A Dual Path Deep Network for Single Image Super-Resolution Reconstruction

Fateme S. Mirshahi, Parvaneh Saeedi
School of Engineering Science
Simon Fraser University, Burnaby, BC, Canada
Email: {fmirshah,psaeedi}@sfu.ca

***Abstract*—Super-resolution reconstruction based on deep learning has come a long way since the first proposed method in 2015. Numerous methods have been developed for this task using deep learning approaches. Among these methods, residual deep learning algorithms have shown better performance. Although all early proposed deep learning based super-resolution frameworks used bicubic upsampled versions of low resolution images as the main input, most of the current ones use the low resolution images directly by adding up-sampling layers to their networks. In this work, we propose a new method by using both low resolution and bicubic upsampled images as the inputs to our network. The final results confirm that decreasing the depth of the network in lower resolution space and adding the bicubic path lead to almost similar results to those of the deeper networks in terms of PSNR and SSIM, yet making the network computationally inexpensive and more efficient.**

I. Introduction

High resolution (HR) images are exceedingly on demand in various fields such as surveillance video, remote sensing, medical imaging, and video standard conversion. There are different solutions for increasing the spatial resolution including: reduction of the pixel size, increase of the chip size, as well as signal processing techniques.

The idea of super-resolution reconstruction was proposed initially in 1998 for enhancing the resolution of remotely sensed images using signal processing-based techniques [1]. In general, three different categories can be considered for existing image processing based super-resolution (SR) methods [2]: interpolation-based [3]–[7], reconstruction-based [8]–[12], and learning-based [13]–[14]. Interpolation-based methods use a base function or an interpolation kernel to estimate the value of the unknown pixels on the high resolution grid [15]. Although they are fast and simple, they blur high frequency details and soften texture in the reconstructed high resolution image. Reconstruction-based methods assume that the blurred downsampled version of the reconstructed high resolution image is the same as the low resolution (LR) image. They try to minimize the difference between these two LR images through some prior knowledge [16]. In learning based methods, a training dataset based on low-high resolution images is used to estimate the high frequency details in the reconstructed HR image [2]. The training process in these methods can be either based on internal similarities of the image itself or an external low-high resolution training set pair. In 2015 the first deep learning super resolution framework was proposed by Dong *et al.* [17]. They showed that using convolutional neural networks for super-resolving images is similar to sparse-coding-based learning methods. However, the process of learning dictionaries or manifolding for modeling the patch space was replaced by hidden layers. As a result, there is no need for an external dictionary of the low-high resolution patches in CNN based models. Introduction of the first deep framework of SR models has led to great improvements in high resolution reconstruction results compared to the state-of-the-art learning based methods.

Current state-of-the-art deep learning based methods for SR are based on very deep models that utilize residual layers. Although deeper networks lead to better results, the overfitting issue is an inevitable consequence in these types of networks. To address the problem of overfitting, different strategies have been proposed. Zagoruyko *et al.* [18] had a comprehensive study on wide residual networks. Their study showed that wider residual networks with higher number of feature maps performed better than the deeper residual nets with higher number of layers. According to their results, having deeper networks will not always lead to the best outputs.

In this paper, our goal is to create a shallower network without losing the accuracy. A dual path deep network (DPDN) is proposed to reconstruct a high resolution image from a single low resolution image. The second path of the network helps to provide more information for the upsampling layer without increasing the number of feature maps, which makes the training process computationally inexpensive compared to the wider networks. Moreover, using two different resolution paths provides the opportunity to utilize multi-resolution information in the upsampling process.

II. Related Work

SRCNN model [17], the first deep learning based SR model, used the initial bicubic up-sampled version of a low resolution image as the main input. The basis structure of the network consisted of one patch extraction and representation layer followed by a non-linear mapping layer that mapped the feature maps of low resolution image to the feature maps of the high resolution image. At the end of the network, a reconstruction layer produced the final high resolution image from the extracted feature maps. To improve the efficiency of the network, FSRCNN [19] was proposed in which the pre-processing bicubic interpolation step was replaced by

a deconvolutional post-processing layer. Later, VDSR [20] network was created based on the idea that a deeper CNN should lead to more accurate results. In addition, the idea of residual learning in deep learning-based SR models was utilized in this work for the first time. It was reported that a 20 layer depth using small filters multiple times provided the opportunity to extract contextual information from larger regions of the original image. Results obtained by VDSR outperformed all the previous deep learning SR methods. Although a deeper structure provides more accurate results, it increases the chance of overfitting and the number of trainable parameters. To address these problems, Kim *et al.* [21] proposed the deeply recursive convolutional network (DRCN). The authors used the same convolutonal layers 16 times to increase the depth while avoiding additional parameters and vanishing gradients using a recursive-supervision strategy. Tie *et al.* [22] improved the idea of DRCN by replacing the recursive layer with a recursive block that consisted of two residual units. Each corresponding convolutional layer within the residual units shared weights in a recursive block. Using a LR image as the main input required an extra up-sampling layer to be added to the network as a post-processing step. The deconvolutional layer upsampled an image by adding zero values in between the non-zero pixels. These non-zero values had to be filled with meaningful values during the training process. Shi *et al.* [23] proposed a subpixel convolutional up-sampling layer, called ESPCN. In this method, an up-scaled image was reconstructed by reshaping r^d times more feature maps extracted at the lower resolution, where r is the scale factor and d is the dimension of the image. The performance of ESPCN method was equivalent to SRCNN when using a small training set. However, replacing the training set with the very large ImageNet dataset improved the performance of the ESPCN significantly. In addition, it was shown in [23] that changing the final activation function from $tanh$ to $Relu$ can improve the performance of the network. Inspired by the ResNet network, Ledig *et al.* [24] proposed the SRResNet architecture. This network utilized 16 residual blocks to extract feature maps at the lower resolution. A subpixel upsampling layer created higher resolution feature maps that were used to reconstruct the final high resolution image. The other contribution of [24] was proposing a generative adversarial network (SRGAN) to produce photo realistic high resolution images. Results of proposed SRGAN model were rated as the most similar results to the ground truth by a group of 12 raters, yet they had lower PSNR and SSIM compared to that of the SRResNet model quantitatively. Lim *et al.* [25] improved the performance of SRResNet by eliminating the batch normalization layers. Moreover, the number of parameters were increased in that network not only by increasing the number of residual layers from 16 to 32, but also by increasing the number feature maps in each residual block from 64 to 256. To prevent the network from overfitting, a constant scaling layer was added between two convolutional layers in each residual block, which helped to stabilize the learning process.

In this work, we add another path to the network called

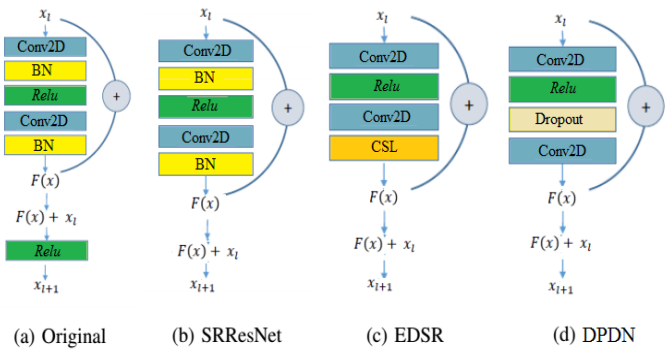

Fig. 1. Comparison of residual blocks.

bicubic path. This extra path allows us to utilize information coming from the higher resolution domain and therefore improve the performance of the network while keeping the network shallow. By adding the second path to the network, we avoid going too deep in the lower resolution path. The lack of information in the up-sampling layer is compensated by concatenating the results of corresponding features from the bicubic path. Also, using the the upsampled image as the input in the second path provides the opportunity to incorporate multi-resolution information in different steps of the upsampling process. The low resolution path consists of three residual blocks to extract feature maps from the input followed by an up-sampling layer. Based on the scale factor, the network will be up-scaled gradually in separate steps. Each up-sampling layer in the low resolution path has an equivalent down-sampling layer in the bicubic path. The proposed DPDN network has shallower structure compared to the state-of-the-art methods which makes the network computationally inexpensive. The methodology of proposed method will be discussed in next section.

III. Methodology

In this section, we describe our proposed method. There are two different types of blocks in our proposed architecture, including feature extraction and resizing. Both paths in the network contain multiple numbers of such blocks.

A. Feature Extraction Block

Recently, residual networks have been used widely in different fields of computer vision applications. Residual networks (ResNets), first proposed by He *et al.* [26], had a deeper network while easing the process of training by adding a skip connection between input layers and outputs of a block to learn the residual functions. ResNet architecture has been successfully customized for the SR reconstruction by removing the $Relu$ activation function layer, located after the addition layer at the original structure [24]. Lim *et al.* [25] showed that the performance of the residual blocks for the SR reconstruction can be increased by eliminating the batch normalization layers.

Inspired by previous works, we incorporate residual blocks without any batch normalization layers to extract feature maps

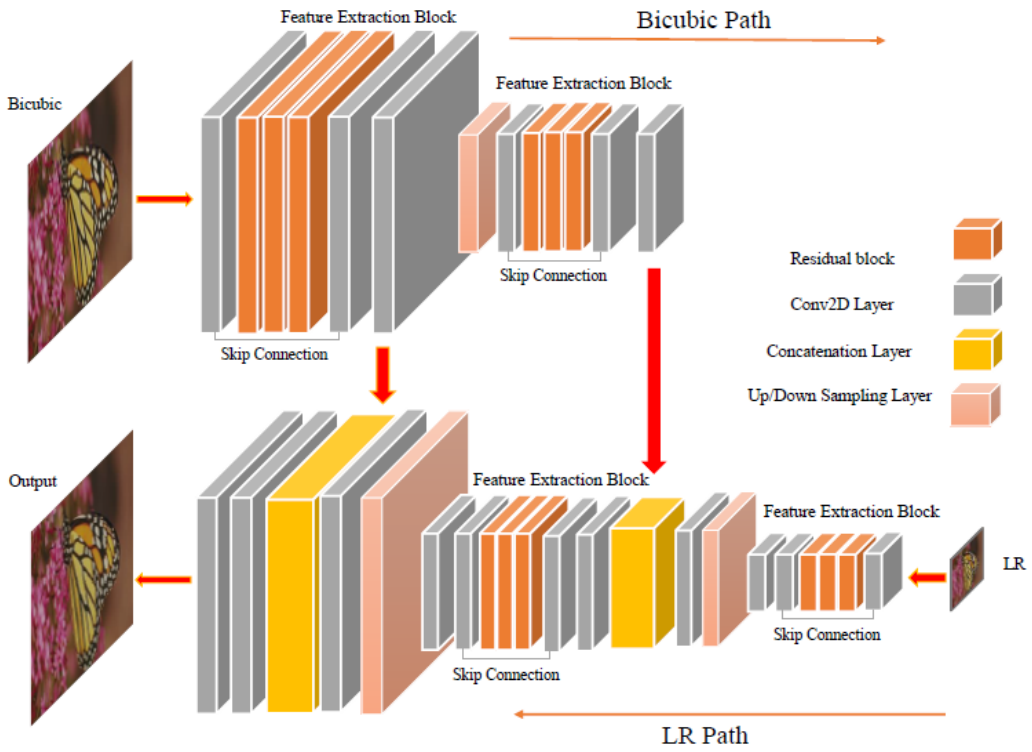

Fig. 2. Overall view of the proposed DPDN.

in both lower and higher resolution levels. A 2D spatial dropout layer is added between the two convolutional layers in each residual block to prevent the network from overfitting. Each feature extraction block in our network consists of three residual blocks and two convolutional layers, located right at the beginning and the end of that block.

Figure 1 shows a comparison of the different residual blocks used in original ResNet, SRResNet, EDSR, and our method.

B. Resizing Block

Figure 2 depicts the proposed architecture in this work. In this figure, the resizing block in the proposed architecture refers to the blocks in which the size of the inputs are changed.

In the low resolution path, the main input is a LR image. Consequently, we add an upsampling layer at the end of the network to reconstruct the higher resolution image. The subpixel upsampling method [23] is used in our network. The lower resolution extracted feature maps are upscaled by a factor of 4 in two separate steps. First, extracted features from the LR image are doubled in size. Then, a feature extraction block is applied to produce the intermediate feature maps. Next, another upsampling process with a scale factor of 2 is used to reconstruct the final image.

In the bicubic path, one bilinear downsampling layer is used to produce the equivalent feature maps of the corresponding layer at the low resolution path.

C. Loss Function

Choosing an appropriate loss function plays an important role in the training of a deep model. A most common loss function used in SR networks is mean square error or *L2 norm*. The ultimate goal of each super-resolution algorithm is to reduce the distance between the reconstructed HR image and the ground truth. However, Lim *et al* [25] reported that replacing *L2 norm* with *L1 norm* boosted the performance and helped the network to converge sooner. In this work, *L1 norm* is used as the loss function in training process. Also, a total variation regularization term is added to the loss function with the weight of 2×10^{-8}.

IV. Experimental Results

A. Dataset

The training set used in this paper is DIV2K dataset [27]. This dataset contains RGB images with a great diversity in content. 800 HR images and their corresponding LR images are provided as the training set. Two sets of LR images are available in this dataset generated by:

- bicubic method, and
- an unknown downsampling method.

1) Training details: To train the network, patches of 48×48 pixels are extracted from RGB bicubic downsampled LR images with their corresponding 192×192 high resolution patches. All the intensity values in both LR and HR images are rescaled to $[0, 1]$. We use Adam optimizer by setting $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$. The initial learning rate is set to

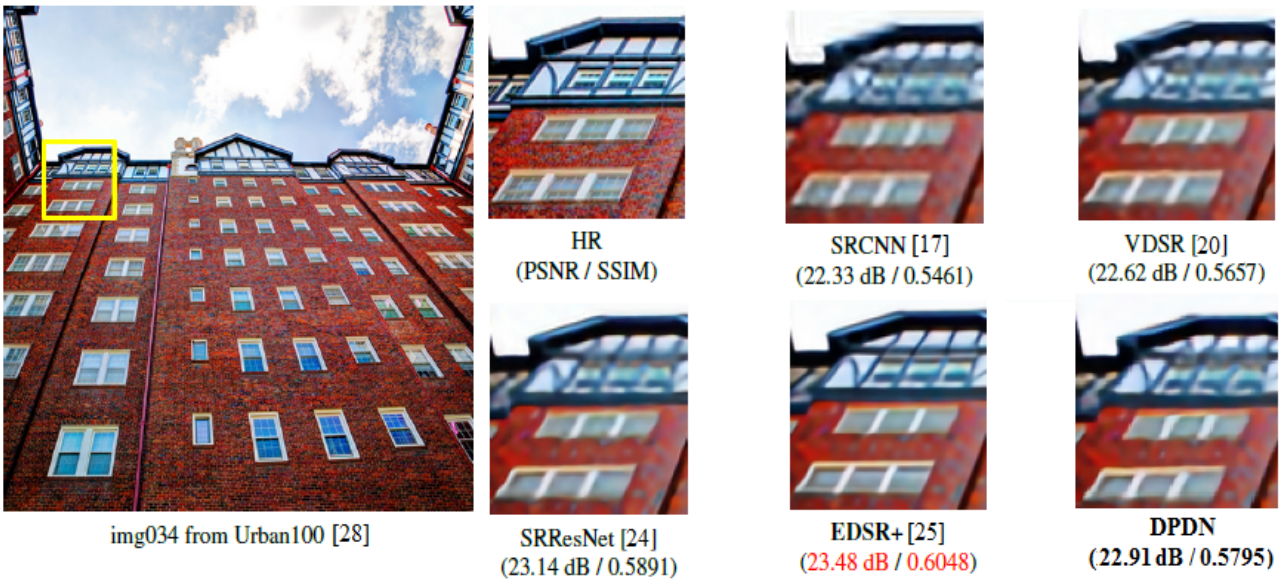

Fig. 3. Visual results compared to SRCNN, VDSR, SRResNet, and EDSR. All the figures are from [25] except our result (DPDN).

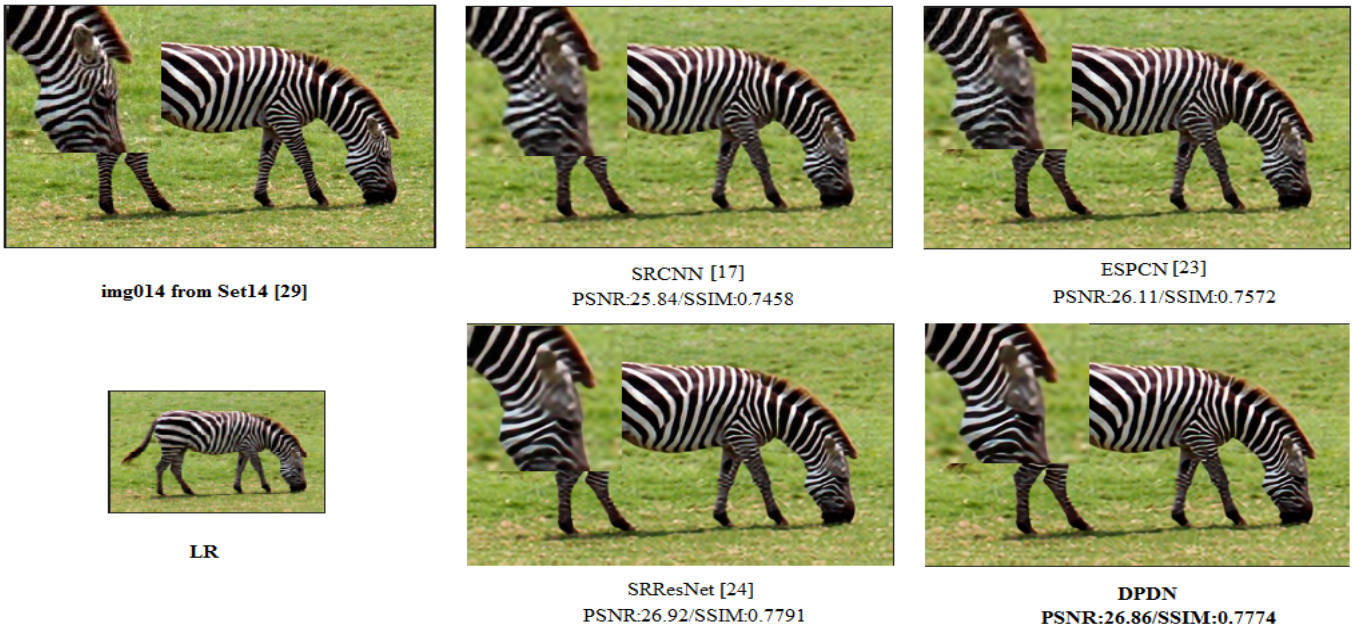

Fig. 4. Visual results upscaled 4 times compared to three stat-of-the-art methods on one of Set14 images.

10^{-4} and then is decreased by a factor of 10 after having the same validation metrics for 5 epochs in a row. The number of feature maps in each layer is 64. During the test phase, the dropout layers are eliminated from residual blocks.

The performance of the network is tested over four commonly used test datasets in SR field, including Set5 [28], Set14 [29], BSD100 [30], and Urban100 [31]. Some visual results are provided in Figures 3 and 4.

2) Evaluation Metrics: The most common metrics used for evaluation of SR models are PSNR and SSIM. PSNR is the ratio of the maximum possible power of original signal to the power of the noise, which is the reconstruction error in the SR reconstruction. While PSNR is more related to the quality of the reconstruction, SSIM is an approximation of sensitivity of the human vision system to the structural information [32].

$$\text{PSNR} = 10\log_{10}\frac{Max^2}{\frac{1}{WH}\sum_{i=0}^{W-1}\sum_{j=1}^{H-1}(I(i,j)-K(i,j))^2} \quad (1)$$

TABLE I
COMPARISON OF DIFFERENT STATE-OF-THE-ART METHODS WITH OUR METHOD ON SET5, SET14, BSD100, AND URBAN100 DATASET, UPSCALED BY A FACTOR OF 4

Method	PSNR/SSIM on Set14	PSNR/SSIM on Set5	PSNR/SSIM on BSD100	PSNR/SSIM on Urban100	average PSNR/SSIM
SRCNN [17]	27.49/0.7503	30.48/0.8628	26.90/0.7101	24.52/0.7221	27.35/0.7613
ESPCN [23]	27.77/0.7586	30.83/0.8687	–	–	–
VDSR [20]	28.01/0.7674	31.35/0.8838	27.29/0.7251	25.18/0.7524	27.96/0.7822
DRCN [21]	28.02/0.7670	31.53/0.8854	27.23/0.7233	25.14/0.7510	27.98/0.7817
DRRN [22]	28.21/0.7720	31.68/0.8888	27.38/0.7284	25.44/0.7638	28.18/0.7883
DPDN	**28.44/0.7777**	**31.98/0.8917**	**27.38/0.7319**	**25.71/0.7740**	**28.38/0.7938**
SRResNet [24]	28.53/0.7804	32.05/0.8910	27.57/0.7354	26.07/0.7839	28.56/0.7977
EDSR [25]	28.80/0.7876	32.46/0.8968	27.71/0.7420	26.64/0.8033	28.90/0.8074

Where $I(i,j)$ is the ground truth intensity, $K(i,j)$ is the reconstructed image intensity, W is number of pixels in the horizontal direction, H is number of pixels in the vertical direction, and Max is the maximum possible power of the signal, which is 1 in this work.

Matlab built-in function has been used to calculate SSIM metric. In this function, an isotropic Gaussian function is used for weighting the neighborhood pixels around a pixel to estimate local statistics. The equation used to estimate SSIM metric is as follows:

$$x = I_{w(i,j)},$$
$$y = K_{w(i,j)},$$
$$\text{SSIM} = \frac{1}{WH}\{\sum_{i=0}^{W-1}\sum_{j=1}^{H-1}\frac{(2\mu_x\mu_y + C_1) + (2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}\} \quad (2)$$

The SSIM index is measured between two $N \times N$ windows where $K_{w(i,j)}$ is the local window from reconstructed image, $I_{w(i,j)}$ is that of ground truth image, and (i,j) is the center of those windows. In Equation 2, μ_x and σ_x, and μ_y and σ_y are the average and variance of intensity values within the reconstructed and the ground truth local windows, respectively. σ_{xy} is the covariance of two local windows. C_1 and C_2 are two variables which stabilize the division with weak denominator where $C_1 = k_1L^2$ and $C_2 = k_2L^2$. L stands for the dynamic range of pixel values and k_1 and k_2 are 0.01 and 0.03, respectively.

B. Comparison with the State-of-the-Art Models

We test our proposed method on four different test sets, including Set5 [28], Set14 [29], BSD100 [30], and Urban100 [31]. Results of four most recent state-of-the-art methods are compared to our method in Table 1. All the PSNR and SSIM values are calculated using Y channel of YCpCr images. As it can be seen from Table 1, our method outperforms the first five deep models. VDSR, DRCN, and DRRN are categorized as the very deep networks. The last two methods are the state-of-the-art for very deep SR models that contain 16 and 32 residual blocks, respectively. Results show that our network performs very close to these methods. Figure 3 and 4 represent qualitative/visual results for two sample images from dataset Urban100 and Set14, respectively. As displayed in Figure 3 and 4, the proposed DPDN method visually leads to very similar results while containing a lower number of trainable parameters. This feature makes the training process more efficient and less time-consuming. For example, as it was mentioned in [25], which is currently the best state-of-the-art model, the process of the training takes 8 days to be completed using NVIDIA Titan X GPUs, while the process of training in our proposed model takes only 3 days using NVIDIA P100 Pascal GPUs.

V. CONCLUSION

In this work, we proposed a dual path network. The main objective of our work was to decrease the depth and width of the network without negatively affecting the performance. Results show that adding a second path to the network leads approximately to a very similar performance compared to the very deep models. Although our network outperforms the VDSR, DRCN, and DRRN models, it slightly performed lower compared to the SRResNet and EDSR models. Both SRResNet and EDSR networks are very deep convolutional networks based on residual layers. SRResNet includes 16 and EDSR comprises 32 residual layers. In each path of our proposed method, only 6 residual layers are utilized. The information from the second path, the bicubic path, helps the network to compensate for the lack of information compared to the methods using deeper networks. Therefore, instead of having deeper model, which may lead to sever overfitting during the training process, parallel information from bicubic upsampled image are added to their corresponding upsampling layers to increase the accuracy of the upsacling process. One future direction to improve the performance of the network could be exploring different upsampling methods to be used on the second path of the proposed network.

REFERENCES

[1] R. Tsai, "Multiframe image restoration and registration," *Adv. Comput. Vis. Image Process.*, vol. 1, no. 2, pp. 317–339, 1984.

[2] J. Jiang, X. Ma, C. Chen, T. Lu, Z. Wang, and J. Ma, "Single image super-resolution via locally regularized anchored neighborhood regression and nonlocal means," *IEEE Transactions on Multimedia*, vol. 19, no. 1, pp. 15–26, 2017.

[3] L. Jing, G. Zongliang, and Z. Xiuchang, "Directional bicubic interpolation-a new method of image super-resolution," *Proceedings of ICMT, Atlantis Press*, pp. 470–477, 2013.

[4] X. Li and M. T. Orchard, "New edge-directed interpolation," *IEEE transactions on image processing*, vol. 10, no. 10, pp. 1521–1527, 2001.

[5] N. Asuni and A. Giachetti, "Accuracy improvements and artifacts removal in edge based image interpolation." *VISAPP (1)*, vol. 8, pp. 58–65, 2008.

[6] H. Kim, Y. Cha, and S. Kim, "Curvature interpolation method for image zooming," *IEEE transactions on image processing*, vol. 20, no. 7, pp. 1895–1903, 2011.

[7] L. Wang, S. Xiang, G. Meng, H. Wu, and C. Pan, "Edge-directed single-image super-resolution via adaptive gradient magnitude self-interpolation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 8, pp. 1289–1299, 2013.

[8] F. Li, X. Jia, D. Fraser, and A. Lambert, "Super resolution for remote sensing images based on a universal hidden markov tree model," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 3, pp. 1270–1278, 2010.

[9] M. Li and T. Q. Nguyen, "Markov random field model-based edge-directed image interpolation," *IEEE Transactions on Image Processing*, vol. 17, no. 7, pp. 1121–1128, 2008.

[10] G. Zhong, S. Xiang, P. Zhou, and L. Yu, "Spatially adaptive tensor total variation-tikhonov model for depth image super resolution," *IEEE Access*, vol. 5, pp. 13 857–13 867, 2017.

[11] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE transactions on image processing*, vol. 13, no. 10, pp. 1327–1344, 2004.

[12] X. Li, H. He, R. Wang, and D. Tao, "Single image superresolution via directional group sparsity and directional features," *IEEE Transactions on Image Processing*, vol. 24, no. 9, pp. 2874–2888, 2015.

[13] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael, "Learning low-level vision," *International journal of computer vision*, vol. 40, no. 1, pp. 25–47, 2000.

[14] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, "Video super-resolution with convolutional neural networks," *IEEE Transactions on Computational Imaging*, vol. 2, no. 2, pp. 109–122, 2016.

[15] S. Dai, M. Han, W. Xu, Y. Wu, Y. Gong, and A. K. Katsaggelos, "Softcuts: a soft edge smoothness prior for color image super-resolution," *IEEE Transactions on Image Processing*, vol. 18, no. 5, pp. 969–981, 2009.

[16] J. Sun, Z. Xu, and H.-Y. Shum, "Image super-resolution using gradient profile prior," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*. IEEE, 2008, pp. 1–8.

[17] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 2, pp. 295–307, 2016.

[18] S. Zagoruyko and N. Komodakis, "Wide residual networks," *arXiv preprint arXiv:1605.07146*, 2016.

[19] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *European Conference on Computer Vision*. Springer, 2016, pp. 391–407.

[20] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1646–1654.

[21] J. Kim, K. Lee, and M. Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1637–1645.

[22] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3147–3155.

[23] W. Shi, J. Caballero, F. Huszár, 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1874–1883.

[24] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4681–4690.

[25] B. Lim, S. Son, H. Kim, S. Nah, and K. M. Lee, "Enhanced deep residual networks for single image super-resolution," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*. IEEE, 2017, pp. 1132–1140.

[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[27] E. Agustsson and R. Timofte, "Ntire 2017 challenge on single image super-resolution: Dataset and study," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

[28] M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L. A. Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *British Machine Vision Conference (BMVC)*, 2012.

[29] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International conference on curves and surfaces*. Springer, 2010, pp. 711–730.

[30] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proc. 8th Int'l Conf. Computer Vision*, vol. 2, July 2001, pp. 416–423.

[31] J.-B. Huang, A. Singh, and N. Ahuja, "Single image super-resolution from transformed self-exemplars," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 5197–5206.

[32] Z. Kotevski and P. Mitrevski, "Experimental comparison of psnr and ssim metrics for video quality estimation," in *ICT Innovations 2009*. Springer, 2010, pp. 357–366.