U-Net based discriminator for real-world super-resolution

Reference:
Ruiz Vargas Kevin Ian, Guerrero Pena Fidel Alejandro, Marrero Fernandez Pedro Diamel, Lanfranchi Leonardo, Tsang Ing Jyh, Ren Tsang Ing.- U-Net based discriminator for real-world super-resolution
International Conference on Tools with Artificial Intelligence : [proceedings] - ISSN 1082-3409 - Los alamitos, Ieee computer soc, (2022), p. 873-880
Full text (Publisher's DOI): https://doi.org/10.1109/ICTAI56018.2022.00134
To cite this reference: https://hdl.handle.net/10067/1973700151162165141

U-Net Based Discriminator for Real-World Super-Resolution

Kevin Ian Ruiz Vargas
Universidade Federal de Pernambuco
Centro de Informática
Recife, Brasil
kirv@cin.ufpe.br

Fidel Alejandro Guerrero Pena
Universidade Federal de Pernambuco
Centro de Informática
Recife, Brasil
LIVIA/ETS
Montreal, Canada
fagp@cin.ufpe.br

Pedro Diamel Marrero Fernandez
Universidade Federal de Pernambuco
Centro de Informática
CESAR
Recife, Brasil
pdmf@cin.ufpe.br

Leonardo Lanfranchi
Duke University
Pratt School of Engineering
Durham, USA
leonardoslanfranchi@gmail.com

Ing Jyh Tsang
University of Antwerp-IMEC
IDLab-Department of Computer Science
Antwerpen, Belgium
ingjyh.tsang@uantwerpen.be

Tsang Ing Ren
Universidade Federal de Pernambuco
Centro de Informática
Recife, Brasil
tir@cin.ufpe.br

***Abstract*—In recent years, single image super-resolution (SISR) deep learning techniques have achieved remarkable improvements in recovering a high-resolution (HR) image from an observed low-resolution (LR) input. Nevertheless, these proposed methods fail in many real-world scenarios since their models are usually trained using a pre-defined degradation process from HR ground truth images to LR ones. To address this issue, new architectures have been proposed focusing on adopting more complicated degradation models to emulate real-world degradation achieving prominent performance but still limited to certain kinds of inputs and dropping considerably in other cases.**

In this paper, we present a GAN structure for blind super-resolution tasks, applying a technique that has not been very commonly used in SISR proposals: a U-Net architecture as a discriminator of the GAN network. Adding this structural change will encourage the discriminator to focus more on semantic and structural changes between real and fake images and to attend less to domain-preserving perturbations. In addition, the loss function of the generator was modified by adding the LPIPS loss function for the perceptual loss and a per-pixel consistency regularization technique based on the CutMix data augmentation. Numerous novel solutions that have been proposed recently involve powerful deep learning techniques.

The proposed model was trained using the DF2K dataset employing a degradation framework for real-world images by estimating blur kernels and real noise distributions to obtain more realistic LR samples. Finally, we present a benchmark comparing our results with other methods in the state-of-the-art. The commonly-used evaluation metrics for image restoration PSNR, SSIM, and LPIPS were used for this evaluation.

***Index Terms*—Deep Learning, Degradation Modeling, Image Super-Resolution, Loss Functions, U-Net Discriminator.**

I. Introduction

In the wake of the increasing progress in deep learning techniques, several super-Resolution (SR) models have been developed, continuously achieving state-of-the-art performance on the SR image paradigms starting from the initial method based on Convolutional Neural Networks (CNN) to recent and promising SR approaches based on Generative Adversarial Networks (GAN) [1, 2, 3, 4, 5]. Consequently, the image super-resolution problem, strictly speaking to Single Image Super-Resolution (SISR), has gained much attention in the research community.

SISR aims to reconstruct a high-resolution image I_{SR} from a single low-resolution image I_{LR}. The relation between I_{LR} and the high-resolution image I_{HR} can differ depending on the situation. Several related works assume that I_{LR} is a bi-cubic down-sampled version of I_{HR}. Most uses fixed bi-cubic operations for down-sampling to construct training data pairs. Similarly, the input image down-sampled by the bi-cubic kernel in the test phase is applied to train a deep neural network [6]. However, to have better results in real-world super-resolution images, other degrading factors such as blur, decimation, or noise have been considered for practical applications.

Real complex degradation, which frequently is part of real-world images, is a complicated combination of different degradation processes that may occur in the imaging system of cameras, image editing, and channel transmission. Blind SR attempts to resolve this problem by enhancing low-resolution images with unknown degradation and, therefore, has attracted increasing attention due to its significance in promoting real-world applications. Many recent novel solutions have been proposed, especially with powerful deep learning techniques. Despite all the efforts of recent years, this problem remains a challenging research area [7].

This article focuses on improving the SR method by modeling the corrupted data via image processing artifacts to restore real-world low-resolution (LR) images. The degradation might contain these kinds of unpredictable noises or blurring. We evaluated the RealSR [6] proposal and employed its process for kernel estimation and noise injection to preserve the original domain attributes. In contrast, our main contribution is to train a deep learning model to generate high-resolution (HR) images with two crucial alterations. First, performing experiments adding an extra loss function associated with LPIPS [12], replacing the patch discriminator with a U-Net discriminator [9], and applying CutMix operation [10].

The article is organized as follows. Section II presents some related works in the context of the proposed approach. Section III describes the experiments and the obtained results. And Section IV presents the conclusions and some discussions.

II. Proposed Method

The proposed method can be described in three parts. First, to model the noise from real-world images, we applied the degradation process proposed by RealSR work [6]. This degradation process is injected into generating LR images to make them more realistic to real-life SR problems and situations.

Secondly, we trained the SR model based on the constructed data using an SRGAN architecture based on the BasicSR implementation [13]. In this case, the SR structure selected is based on the ESRGAN [8]. Furthermore, we performed several experiments adding a new LPIPS perceptual loss [12] to the loss functions. The basic idea is to improve the network's performance by regularizing the loss function with a perceptual metric.

Finally, we included a GAN module, adding a U-Net discriminator [11] to consider both the global and local context of an input image. Experimental results show that the proposed model significantly improves the results compared to the model presented by the Impressionism group at the *NTIRE 2020 Challenge* [14].

We based our proposal on the RealSR method of real image degradation modeling based on kernel estimation and noise injection [6]. The elemental degradation process from I_{HR} to I_{LR} can be expressed with the following equation:

$$y = f(x; s) \tag{1}$$

where x and y denote HR and LR images, respectively. f is the degradation function with a scale factor of s. Hence, the SR problem equals solving the inverse function f^{-1}. In the background of non-blind SR, f is generally assumed to be a bicubic downsampling:

$$y = x \downarrow_s^{bic} \tag{2}$$

or the combination of downsampling and a fixed Gaussian blur with kernel k_g, where $\otimes$ represents the convolutional operation in the image:

$$y = (x \otimes k_g) \downarrow_s \tag{3}$$

Subsequently, to estimate the degradation method more accurately, we applied the Realistic degradation framework for Super-Resolution (RealSR) proposed by [6] to estimate the kernel and noise from the real-world image. Once the estimated kernel and noise patch are obtained, it is possible to reconstruct a degradation pool, which is used to degrade clean HR images into blurry and noisy images, generating image pairs to train the SR models.

A. Generator

The base SR model is composed of a U-Net discriminator, as shown in Figure 1. The proposed model is trained on paired data $\{I_{LR}, I_{HR}\} \in \{X, Y\}$. The generator has an RRDB [15] structure, and the resolution of the output image by the generator is four times larger. Similarly to the ESRGAN model, we remove the batch normalization layers from SRGAN [16] to avoid unpleasant artifacts and replace the original residual block with the RRDB to boost performance. Formally, the generator returns ×4 super-resolved image I^{Gen} from an input image I^{I_n}:

$$I^{Gen} = G(I^{I_n}) \tag{4}$$

B. Discriminator

Many GAN-based SR methods have an encoder structure as a discriminator [8, 16, 17], which is simply a classifier that tries to distinguish real data from the data created by the generator. This module could use any network architecture appropriate to the type of data it's classifying. For instance, in RealSR, a Patch Discriminator [18, 19] is used instead of VGG-128 [20]. This is because of the limits in the size of the generated image to 128; moreover, it makes the discriminator pay more attention to global features and ignore local features. Although this approach has optimal performance, replacing this scheme with a U-Net discriminator [9] can accomplish better predictions because it considers both the global and local context and gives effective feedback to the generator.

C. Loss Functions

1) Generator: One part that is very relevant in this proposal is the employment of multiple losses applied to the training process. Concisely, the losses implemented for our generator are a combination of pixel loss, perceptual loss [18], and adversarial loss. The pixel loss L_1 uses L_1 distance. The perceptual loss L_{per} uses the inactive features of VGG-19 [20], which improve the visual effect of low-frequency features such as edges. And the adversarial loss L_{adv} has the task of enhancing the texture details of the generated image to obtain a more realistic image output. The total loss function is the weighted sum of all the previous losses:

$$L_{total} = \lambda_1 \cdot L_1 + \lambda_{per} \cdot L_{per} + \lambda_{adv} \cdot L_{adv} \tag{5}$$

Nonetheless, as several related works are mostly based on the GAN with the VGG perceptual loss, there were few considerations about the loss functions. Therefore, we have experimented with the learned perceptual similarity (LPIPS) loss functions for perceptual extreme SR. Instead of replacing the VGG perceptual loss with the LPIPS perceptual loss, we added to the loss function as shown in Equation 5. To this end, we use LPIPS [12] for the perceptual loss:

$$L_{lpips} = \sum_{k} \tau^k(\phi^k(I^{Gen}) - \phi^k(I^{GT})), \tag{6}$$

where ϕ is a feature extractor, τ transforms the embedding to scalar LPIPS score, and the score is computed and averaged from k layers. In addition, we use the discriminator's feature matching loss L_{fm} to alleviate the undesirable noise from the adversarial loss where D^l denotes the activation from the l^{th} layer of the discriminator D, and H is the Huber loss (smooth $L1$ loss) [11]:

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

As a result, the final generator loss function is defined as:

$$\begin{aligned} L_{total} = \lambda_1 \cdot L_1 + \lambda_{per} \cdot L_{per} + \lambda_{adv} \cdot L_{adv} \\ + \lambda_{fm} \cdot L_{fm} + \lambda_{lpips} \cdot L_{lpips} \end{aligned} \tag{8}$$

2) Discriminator: For the U-Net discriminator loss functions, besides the normal encoder structure, D_{enc} a decoder structure D_{dec} is also used to provide per-pixel feedback to the generator while preserving global context. Jo et. el. [11] have shown that the discriminator loss L_D can be computed at both the encoder head $L_{D_{enc}}$ and the decoder head $L_{D_{dec}}$. The formulation for the discriminator loss as hinge loss is:

$$\begin{aligned} L_{D_{enc}} &= -\mathbb{E}\left[\sum_{i,j} min\left(0, -1 + \left[D_{enc}(I^{GT})\right]_{i,j}\right)\right] \\ &= -\mathbb{E}\left[\sum_{i,j} min\left(0, -1 - \left[D_{enc}(I^{Gen})\right]_{i,j}\right)\right] \end{aligned} \tag{9}$$

$$\begin{aligned} L_{D_{dec}} &= -\mathbb{E}\left[\sum_{i,j} min\left(0, -1 + \left[D_{dec}(I^{GT})\right]_{i,j}\right)\right] \\ &= -\mathbb{E}\left[\sum_{i,j} min\left(0, -1 - \left[D_{dec}(I^{Gen})\right]_{i,j}\right)\right] \end{aligned} \tag{10}$$

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

$$L_{adv} = -\mathbb{E}\left[\sum_{i,j} \left[D_{enc}(I^{GT})\right]_{i,j} + \sum_{i,j} \left[D_{dec}(I^{Gen})\right]_{i,j}\right] \tag{11}$$

Additionally, as the loss functions contemplated in the U-Net discriminators [11], a consistency regularization [9] was applied to synthesize the training samples by using CutMix transformation [10] and minimizing the loss $L_{D_{cons}}$. Finally, the total discriminator loss is defined as:

$$L_D = L_{D_{enc}} + L_{D_{dec}} + L_{D_{cons}} \tag{12}$$

III. Experiments and Results

For the experiments, we use the DF2K dataset, which contains 3,450 images. This dataset is composed of the DIV2K dataset [21], proposed in NTIRE17 with 800 train and 100 validation images, and the Flikr2K dataset [22], with 2650 2K images for training [12].

The proposed model was implemented using PyTorch 1.7.1 and trained on a single NVIDIA GeForce RTX 2080 Ti (12G). We further trained the generator using the proposed loss functions for about 60K iterations with a mini-batch size of 16, which took about 16 hours. We empirically set $\lambda_1 = 1E^{-2}$, $\lambda_{per} = 1$, $\lambda_{adv} = 1E^{-3}$, $\lambda_{lpips} = 1E^{-3}$, and $\lambda_{fm} = 1$. The Adam optimizer was applied, and the learning rate was set to 0.0001 for training both the generator and the discriminator networks. The number of parameters for the generator and discriminator was 17M and 13M, respectively.

Three experiments were tested to evaluate the performance of this model. Initially, we trained the proposed model only with the LPIPS and Feature Matching losses and compared it to the original RealSR architecture. Following this, we trained the original RealSR model, only replacing its patch discriminator with the U-Net discriminator. And lastly, the model was evaluated with the combination of the new loss and the U-Net discriminator.

The image quality metric used to evaluate the results were PSNR, SSIM, and LPIPS, obtained by comparing the images with the ground truth. These results were also compared with the RealSR model that obtained first place in the *NTIRE 2020 perceptual extreme SR challenge*. The experiments demonstrated that the proposed new architecture significantly improved the results obtaining better LPIPS performance.

Table I show the overall results comparing the ESRGAN and RealSR model with an ablation study of the proposed model. For all three metrics, our proposal showed the best results. In the experiment, by applying only the new loss functions, the PSNR and SSIM metrics showed superior results. However, for the LPIPS metric, the RealSR model showed better results. Furthermore, by applying only the U-Net architecture, we observe that it has already improved the proposed model for the LPIPS metrics. Finally, applying both modifications, the results of the LPIPS metric further improve even though it does not exceed the PSNR and SSIM compared with just the loss function modification.

Table II compares the RealSR model with our model listing the number of images that obtained a better image metric. Since the DF2K test dataset contains 100 images, each evaluation was recorded and added to each model type if the image exceeded the value obtained with the other model. This

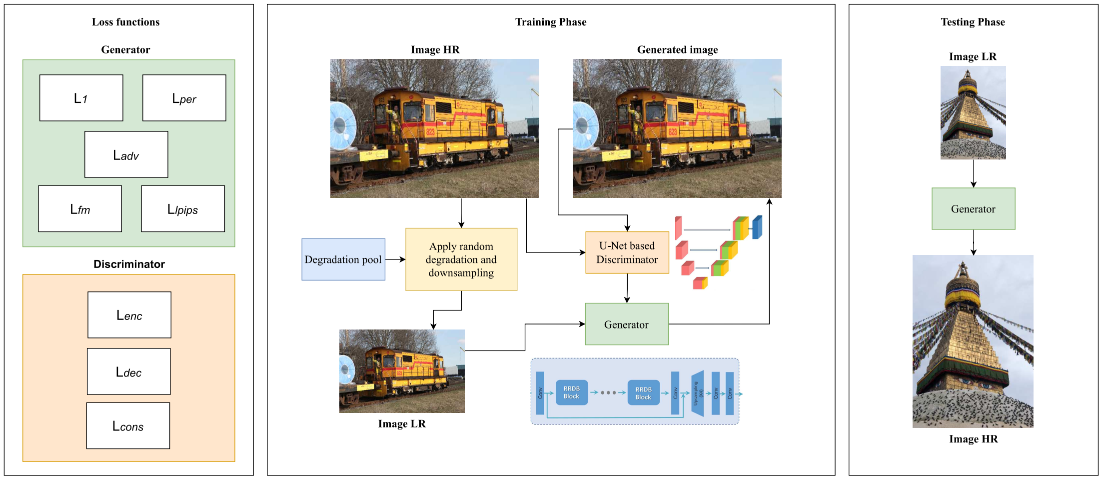


Fig. 1: Architecture of the proposed U-Net-based discriminator. The degradation pool provides diverse blur kernels and noise distributions for constructing realistic low-resolution images. During the training phase, the SR model is optimized to reconstruct high-resolution images.

comparison was made using just the RealSR since Table I shows that ESRGAN presents much lower results than RealSR and the proposed model.

TABLE I: Results of the model applied on DF2K dataset

Model	*PSNR* ↑	*SSIM* ↑	*LPIPS* ↓
ESRGAN	19.06	0.2423	0.755
RealSR	24.82	0.6619	0.227
Only loss functions (Ours)	**27.49**	**0.769**	0.322
Only U-Net discriminator (Ours)	26.14	0.719	0.222
Both modifications (Ours)	26.13	0.721	**0.219**

Table I. Quantitative results on DF2K dataset compared with ESRGAN and RealSR. Note that the ↑ and ↓ mean higher or lower value for the specific metric, the better the result.

TABLE II: Number of images with best metrics.

Experiment	Model	PSNR	SSIM	LPIPS
Only loss functions	RealSR	0	0	**91**
	Ours	**100**	**100**	9
Only U-Net discriminator	RealSR	0	2	40
	Ours	**100**	**98**	**60**
Both modifications	RealSR	0	0	33
	Ours	**100**	**100**	**67**

Table II. Evaluation of each of the 100 test images from DF2K. The number in the table shows how many images obtained higher values between the two SR models.

Table III compares six individual images using both RealSR and the proposed model. These images are shown in Figure 2 through Figure 7 so that the differences between both generated high-resolution images and ground-truth images can be observed. We used the referred number to identify the images from the test dataset.

1) Image 0804: We can observe in Figure 2 that the image generated by the proposed model is closer to the ground truth image since the RealSR image presents more artifacts on the part of the man's shirt and the left hand.

TABLE III: Metrics outcome from the example images

Experiment	Model	PSNR ↑	SSIM ↑	LPIPS ↓
Image 0804	RealSR	23,67	0,631	0,2
	Ours	**24,91**	**0,689**	**0,185**
Image 0812	RealSR	23,84	0,668	0,211
	Ours	**25,48**	**0,72**	**0,202**
Image 0813	RealSR	27,24	0,747	0,184
	Ours	**28,41**	**0,789**	**0,175**
Image 0814	RealSR	24,48	0,806	**0,217**
	Ours	**26,12**	**0,835**	0,234
Image 0836	RealSR	22,33	0,656	0,181
	Ours	**23,95**	**0,715**	**0,180**
Image 0853	RealSR	28,63	0,825	0,146
	Ours	**30,42**	**0,849**	**0,129**

Table III. Six individual images were compared using both RealSR and the proposed model. These images are shown in the following figures so that the differences between both generated high-resolution images and ground-truth images can be observed. We used the referred number to identify the images from the test dataset.

2) Image 0812: In Figure 3, the image obtained from our model shows a better result in detail and is more similar to the ground-truth image. The pillar details on the right present lines are more even and consistent with the GT image.

3) Image 0813: In Figure 4, the image generated by our proposal is a little closer in detail to the ground-truth image. For instance, the telephone on the wall has buttons more similar to the original image. Also, the faces present fewer artifacts than in the RealSR image.

4) Image 0814: In Figure 5, we can see that, although the RealSR image has a higher rating in the LPIPS metric, it still has many artifacts that are distorted from the original image. For example, the windows of the building on the left look very blurry and have strange lines. In our solution, it is much closer to the ground-truth image.

5) Image 0836: In Figure 6, we can observe that the yellow window lines have a more pleasant perception in our result. Also, the railing lines in the upper right part are finer and more detailed than the other generated image.

6) Image 0853: In Figure 7, we can see that the image obtained by the RealSR model presents some artifacts and lines that were hallucinated in an over-detailed way in the bird's feathers. Our image is more similar to the ground-truth image.

IV. Conclusion

In this paper, we present a novel GAN architecture implementing a U-Net structure as a discriminator and adding an LPIPS loss function to the generator to train a super-resolution model oriented to tackle blind SR issues. Furthermore, adding more techniques to increase the effectiveness of our model by including CutMix data augmentation and kernel estimation, and noise injection during the generation of the samples for training our model.

Based on these modifications, we trained our model using the DF2K dataset with each modification in the generator and the discriminator separately and finally all together. Thus, it was possible to compare our proposal with other state-of-the-art generative methods by upscaling the LR images in the test dataset used on the NTIRE 2020 perceptual extreme SR challenge. Finally, we can observe that our trained SR architecture outperforms the two SR models based on the most relevant image metrics PSNR, SSIM, and LPIPS.

Thus, we can observe that the images generated by our work show a reduction of artifacts in the generation of images and gets more approximate to the Ground-Truth image.

One proposal for future work is to increase the SR scale of this model. In this experiment, images [x4] times larger than the input image were generated. Working on the architecture or adding modules that allow higher resolution, e.g. [x8, x16], is a very important approach to be analyzed for future works and projects.

References

[1] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in ECCV, 2014, pp.184–199.

[2] C. Dong, C. C. Loy, and X. Tang, "Accelerating the superresolution convolutional neural network," in ECCV, 2016, pp. 391–407.

[3] W. Shi, J. Caballero, F. Huszar, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in CVPR, 2016, pp. 1874–1883.

[4] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang et al., "Photorealistic single image super-resolution using a generative adversarial network," in CVPR, 2017, pp. 4681–4690.

[5] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image superresolution using very deep residual channel attention networks," in ECCV, 2018, pp. 286–301.

[6] Xiaozhong Ji, Yun Cao, Ying Tai, Chengjie Wang, Jilin Li, and Feiyue Huang. Real-world super-resolution via kernel estimation and noise injection. In CVPRW, 2020.

[7] Anran Liu, Yihao Liu, Jinjin Gu, Yu Qiaoand, and Chao Dong. Blind image super-resolution: A survey and beyond. arXiv:2107.03055, 2021.

[8] 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, 2018.

[9] Edgar Schonfeld, Bernt Schiele, and Anna Khoreva. A u-net based discriminator for generative adversarial networks. arXiv preprint arXiv:2002.12655, 2020.

[10] 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 International Conference on Computer Vision (ICCV), 2019.

[11] Younghyun Jo, Sejong Yang, Seon Joo Kim, Yonsei University, Facebook, Investigating Loss Functions for Extreme Super-Resolution, 2020.

[12] 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.

[13] Xintao Wang, Ke Yu, Kelvin C.K. Chan, Chao Dong and Chen Change Loy. BasicSR: Open Source Image and Video Restoration Toolbox. https://github.com/xinntao/BasicSR, 2018.

[14] Andreas Lugmayr, Martin Danelljan, Radu Timofte, et al.Ntire 2020 challenge on real-world image super-resolution: Methods and results. CVPR Workshops, 2020.

[15] 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 Proceedings of the European Conference on Computer Vision (ECCV), pages 0–0, 2018.

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

[17] 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.

[18] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1125–1134, 2017.

[19] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycleconsistent adversarial networks. In Proceedings of the IEEE international conference on computer vision, pages 2223–2232, 2017.

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

[21] Radu Timofte, Eirikur Agustsson, Luc Van Gool, MingHsuan Yang, and Lei Zhang. Ntire 2017 challenge on single image super-resolution: Methods and results. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pages 114–125, 2017.

[22] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pages 126–135, 2017.

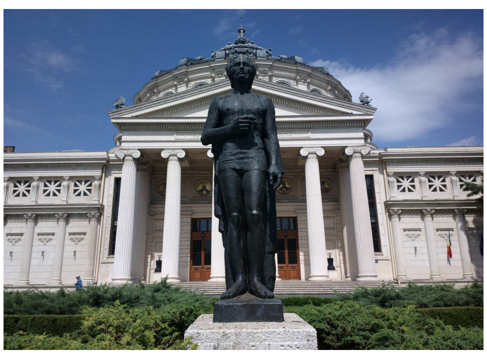
(a) Image Ground-Truth

(b) Ground-Truth Patch Image

(c) RealSR generated image

(d) Model trained with both modifications (Ours)

Fig. 2: Image 0804 from DF2K test dataset



(a) Image Ground-Truth

(b) Ground-Truth Patch Image

(c) RealSR generated image

(d) Model trained with both modifications (Ours)

Fig. 3: Image 0812 from DF2K test dataset

(a) Image Ground-Truth

(b) Ground-Truth Patch Image

(c) RealSR generated image

(d) Model trained with both modifications (Ours)

Fig. 4: Image 0813 from DF2K test dataset

(a) Image Ground-Truth

(b) Ground-Truth Patch Image

(c) RealSR generated image

(d) Model trained with both modifications (Ours)

Fig. 5: Image 0814 from DF2K test dataset

(a) Image Ground-Truth

(b) Ground-Truth Patch Image

(c) RealSR generated image

(d) Model trained with both modifications (Ours)

Fig. 6: Image 0836 from DF2K test dataset

(a) Image Ground-Truth

(b) Ground-Truth Patch Image

(c) RealSR generated image

(d) Model trained with both modifications (Ours)

Fig. 7: Image 0853 from DF2K test dataset