

Multi-scale deep neural networks for real image super-resolution

Shangqi Gao
Fudan University
Shanghai, China

18110980005@fudan.edu.cn

Xiahai Zhuang
Fudan University
Shanghai, China

zxh@fudan.edu.cn

Abstract

Single image super-resolution (SR) is extremely difficult if the upscaling factors of image pairs are unknown and different from each other, which is common in real image SR. To tackle the difficulty, we develop two multi-scale deep neural networks (MsDNN) in this work. Firstly, due to the high computation complexity in high-resolution spaces, we process an input image mainly in two different downscaling spaces, which could greatly lower the usage of GPU memory. Then, to reconstruct the details of an image, we design a multi-scale residual network (MsRN) in the downscaling spaces based on the residual blocks. Besides, we propose a multi-scale dense network based on the dense blocks to compare with MsRN. Finally, our empirical experiments show the robustness of MsDNN for image SR when the upscaling factor is unknown. According to the preliminary results of NTIRE 2019 image SR challenge, our team (ZXHresearch@fudan) ranks 21-st among all participants. The implementation of MsDNN is released at: <https://github.com/shangqigao/gsq-image-SR>

1. Introduction

Single image super-resolution (SR) aims at estimating the mapping from low-resolution (LR) to high-resolution (HR) spaces [4, 1, 5]. Interpolation is one of the most common methods in super-resolving an image without referring the priors of its ground truth. However, several works showed that it would decimate the details of an image [27, 6, 26]. Recently, the learning-based methods have been widely applied in image SR thanks to its robust ability of recovering details. The learning-based methods aim at estimating the end-to-end mapping from LR to HR image pairs by adopting some well-known deep neural networks (DNN). The methods could be classified into two categories according to the manners of processing LR images, i.e., the methods in LR and HR spaces.

The learning-based methods in LR space are generally proposed to reconstruct the SR image of an input under the

given upscaling factor [17, 20, 7, 14, 15, 28, 3]. For a given upscaling factor, the results of NTIRE 2017 image SR challenge showed that the learning-based methods perform robust when the input images are noise free [22]. In reality, natural images are inevitably accompanied with unknown noise, i.e., the downscaling operators from HR to LR images are different. Therefore, the noisy image SR is extremely challenging, which was verified in the NTIRE 2018 image SR challenge [23].

The learning-based methods in HR space are also developed to approximate the mapping from LR to HR image pairs using networks [6, 13, 12], but their inputs have the same size with the corresponding outputs. If the upscaling factor is given, the input will be the interpolation of an LR image. In reality, both close-shot and long-shot could be existed in an image dataset. The upscaling factors of close-shot and long-shot should be different if we want to obtain the images with high quality. Therefore, we expect to develop a method which has generalization capability to different upscaling factors.

The NTIRE 2019 image SR challenge¹ provided a dataset in which the sizes of each LR and HR image pair are the same, i.e., the upscaling factors of LR images are unknown and possibly different. The training dataset is small due to the fact that there are only 60 image pairs provided. Besides, the dataset is composed of the images obtained by two different cameras. Overall, the downscaling operators from HR to LR images are different. Therefore, one difficulty of this challenge is to develop a method which performs robust in super-resolving the LR images obtained by different downscaling operators. Another challenge is to explore an approach with low computation complexity since the LR images are very large in the testing phase. To tackle the former, we devote to designing a deep neural network (DNN) thanks to its robust ability in reconstructing the details of an image. To solve the latter, we propose a multi-scale model to downscale the input images.

In this work, we develop two multi-scale deep neural networks to solve the problems from NTIRE 2019 image

¹<http://www.vision.ee.ethz.ch/ntire19/>

SR challenge. For the learning-based methods developed in HR space, the number of filters in each convolutional layer should be small due to the large sizes of testing images. To increase the features maps of each layer, we downscale an input to obtain its downscaling, and process the input and two downscaling images in HR space (DS0), downscaling $\times 2$ (DS2) and downscaling $\times 4$ (DS4) spaces, respectively. Overall, we develop two networks in downscaling spaces to recover the details of input images, i.e., multi-scale residual networks (MsRN) and multi-scale dense network (MsDN). MsRN is mainly developed in DS2 and DS4 while MsDN is exploited in each space. Our experiments show that the training and testing efficiency in downscaling spaces are much higher due to the fact that the size of a downscaling image is greatly decreased.

The rest of this work is organized as follows. Section 2 shows the related works about the residual and dense networks for image SR. We develop two multi-scale networks in Section 3. Section 4 gives the details of training strategies. The description of our experiments is presented in Section 5. We discuss the proposed method in Section 6 and conclude this work in Section 7.

2. Related works

The aim of single image SR is to estimate the mapping from LR to HR image pairs. Recently, DNN has been widely applied in image SR due to its ability of simulating complex mappings. Dong *et al.* [6] first proposed to approximate the mapping from LR to HR image pairs using a three layers convolutional neural network. Since then, other architectures, such as RNN [21, 8, 16], ResNet [10], and GAN [9], have been applied in image SR.

ResNet was first proposed by He *et al.* [10] for the task of classification. Ledig *et al.* [15] successfully introduce it to image SR and developed an approach referred as SR-ResNet. SRResNet preserved the batch normalization from original residual blocks, which was showed to consume amounts of memory in computation and restrict the range flexibility from networks for image deblurring [18] and SR [17]. Therefore, Lim *et al.* [17] proposed a new residual block by removing the batch normalization and developed an enhanced deep residual network referred as EDSR. The comparison of ResNet, SRResNet and EDSR is showed in Figure 1.

The skip connections in ResNet are critical since they constraint a residual block to learn the residual between its input and output [10]. To explore the advantages of skip connection, DenseNet proposed to link all layers in the networks to efficiently train a very deep networks [11]. To adopt the idea of DenseNet for image SR, Tong *et al.* [25] developed a method, referred as SRDenseNet, to estimate the mapping from LR to HR image pairs. Motivated by the idea of EDSR and SRDenseNet, we will design two multi-

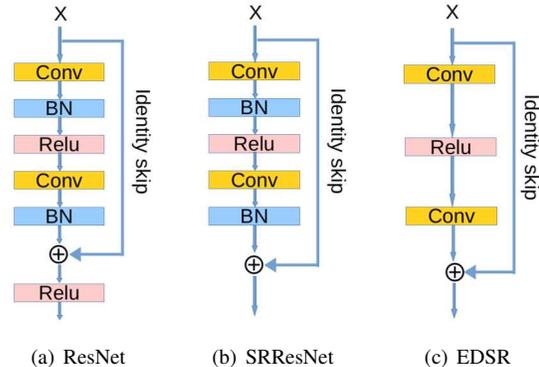


Figure 1. The comparison of residual blocks in ResNet [10], SR-ResNet [15], and EDSR [17].

scale deep neural networks for single image SR with unknown upscaling factors and downscaling operators in Section 3.

3. Multi-scale deep neural networks

In reality, images are often obtained by different kinds of cameras. Besides, to reconstruct the SR image with high quality, the upscaling factors of close-shot and long-shot could be different. Overall, the challenge of realistic image SR is that we need to develop a method which has generalization capacity to both the upscaling factors and downscaling operators.

3.1. Multi-scale residual networks

Deep residual networks show a big potentiality in approximating the end-to-end mapping from LR to HR spaces. The results of NTIRE 2018 image SR challenge [23] showed that the ResNet-based methods perform robust when LR images are obtained by unknown downscaling operators. However, for NTIRE 2019 image SR challenge, it is extremely expensive to develop a deep residual networks in LR space since the sizes of LR and HR image pair are the same. To be specific, the HR image from this challenge is firstly downscaled by an unknown operator, then the downscaling image is interpolated to obtain a LR image with the same size as the HR image. Therefore, the number of filters in each layer will be limited by the single GPU memory due to the high computation complexity in LR space.

The previous works have verified that a wide convolutional layer could improve the performance of deep residual networks [17, 28]. Inspired by this, we propose a multi-scale residual network (MsRN) to get rid of the limitation of GPU memory and widen the convolutional layers of this network, as shown in Figure 2. The architecture of MsRN in DS2 or DS4 space is a deep residual network, and in DS0 space is composed of a convolutional layer. Moreover, the *Downscale* in Figure 2 denotes a convolutional layer

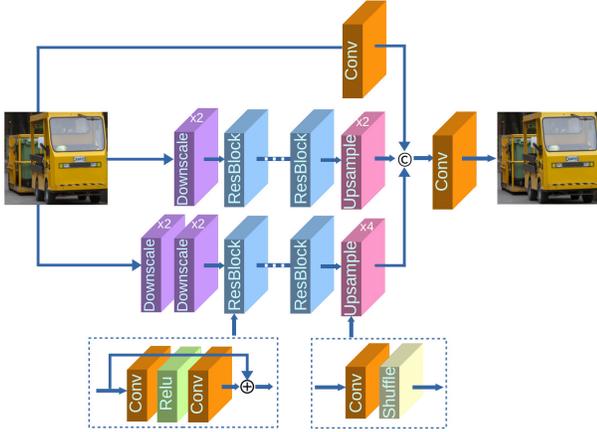


Figure 2. The architecture of the proposed multi-scale residual networks.

with the strides equal to 2. There are several advantages about this architecture. First, the training and testing of MsRN are cheap due to the fact that the main computation is implemented in the two downscaling spaces. Besides, the maximal number of filters in DS2 could be four times of it in DS0, which is important to improve the performance of MsRN. Finally, the connection in DS0 could reduce the loss of information from an LR image in the process of downscaling.

3.2. Multi-scale dense networks

As shown in ResNet, the output of an residual block is the sum of its main and identity connections. Although the structure performs robust in learning the residual between its input and output, it may leads to the loss of useful information from the input. To solve the problem, we have designed a skip connection in DS0 space for MsRN, but it can not prevent the loss of information in DS2 and DS4 spaces which is induced by the depth of networks.

The DenseNet is proposed to sufficiently take the advantages of skip connections. Motivated by this, we develop a multi-scale dense network (MsDN) for image SR, as shown in Figure 3. The networks in DS0, DS2, and DS4 are mainly composed of the *DenBody*, of which the detailed architecture is presented in Figure 4. As shown in Figure 4, the first three blocks are concatenated with the final three blocks to reduce the loss of information. Compared with MsRN, the depth of MsDN could be much smaller due to the amount of skip connections. Therefore, the training and testing of MsDN are cheaper than MsRN although there is a dense network in DS0.

4. Training and testing strategies

As well as we known, the training of DNN on a small dataset is very difficult due to the overfitting. One method to

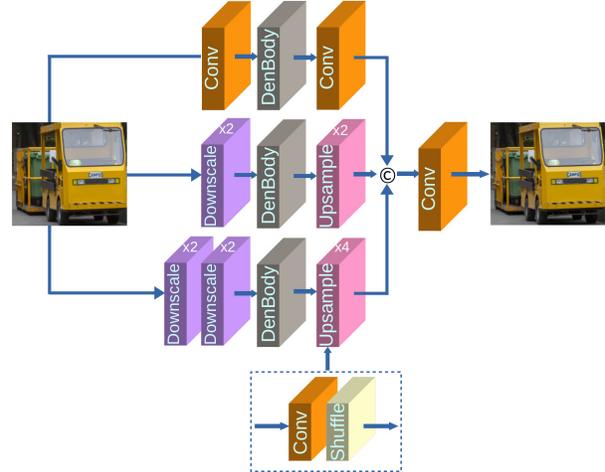


Figure 3. The architecture of the proposed multi-scale dense networks.

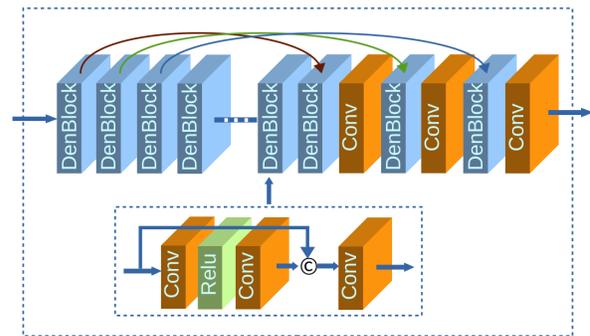


Figure 4. The detailed structure of *DenBody* in the multi-scale dense networks.

tackle the difficulty is to use regularization, such as weight decay, but it would increase the computation complexity. Another approach to solve the problem is to augment a small dataset by flipping and rotations, which is widely utilized in computer vision.

The NTIRE 2019 image SR challenge provided a training dataset with only 60 image pairs, which is not enough for the training of deep networks. Therefore, in the training stage, we use the strategy of augmentation to prevent our models from overfitting. To be specific, let f_{lr} and f_{ud} denote the left-right and up-down flipping, respectively, and r denotes the 90 degree rotations. Then every training image would be mapped by the three mappings one by one with a possibility of one in two. This strategy has showed to be useful in increasing the diversity of a training dataset.

For image SR, the SR image of an input could be enhanced by averaging the SR images derived from the transformations of the input [24]. Therefore, in the testing phase, we utilize the strategy to enhance the performance of image SR. Concretely, Let us denote an input LR image as I_{LR} ,

Table 1. The settings of multi-scale deep neural networks.

Options	Baseline-R	MsRN	Baseline-D	MsDN
#Total blocks	(32, -, -)	(0, 32, 32)	(12, -, -)	(12, 12, 12)
#Filters	(16, -, -)	(3, 96, 96)	(16, -, -)	(12, 48, 96)
#Downscale	-	{96, (48, 96)}	-	{48, (48, 96)}
#Additions	32	64	0	0
#Concatenations	0	1	15	46
#Parameters	9.4K	113.3K	5.8K	91.9K

and MsDNN as F , then the SR image of I_{LR} will be obtained by averaging the elements in the following set,

$$\{F(I_0), F(I_1), F(I_2), F(I_3)\} \\ \cup \{F(r(I_0)), F(r(I_1)), F(r(I_2)), F(r(I_3))\} \quad (1)$$

, where $I_0 = I_{LR}$, $I_1 = f_{lr}(I_{LR})$, $I_2 = f_{ud}(I_{LR})$, and $I_3 = f_{ud}(f_{lr}(I_{LR}))$.

5. Experiments

5.1. Details of implementation

The dataset from NTIRE 2019 image SR challenge is composed of 60 training images, 20 validation images, and 20 testing images. The LR images of this dataset have the same size with the corresponding HR images, i.e., a LR image is the interpolation of an unknown downscaling HR image. We will test the performance of our method on the validation dataset since the ground truth of testing dataset is not public.

There is a data pre-processing before training. To be specific, we randomly crop 64×64 pixels patches from LR images and augment them via the training strategies in Section 4. Then these patches are shuffled and utilized to assemble the batches of size 16. Finally, the batches are used to train all models. Unless otherwise stated, the kernel size of convolutions in MsDNN is 3×3 . To demonstrate the advantages of MsDNN, we set two baselines for MsRN and MsDN. Concretely, we design a deep residual network in DS0 space as the baseline of MsRN, which is referred as Baseline-R. In the same way, a deep dense network is developed in DS0 to be the baseline of MsDN, which is referred as Baseline-D. The detailed settings of baselines, MsRN and MsDN are showed in Table 1. Here, #Total blocks denotes the number of blocks in DS0, DS2, and DS4 spaces, respectively. #Filters denotes the number of filters of a convolutional layer in DS0, DS2, and DS4 spaces, respectively. *Downscale* in MsDNN is composed of a convolutional layer with the strides equal to 2. Therefore, the first element of #Downscale in Table 1 denotes the number of filters of *Downscale* in DS2 space, and the second element denotes the number of filters of two *Downscale* in DS4 space. #Additions and #Concatenations denote the number of addition and concatenation existed in MsDNN, i.e., the amount of \oplus and \odot , respectively.

For training, each model is trained by optimizing the L_2 or L_1 norm. All methods are trained by the ADAM optimizer, and settings of parameters are $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 1 \times 10^{-8}$. The training of each model is up to 100 million updates. The initial learning rate is 1×10^{-4} , it decreases to 20 percent every 10 million updates when the updates are greater than 60 million. Finally, We implement our networks with TensorFlow and train our models on a device with 40 Intel Xeon 2.20 Ghz CPUs and 4 DTX 1080 Ti GPUs. The training of MsRN and MsDN cost about 30 and 20 hours on a single GPU, respectively.

During testing, we utilize the testing strategy of enhancement in Section 4 to improve the performance of models on testing dataset. To evaluate the performance of our method, the standard Peak Signal To Noise Ratio (PSNR) and the Structural Similarity (SSIM) index are used. We could directly compute the PSNR and SSIM of a LR- and SR image pair since their sizes are the same. Finally, we show the quantitative comparisons of our models via the standard criteria.

5.2. Comparisons of L_1 and L_2 losses

The loss functions is critical to the training of networks due to the fact that each parameter of the networks is updated by optimizing the given losses. To compare the performance of MsDNN under different loss functions, we have trained MsDNN by optimizing L_1 and L_2 norm. Concretely, we first obtain two models, i.e., MsRN-1 and MsRN-2, by training MsRN with L_1 and L_2 . Then compute the PSNR and SSIM of two models in the training phase, to plot the convergent curves of MsRN-1 and MsRN-2 on a cropped validation dataset. We repeat the above process for MsDN to obtain two models, i.e., MsDN-1 and MsDN-2, and plot the convergent curves of these two models.

Figure 6 shows the convergent curves of MsRN-1 and MsRN-2, from which we can see that the convergent speed of MsRN-1 is faster, and the average PSNR and SSIM is higher. Figure 7 shows the convergent curves of MsDN-1 and MsDN-2. The performance of MsDN-1 is more robust since its average PSNR and SSIM are higher. Therefore, according to the convergent curves of MsRN and MsDN, we conclude that L_1 norm minimization could be better in the training of MsDNN.

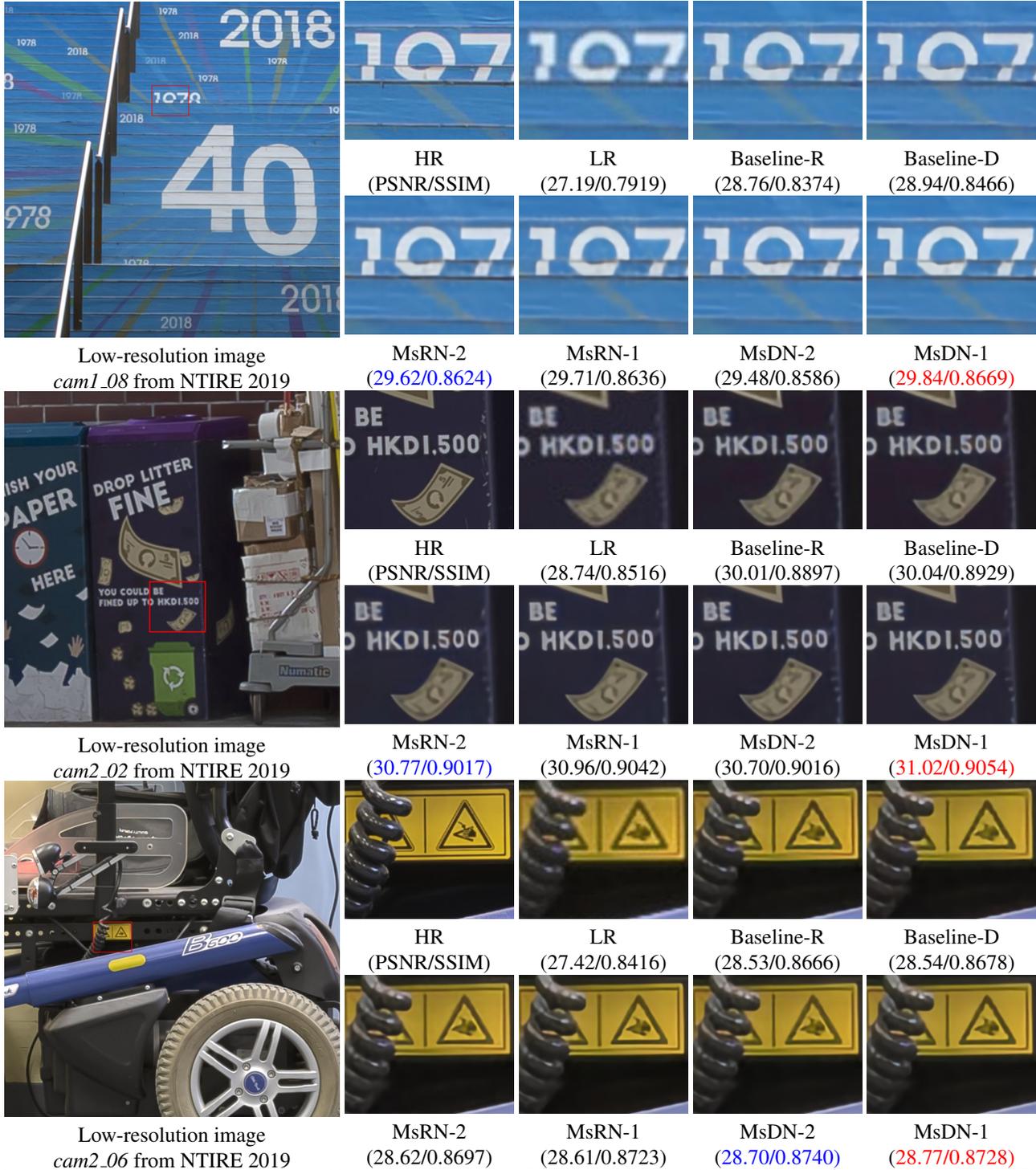
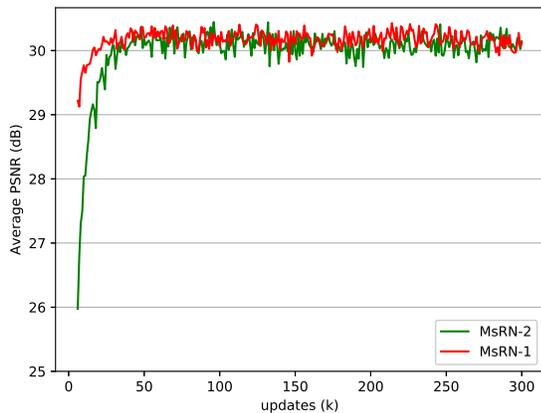


Figure 5. The visualization of SR images which are reconstructed by MsRN and MsDN on the three images from validation dataset. The blue value denotes the maximum among the values obtained by L_2 norm based models. The red value denotes the maximum among the values obtained by L_1 norm based models.

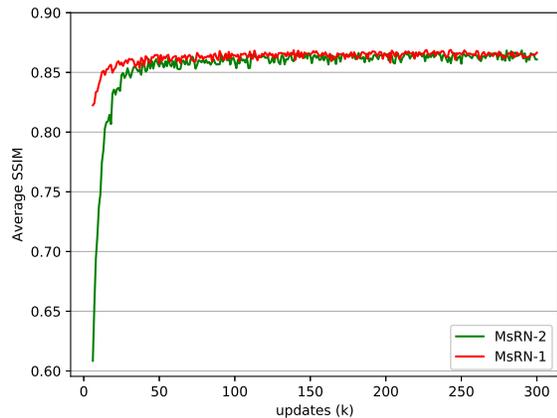
5.3. Comparisons of MsRN and MsDN

To solve the image SR challenge from NTIRE 2019, we have developed two multi-scale networks, i.e., MsRN and

MsDN. The architecture of MsRN is composed of the residual blocks while the development of MsDN is based on the dense blocks. To compare MsRN with MsDN quanti-

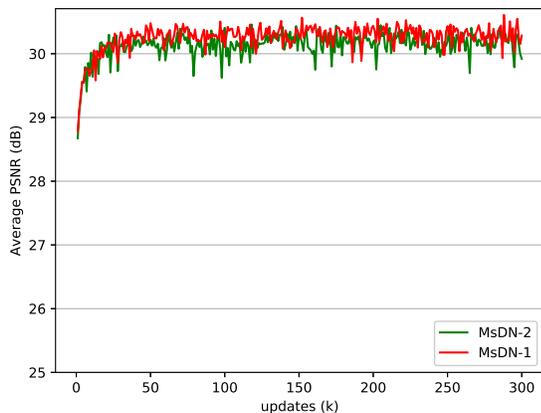


(a) PSNR of MsRN

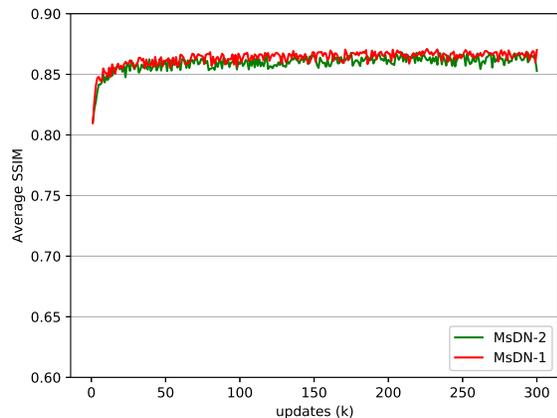


(b) SSIM of MsRN

Figure 6. The convergent curves of average PSNR and SSIM of MsRN on a cropped validation dataset in the training phase.



(a) PSNR of MsDN



(b) SSIM of MsDN

Figure 7. The convergent curves of average PSNR and SSIM of MsDN on a cropped validation dataset in the training phase.

tatively, we evaluate the performance of them on the validation dataset. As stated in Section 5.2, we obtain four models by optimizing L_1 and L_2 losses, i.e., MsRN-2, MsDN-2, MsRN-1, and MsDN-1. Besides, the baselines of MsRN and MsDN are trained by L_1 norm to obtain two models, i.e., Baseline-R and Baseline-D. Finally, we compute the average PSNR and SSIM of each model on the validation dataset via the strategy in Section 4.

The quantitative evaluation of the models is showed in Table 2, which shows that both PSNR and SSIM of MsDN are higher than them of MsRN. Figure 5 shows the visualization of SR images obtained by the models, which demonstrates that PSNR and SSIM of MsDN are comparable with them of MsDN. The average runtime of MsRN and MsDN on validation dataset are about 21 and 20 seconds, respec-

tively. Therefore, we conclude that MsDNN is an effective method to solve the SR challenge from NTIRE 2019 according to our baselines, and the performance of MsRN and MsDN are comparable for single image SR.

6. Discussion

Utilizing the empirical experiments in Section 5, we have showed the robustness of MsDNN for image SR with unknown upscaling factors. To test the performance of MsDNN for image SR with different downscaling operators, i.e., LR images are obtained by different cameras, we divide the validation dataset into two parts, i.e., *Validation-1* and *Validation-2*. the former is composed of the LR- and HR image pairs obtained by camera-1, the latter consists of the LR- and HR image pairs obtained by camera-2.

Table 2. The average PSNR and SSIM of compared methods on the validation dataset. The blue value denotes the maximum among the values obtained by L_2 norm based models. The red value denotes the maximum among the values obtained by L_1 norm based models.

Method	Interpolation	Baseline-R	MsRN		Baseline-D	MsDN	
Model			MsRN-2	MsRN-1		MsDN-2	MsDN-1
PSNR	27.78	29.09	29.42	29.49	29.02	29.47	29.51
SSIM	0.8163	0.8557	0.8646	0.8664	0.8528	0.8651	0.8671

Then we compute the average PSNR and SSIM of every model on *Validation-1* and *Validation-2*, as shown in Table 3. The quantitative evaluation shows that the performance of MsDNN on *Validation-2* is better than it on *Validation-1*. Besides, we test the performance of models on Set5 [2], Set14 [19] and DIV2K [22]. To be specific, each LR image is the bicubic interpolation ($\times 3$) of corresponding down-sampled HR image. Table 4 shows that the performance of MsDNN on the datasets is even worse than it of bicubic, of which the reason could be that there does not contain bicubic downscaling ($\times 3$) operator from HR to LR image pairs in the training dataset of NTIRE 2019. Therefore, for the LR images obtained by different downscaling operators, the exploration of improving the generalization capability of DNN is still a difficult but interesting task.

7. Conclusion

The NTIRE 2019 image SR challenge is difficult since the upscaling factor of each LR image is unknown and the downscaling operators from HR to LR image pairs are different. Due to the fact that the sizes of an LR image and its ground truth are the same, processing the input images directly in LR space is extremely expensive. Besides, in the testing phase, the maximal number of filters of a convolutional layer is limited by the single GPU memory. For example, the number of filters of each convolutional layer must be less than 32 for a GPU with 11 GB memory in our experiments. To tackle these difficulties, we have proposed a multi-scale deep neural network to solve the SR challenge from NTIRE 2019. First, we have processed the input images mainly in the two downscaling spaces to simplify the computation complexity in the training and testing phases. Second, we have developed two deep neural networks in the downscaling spaces to super-resolve the LR images with unknown upscaling factors. Finally, we have tested the performance of MsDNN in Section 5 to show its robustness for image SR with unknown upscaling factors.

References

[1] M. Bevilacqua, A. Roumy, C. Guillemot, and M. L. A. Morel. Low-complexity single-image super-resolution based on non-negative neighbor embedding. In *Proceedings of the British Machine Vision Conference*, pages 1–10, 2012. 1

[2] Marco Bevilacqua, Aline Roumy, Christine Guillemot, and Marie-Line Alberi Morel. Low-complexity single-image

super-resolution based on nonnegative neighbor embedding. In *British Machine Vision Conference*, 2012. 7

[3] Jose Caballero, Christian Ledig, Andrew Aitken, Alejandro Acosta, Johannes Totz, Zehan Wang, and Wenzhe Shi. Real-time video super-resolution with spatio-temporal networks and motion compensation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2848–2857, 2017. 1

[4] H. Chang, D. Y. Yeung, and Y. Xiong. Super-resolution through neighbor embedding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 275–282, 2004. 1

[5] D. Dai, R. Timofte, and L. Van Gool. Jointly optimized regressors for image super-resolution. *Eurographics*, 34(2):95–104, 2015. 1

[6] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(2):295–307, 2016. 1, 2

[7] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *Proceedings of European Conference on Computer Vision (ECCV)*, pages 391–407, 2016. 1

[8] David Eigen, Jason Rolfe, Robert Fergus, and Yann Lecun. Understanding deep architectures using a recursive convolutional network. In *International Conference on Learning Representations (ICLR)*, 2014. 2

[9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2672–2680, 2014. 2

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 2

[11] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2261–2269, 2017. 2

[12] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1646–1654, 2016. 1

[13] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1637–1645, 2016. 1

[14] Weisheng Lai, Jiabin Huang, Narendra Ahuja, and Mingshan Yang. Deep laplacian pyramid networks for fast and

Table 3. The average PSNR and SSIM of compared methods on the datasets obtained by two different cameras. The blue value denotes the maximum among the scores of four models tested on Validation-1. The red value denotes the maximum among the scores of four models tested on Validation-2.

Dataset	Validation-1				Validation-2			
	MsRN-2	MsRN-1	MsDN-2	MsDN-1	MsRN-2	MsRN-1	MsDN-2	MsDN-1
PSNR	30.08	30.16	30.11	30.27	28.77	28.82	28.83	28.76
SSIM	0.8638	0.8661	0.8639	0.8677	0.8654	0.8666	0.8662	0.8665

Table 4. The average PSNR and SSIM of several models on Set5, Set14 and DIV2K under bicubic downscaling ($\times 3$) assumption. The red value indicates the best performance.

Dataset	Criteria	Bicubic	Baseline-R	MsRN		Baseline-D	MsDN	
				MsRN-2	MsRN-1		MsDN-2	MsDN-1
Set5	PSNR	28.63	27.38	26.36	26.56	27.12	26.83	26.22
	SSIM	0.8378	0.8242	0.8059	0.8058	0.8131	0.8124	0.8025
Set14	PSNR	25.89	25.20	24.54	24.44	24.96	24.60	24.43
	SSIM	0.7401	0.7315	0.7232	0.7216	0.7166	0.7190	0.7172
DIV2K	PSNR	29.19	28.12	27.17	27.05	27.72	27.33	27.11
	SSIM	0.8359	0.8250	0.8136	0.8098	0.8117	0.8130	0.8108

accurate super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5835–5843, 2017. 1

- [15] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 105–114, 2017. 1, 2
- [16] Ming Liang and Xiaolin Hu. Recurrent convolutional neural network for object recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3367–3375, 2015. 2
- [17] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 1132–1140, 2017. 1, 2
- [18] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 257–265, 2017. 2
- [19] Zeyde R, Elad M, and Protter M. On single image scale-up using sparse-representations. In *International Conference on Curves and Surfaces Springer*, 2010. 7
- [20] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1874–1883, 2016. 1
- [21] Richard Socher, Brody Huval, Bharath Bhat, Christopher D. Manning, and Andrew Y. Ng. Convolutional-recursive deep learning for 3D object classification. In *The Conference and Workshop on Neural Information Processing Systems*, 2012. 2
- [22] Radu Timofte, Eirikur Agustsson, Luc Van Gool, and Ming-Hsuan Yang. 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 1110–1121, 2017. 1, 7
- [23] Radu Timofte, Shuhang Gu, Jiqing Wu, and Luc Van Gool. NTIRE 2018 challenge on single image super-resolution: Methods and results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 852–863, 2018. 1, 2
- [24] Radu Timofte, Rasmus Rothe, and Luc Van Gool. Seven ways to improve example-based single image super resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1865–1873, 2016. 3
- [25] Tong Tong, Gen Li, Xiejie Liu, and Qinquan Gao. Image super-resolution using dense skip connections. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 4809–4817, 2017. 2
- [26] C. Y. Yang, J. B. Huang, and M. H. Yang. Exploiting self-similarities for single frame super-resolution. In *Proceedings of European Conference on Computer Vision (ECCV)*, pages 372–386, 2016. 1
- [27] Jianchao Yang, John Wright, Thomas S. Huang, and Yi Ma. Image super-resolution via sparse representation. *Trans. Img. Proc.*, 19(11):2861–2873, 2010. 1
- [28] Jiahui Yu, Yuchen Fan, Jianchao Yang, Ning Xu, Zhaowen Wang, Xinchao Wang, and Thomas Huang. Wide activation for efficient and accurate image super-resolution, 2018. arXiv:1808.08718v2. 1, 2