dataset. In *ICCVW*, pages 3512–3516. IEEE, 2019. 2, 4

[6] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Deep back-projection networks for super-resolution. In *CVPR*, pages 1664–1673, 2018. 2

[7] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2016. 1, 2, 3

[8] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *CVPR*, pages 624–632, 2017. 2

[9] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, pages 4681–4690, 2017. 2, 3

[10] Pablo Navarrete Michelini, Wenbin Chen, Hanwen Liu, and Dan Zhu. Mgbpv2: Scaling up multi-grid back-projection networks. *arXiv preprint arXiv:1909.12983*, 2019. 2

[11] Seong-Jin Park, Hyeongseok Son, Sunghyun Cho, Ki-Sang Hong, and Seungyong Lee. Srfeat: Single image super-resolution with feature discrimination. In *ECCV*, pages 439–455, 2018. 2

[12] Mohammad Saeed Rad, Behzad Bozorgtabar, Urs-Viktor Marti, Max Basler, Hazim Kemal Ekenel, and Jean-Philippe Thiran. Srobb: Targeted perceptual loss for single image super-resolution. In *ICCV*, pages 2710–2719, 2019. 2

[13] Mehdi SM Sajjadi, Bernhard Scholkopf, and Michael Hirsch. Enhancenet: Single image super-resolution through automated texture synthesis. In *ICCV*, pages 4491–4500, 2017. 2

[14] Edgar Schönfeld, Bernt Schiele, and Anna Khoreva. A u-net based discriminator for generative adversarial networks. *arXiv preprint arXiv:2002.12655*, 2020. 1, 3

[15] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 1, 3

[16] Jae Woong Soh, Gu Yong Park, Junho Jo, and Nam Ik Cho. Natural and realistic single image super-resolution with explicit natural manifold discrimination. In *CVPR*, pages 8122–8131, 2019. 2

[17] Radu Timofte, Shuhang Gu, Jiqing Wu, and Luc Van Gool. Ntire 2018 challenge on single image super-resolution: Methods and results. In *CVPRW*, pages 852–863, 2018. 1, 2

[18] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *CVPR*, pages 606–615, 2018. 2

[19] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *ECCVW*, September 2018. 2, 3

[20] Tangxin Xie, Xin Yang, Yu Jia, Chen Zhu, and LI Xiaochuan. Adaptive densely connected single image super-resolution. In *ICCVW*, pages 3432–3440. IEEE, 2019. 2

[21] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, pages 6023–6032, 2019. 3

[22] Kai Zhang, Shuhang Gu, Radu Timofte, et al. Ntire 2020 challenge on perceptual extreme super-resolution: Methods and results. In *CVPRW*, 2020. 1

[23] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*, pages 586–595, 2018. 1, 2, 3

[24] Wenlong Zhang, Yihao Liu, Chao Dong, and Yu Qiao. Ranksrgan: Generative adversarial networks with ranker for image super-resolution. In *ICCV*, pages 3096–3105, 2019. 2, 3

[25] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *ECCV*, pages 286–301, 2018. 2