



Fine Edge and Texture Prior Guided Super Resolution Reconstruction Network

Peng Sun[✉], Jialuo Xu[✉], Shuaishuai Dong[✉], and Yi Chen^(✉)[✉]

School of Computer and Electronic Information/School of Artificial Intelligence,
Nanjing Normal University, Nanjing 210023, China
cs_chenyi@njnu.edu.cn

Abstract. In recent years, significant progress has been made in the field of super-resolution through the use of neural networks. Prior knowledge, such as edges and textures, is commonly incorporated into super-resolution reconstruction networks. However, existing models rely on fixed operators to extract binary edge and texture information, which often capture only rough features and fail to accurately represent the desired edge and texture characteristics. Consequently, this may result in the generation of spurious edges and difficulties in reconstructing image texture details. In this study, we propose a novel super-resolution neural network composed of three branches, with two branches specifically dedicated to extracting fine edges and textures. These two branches take the edge map and texture map of the high-resolution image as the target image, respectively, and are able to construct end-to-end neural networks through loss function constraints. Experimental results demonstrate the superiority of our model in reconstructing sharper edges and finer textures on benchmark datasets, including Set5, Set14, BSDS100, Urban100.

Keywords: Image super-resolution · Prior knowledge · Edge · Texture

1 Introduction

Super Resolution (SR) reconstruction has been a popular research area for the past few decades. The super resolution techniques aim to reconstruct Low Resolution (LR) images to High Resolution (HR) images, and while improving the quality of our life. There are many applications based on super-resolution technology, such as video reconstruction [3, 4], social security [22], medical image enhancement [14, 16], and military remote sensing [26].

With the development of deep learning techniques, super-resolution networks have achieved rapid progress in the field of single image super-resolution (SISR) for the past few years. The SRCNN [33] network was the first neural network designed for this task and showed better results than traditional methods using only three convolutional layers. Researchers later developed deeper networks like VDSR [15], EDSR [18], and MDSR, which achieved even better reconstruction

outcomes. These models mostly extract information directly from low-resolution images to generate high-resolution images, but LR contain limited pixel information which makes it difficult to reconstruct higher quality image information.

Some studies have shown that incorporating prior knowledge, such as edge priors [30,36] and texture priors [28], can provide additional pixel information and improve the reconstruction quality of images. For example, Yang et al. [32] proposed a method that combines edge maps with LR images for super-resolution (SR) reconstruction. Fang et al. [11] introduced an edge network to reconstruct image edges and learn edge features. Wang et al. [28] utilized fixed-class texture priors to effectively reconstruct the texture details in images. Although these methods have achieved good reconstruction results by leveraging prior knowledge, they often overlook the differences between textures and edges, as well as the repetitive nature of textures.

Therefore, we utilize the GLCM (Gray-Level Co-occurrence Matrix) to extract the most frequently occurring texture features in the image. Additionally, we employ an edge detection operator to extract fine-grained edge information. As shown in Fig. 1, the extracted edges and textures differ significantly from the binary edge operator. Although the extracted texture map contains sufficient edge information, there are some spurious edges, so we utilize a refined edge constraint reconstruction to fuse them with features. We use these extracted texture and edge maps as the target images for the edge and texture branches of our network. By incorporating loss constraints, we construct an end-to-end network. In summary, our contributions are as follows:

1. We proposed a novel super-resolution network consists of three branches designed to extract fine-grained edge and texture information. By fusing the extracted edge and texture information, not only the internal texture of the image can be reconstructed, but also the problem of false edges can be solved using the fine-grained edge map. Numerous experiments demonstrate that our network helps to guide the super-resolution reconstruction, thus effectively solving the difficulties of image edge blurring and internal texture reconstruction.
2. We have devised a novel loss function incorporating three components: image content, edge, and texture losses. This integrated loss structure guides our model to converge effectively, enabling accurate reconstruction of image edges and texture details.

2 Related Works

2.1 Single Image Super-Resolution

In recent years, super-resolution techniques have been widely used in the field of single image super-resolution. The development of single image super-resolution can be divided into the following two steps:

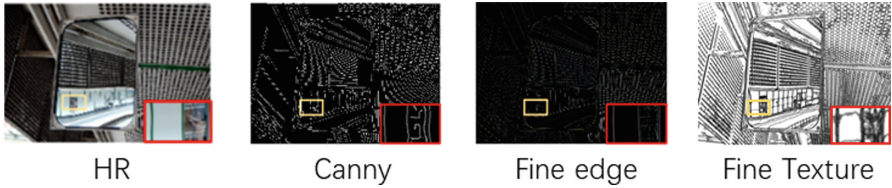


Fig. 1. The figure represents the edge map extracted by Canny, the refined edge map and the texture map used in this paper, respectively.

In the early stages of super-resolution research, conventional methods relied on techniques like linear interpolation and bicubic interpolation for image reconstruction. These interpolation algorithms leveraged neighboring pixel values to estimate missing pixels. While these approaches offered simplicity and flexibility, they faced challenges in accurately reconstructing high-frequency details in super-resolution images.

Subsequently, learning-based methodologies emerged to address the LR-to-HR mapping challenge. These approaches encompassed a range of techniques, including sparse-based methods [31], neighborhood embedding methods [7, 27], random forests [23], and notably, convolutional neural networks (CNNs) [33]. With the rapid advancements witnessed in CNN research, they have risen to prominence, establishing themselves as the prevailing approach in the field. Prominent CNN models such as EDSR [18], RDN [35], SAN [8], and RFA [19] have garnered substantial attention, owing to their remarkable performance in image super-resolution tasks, thus solidifying their position at the forefront of the field.

2.2 Prior Information Assisted Image Reconstruction

In the past few years, super-resolution networks based on prior information have had a great impact on the field of super-resolution. Usually, a complex image contains many edge regions, so the introduction of an edge prior will have an important impact on the reconstruction of complex images. Tai et al. [25] proposed to combine the advantages of edge-directed SR and learning-based SR. Yang et al. [32] proposed an edge-guided recursive residual network (DEGREE) that introduces image edges into a neural network model. The network uses a bicubic interpolation preprocessed LR image as input and uses existing operators (e.g. Sobel detector [10], Canny detector [6] etc.), which introduced additional noise and generates artifacts. Sun et al. [24] used a novel gradient profile prior for super-resolution reconstruction. Li et al. [17] proposed to use edge information to introduce an encoder decoder to reconstruct high-resolution images. Fang et al. [11] proposed the soft-edge information extracted by Edge-Net, which solved the problem of fake edge appearance compared to the ready-to-use edge extractor. Zhao et al. [35] proposed IEGSR to accomplish super-resolution reconstruction using the high-frequency information of the image in the edge region.

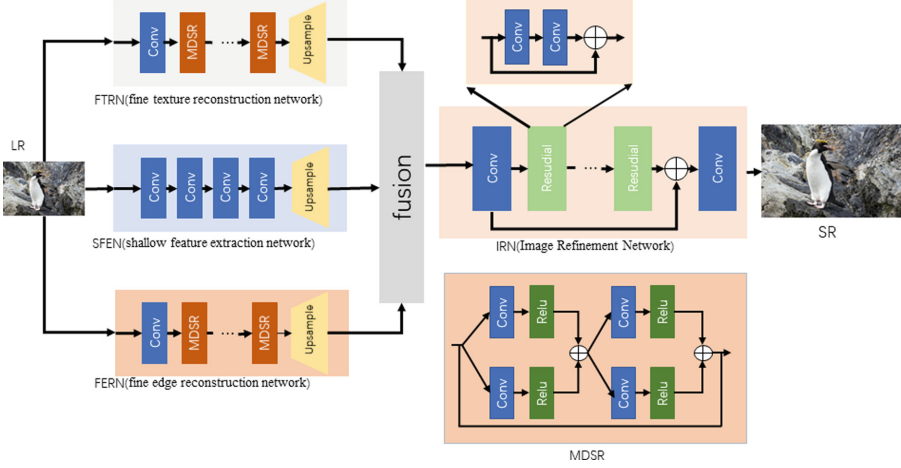


Fig. 2. The general composition of the FETSRSR network consists of four parts: shallow feature extraction network (SFEN), fine texture reconstruction network (FTRN), fine edge reconstruction network (FERN) and Image Refinement Network (IRN).

3 Methodology

3.1 Architecture

The progress of the reconstruction can be divided into the following two steps in our network. As seen in Fig. 2, first, we reconstruct the rough features by the SFEN, fine-edge extraction through the FERN and the texture information extracted by the FTRN. FERN and FTRN contain mainly the MDSR module, which is capable of fully extracting multi-scale edge and texture features through convolutional kernel-size-agnostic convolution. Second, we will concatenate and fuse the fine edge, rough features and fine texture information. The fused image tensor is then fed to the image refinement network and used to recover high quality images. In detail, FETSRSR consists of four modules: shallow feature extraction network (SFEN), a fine edge reconstruction network (FERN), a fine texture reconstruction network (FTRN), and an image refinement network (IRN).

In the first stage, the output of these network can be described as:

$$f_{rough} = F_{SF}(I_{LR}) \quad (1)$$

$$f_{edge} = F_{Edge}(I_{LR}) \quad (2)$$

$$f_{texture} = F_{texture}(I_{LR}) \quad (3)$$

where I_{LR} is the low-resolution image, F_{SF} , F_{edge} and $F_{texture}$ denote the SFEN, FERN and FTRN. f_{rough} , f_{edge} and $f_{texture}$ represent the shallow features, the image fine edge, and the image texture. Then they use fusion layers for merging:

$$f_{fusion} = F_{fusion}([f_{rough}, f_{edge}, f_{texture}]) \quad (4)$$

where \square operation represents the connection of the feature maps and $F_{fusion}()$ denotes the fusion layer, which can achieve features for fusion. In the second stage, we train the fused image tensor using the refinement network.

$$I_{SR} = F_{IRN}(f_{fusion}) \quad (5)$$

where I_{SR} is the reconstructed SR image and $F_{IRN}()$ represents the image refinement network. In training our network, we propose the following loss function to assist the reconstruction process.

$$L_{loss} = L_{content} + \lambda_1 L_{edge} + \lambda_2 L_{texture} \quad (6)$$

where λ_1 and λ_2 are hyper-parameters, $L_{content}$, L_{edge} and $L_{texture}$ denote the loss, and this will be discussed in the following chapters.

3.2 Shallow Feature Extraction Network (SFEN)

First, we use SFEN to extract rough image feature. As a general rule that the rough image features can be easily detected, so a convolutional layer using a 3×3 convolutional kernel is applied to map the image to a high dimension. Then, the low-frequency information of SR image is extracted by using five identical convolutional layers, each layer is represented as:

$$f_n = w_n * f_{n-1} + b_n (n = 1, 2, \dots, 5) \quad (7)$$

where f_n , w_n and b_n represent the feature maps, weights and biases of the current convolutional layer output, respectively. f_{n-1} is the output of the upper layer and it will feed into the current layer, where n varies from 1 to 5. A related study found that sufficient shallow features can be extracted with 5 layers of convolution. Finally, when $n = 5$, we utilize an upsample module to upscale the extracted features to the same dimension of the HR.

$$f_{rough} = F_{up}(f_5) \quad (8)$$

where F_{up} denotes the up-sample module, which consist of a sub-pixel layer and employ two convolutional layers for the transition to the image refinement network. The output f_{rough} represents the image features extracted by convolution at a shallow level.

3.3 Fine Texture Reconstruction Network (FTRN)

A prior information is often used for image reconstruction and has led to significant improvements in image quality. The texture prior is introduced in FTRN, which can reconstruct the texture details of the HR directly from the LR.

The texture information is extracted by the GLCM method. GLCM represents the joint distribution of grayscale of two pixels with some spatial position relationship. The GLCM generation process as follows:

- * A spot (x, y) and a spot $(x + a, y + b)$ in an image form a point pair. Let the pair of spots have a gray value of (f_1, f_2) and let the image have a gray value of at most L , then there will be a $L \times L$ combination of f_1 and f_2 .
- * For the whole image, the number of occurrences of each (f_1, f_2) value is calculated, and then they are arranged into a matrix.
- * The total number of times (f_1, f_2) appears are normalized to obtain the probability $P_{(f_1, f_2)}$, which results in a grayscale co-generation matrix.

After obtaining the GLCM of the HR image, we utilize the properties of GLCM to extract the repeated texture details present in the image. Firstly, a 3×3 convolutional kernel is applied to expand the channels, followed by five multi-scale residual blocks to explore features at different scales. In this part of the network, the texture feature map is upsampled to match the size of the HR image. Throughout this process, the LR image is used as the input to the network, allowing us to directly obtain the texture feature map of the LR image. The specific equations are as follows:

$$L_{texture} = \|T(I_{LR}) - I_{texture}\|_1 \quad (9)$$

where $T()$ denotes the texture reconstruction network, $I_{texture}$ represents the texture of the HR extracted by GLCM, $T(I_{LR})$ displays the reconstructed texture and make L_1 loss with $I_{texture}$

3.4 Fine Edge Reconstruction Network (FERN)

To address the issue of excessive false edges in the texture feature map generated by GLCM, we introduce the following edge extraction operator, which can produce more refined edge features:

$$I_{Edge} = div(u_h, u_v) \quad (10)$$

where $u_i = \frac{\nabla_i I_{HR}}{\sqrt{1 + |\nabla I_{HR}|^2}}$, $i \in \{h, v\}$, h and v represent two dimensions in different directions (horizontal and vertical), ∇ indicate gradient operation and $div()$ indicate the divergence operation. FERN has the same network structure as FTRN, but the target images and loss function utilized are different. The loss function of FERN is shown as:

$$L_{edge} = \|E(I_{LR}) - I_{edge}\|_1 \quad (11)$$

where the method of $E()$ stand for the fine edge reconstruction network, I_{edge} indicates the fine edge extracted by above methods, $E(I_{LR})$ displays the fine edge reconstructed by FERN and make L_1 loss with the fine edge detected by HR images.

3.5 Image Refinement Network (IRN)

For the image refinement module, we integrate the different features extracted from the aforementioned three branches. These three features enable us to capture sufficient texture details and obtain accurate edge features effectively. We

fuse these three features and input them into the feature fine-tuning network. The entire process can be mathematically represented by the following equation:

$$f_{rb}^n = (w_2 * R(w_1 * f_{rb}^{n-1} + b_1) + b_2) + f_{rb}^{n-1} \quad (12)$$

where w_i and b_i are commonly used weight parameters in neural networks of the layer i respectively, $i \in \{1, 2\}$, f_{rb}^n is the output of the n th residual block. $R()$ denotes the ReLU activation function.

In addition to the residual blocks, a long skip connection is used in these part to maintain the features at the input to the IRN and effectively limit the trouble of the disappearance of gradient. For the IRN last layer, we use a convolutional layer to convert the dimension to RGB channels, and then SR images are reconstructed. In training progress, L_1 loss function is used to reduce the gap between SR image and HR image.

$$L_{content} = \|I_{SR} - I_{HR}\|_1 \quad (13)$$

In summary, we have designed a model called FETSR that can effectively reconstruct SR images. Typically, the edge and texture regions of an image contain abundant information that is challenging to reconstruct. Experimental results demonstrate that by incorporating fine-grained edge priors and texture priors, the reconstructed SR images exhibit accurate edges and rich texture details.

4 Experiments

4.1 Datasets

The DIV2K [1] is a common used dataset in super resolution reconstruction tasks, which contains 1000 images of various scenes, 800 of them are used for training, 100 can be used for validation, and 100 can be used for testing. As same as the previous works, we use a training dataset consisting of 800 images from DIV2K to train our model and meanwhile use the validation images from DIV2K to validate our model. During test our model, we employ the following datasets: Set5 [5], Set14 [34], BSDS100 [2] and Urban100 [13]. All of these test datasets are commonly used in super resolution and contain a variety of scenarios that are convincing enough to fully evaluate our model.

4.2 Implements Details

In training our network, we set the patch size to 48 as the input image block size and the batch size to 16, and constrain our training process by L_1 loss, edge loss, and texture loss, and set the weights to 1, 0.1, and 0.001, respectively. Since the pixel values of texture maps range from 0 to 255, we need to constrain it to the same dimension as the edge loss. The parameters of the optimizer are set to $\beta_1 = 0.9, \beta_2 = 0.999$, respectively, and the epoch is set to 600, and our residual block is finally set to 40. All code is based on the pytorch framework and is trained on 2 TITAN Xp GPUS.

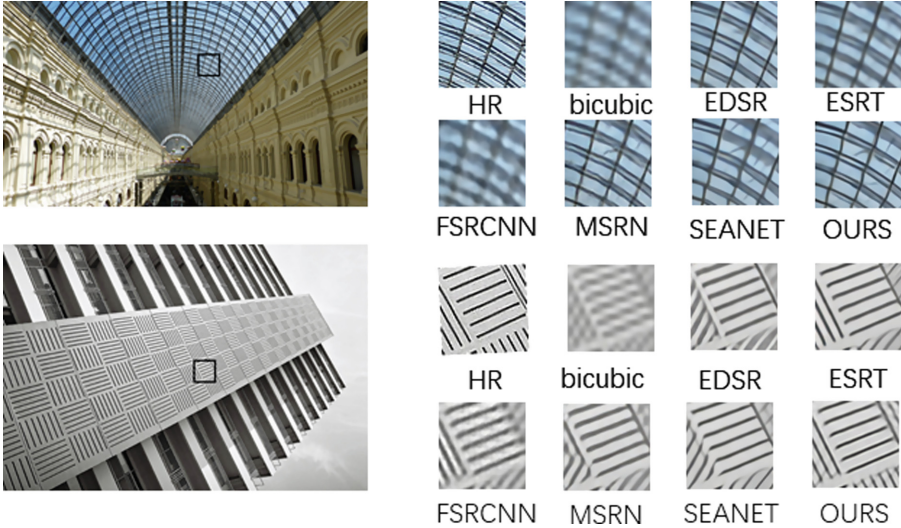


Fig. 3. A comparison of our model with other models shows that our model is able to reconstruct better visual effects and finer texture details.

4.3 Qualitative Comparisons and Discussion

As shown in the Fig. 3, we selected different images from the test dataset and reconstructed them with the available super-resolution. When compared with other super-resolution methods, our network is able to reconstruct not only more accurate texture information, but also sharper edges. In the first image, our reconstructed image has a better visual effect and a clearer reconstruction for some fine textures. In the second image, we reconstructed more accurate texture details. The third image clearly shows that the reconstructed image of our model highlights the edge line part of the floor.

4.4 Quantitative Comparisons and Discussion

As shown in Table 1, our model is compared with other neural network models. PSNR and SSIM are common metrics for judging the quality of reconstruction in super-resolution domains. Other methods have difficulty in reconstructing high quality images by learning the own features of LR images. Our model is able to effectively reconstruct the edges and textures of the images by using the prior generated from HR images, and is higher than other models in both PSNR and SSIM metrics.

5 Analysis and Discussion

5.1 Effectiveness of the Prior Information

It is a very important issue that how to use the effective prior information to aid the super-resolution reconstruction, so we conducted an experimental analysis of

Table 1. Quantitative analysis with existing CNN-based models for super-resolution reconstruction Highlighted indicates the best result.

Scale	Algorithm	Set5 PSNR↑/SSIM↑	Set14 PSNR↑/SSIM↑	BSDS100 PSNR↑/SSIM↑	Urban100 PSNR↑/SSIM↑
×2	SRCNN [33]	36.71/0.9536	32.32/0.9052	31.36/0.8880	29.54/0.8962
	FSRCNN [9]	37.06/0.9554	32.76/0.9078	31.53/0.8912	29.88/0.9024
	VDSR [15]	37.53/0.9583	33.05/0.9107	31.92/0.8965	30.79/0.9157
	SeaNet [11]	38.08/0.9609	33.75/0.9190	32.27/0.9008	32.50/0.9318
	Cross-SRN [20]	38.03/0.9606	33.62/0.9180	32.19/0.8997	32.28/0.9290
	MRFN [12]	37.98/0.9611	33.41/0.9159	32.14/0.8997	31.45/0.9221
	ESRT [21]	38.03/0.9600	33.75/0.9184	32.25/0.9001	32.58/0.9318
	FDSCSR [29]	38.12/0.9609	33.69/0.9191	32.24/0.9004	32.50/0.9315
	FETSR (ours)	38.18/0.9612	33.90/0.9206	32.30/0.9010	32.68/0.9335
×3	SRCNN [33]	32.47/0.9067	29.23/0.8201	28.31/0.7832	26.25/0.8028
	FSRCNN [9]	33.20/0.9149	29.54/0.8277	28.55/0.7945	26.48/0.8175
	VDSR [15]	33.68/0.9201	29.86/0.8312	28.83/0.7966	27.15/0.8315
	SeaNet [11]	34.55/0.9282	30.42/0.8444	29.17/0.8071	28.50/0.8594
	Cross-SRN [20]	34.43/0.9275	30.33/0.8417	29.09/0.8050	28.23/0.8535
	MRFN [12]	34.21/0.9267	30.03/0.8363	28.99/0.8029	27.53/0.8389
	ESRT [21]	34.42/0.9268	30.43/0.8433	29.15/0.8063	28.46/0.8574
	FDSCSR [29]	34.50/0.9281	30.43/0.8442	29.15/0.8068	28.40/0.8576
	FETSR (ours)	34.61/0.9287	30.46/0.8454	29.22/0.8078	28.62/0.8615
×4	SRCNN [33]	30.50/0.8573	27.62/0.7453	26.91/0.6994	24.53/0.7236
	FSRCNN [9]	30.73/0.8601	27.71/0.7488	26.98/0.7029	24.62/0.7272
	VDSR [15]	31.36/0.8796	28.11/0.7624	27.29/0.7167	25.18/0.7543
	SeaNet [11]	32.33/0.8970	28.72/0.7855	27.65/0.7388	26.32/0.7942
	Cross-SRN [20]	32.24/0.8954	28.59/0.7817	27.58/0.7364	26.16/0.7881
	MRFN [12]	31.90/0.8916	28.31/0.7746	27.43/0.7309	25.46/0.7654
	ESRT [21]	32.19/0.8947	28.69/0.7833	27.69/0.7379	26.39/0.7962
	FDSCSR [29]	32.36/0.8970	28.67/0.7840	27.63/0.7384	26.33/0.7935
	FETSR (ours)	32.43/0.8978	28.74/0.7862	27.70/0.7405	26.41/0.7965

Table 2. On scale x4, experimental results generated by different a prior information. Highlighted indicates the best result.

Edge	Texture	Set5 (PSNR↑/SSIM↑)	Set14 (PSNR↑/SSIM↑)	BSDS100 (PSNR↑/SSIM↑)
w/o	w/o	32.15/0.8952	28.55/0.7789	27.41/0.7368
w/o	w	32.32/0.8974	28.71/0.7861	27.67/0.7396
w	w/o	32.33/0.8970	28.72/0.7855	27.65/0.7388
w	w	32.43/0.8978	28.74/0.7862	27.70/0.7405



Fig. 4. Comparison chart of the ablation experiment, w/o indicates that the prior information is not introduced, w indicates that the prior information is introduced.

the factors affecting the super-resolution reconstruction. When we introduce only texture prior, we can see from Table 2 that the reconstruction of the image is not very good. If only fine edge prior information is introduced, the reconstruction effect is not very good for high frequency regions with regular pixel points and textures. As shown in Fig. 4, when we introduce both fine edge prior information and texture prior information, we can reconstruct the details of the image better.

Table 3. For the study of λ

λ	Set5 (PSNR \uparrow /SSIM \uparrow)
$\lambda_1=1, \lambda_2=1$	32.14/0.8874
$\lambda_1=0.1, \lambda_2=0.1$	32.31/0.8950
$\lambda_1=0.1, \lambda_2=0.01$	32.34/0.8962
$\lambda_1=0.1, \lambda_2=0.001$	32.43/0.8978

5.2 Study of λ

During the training process, the setting of hyper-parameters also has an important influence on the reconstruction effect. λ_1 and λ_2 are set to adjust the edge loss and the texture loss respectively. To weigh the influence of fine edges and textures in the reconstruction process, we set λ_1 to 0.1 and λ_2 to 0.001, thus controlling the texture loss and edge loss in the same dimension. From Table 3 We can see that different super parameter settings lead to different reconstruction effects.

6 Conclusion

In this article, we introduce a novel super-resolution network using fine edge and texture priors. The network consists of four components: a shallow feature extraction network, a fine texture reconstruction network, a fine edge reconstruction network, and an image refinement network. Our model uses fine edge and texture prior to not only reconstruct the internal texture details in the image, but also effectively avoid reconstructing the wrong edge information.

Acknowledgements. This work is supported by the National Natural Science Foundation of China (Nos. 62377029 and Nos. 22033002).

References

1. Agustsson, E., Timofte, R.: NTIRE 2017 challenge on single image super-resolution: dataset and study. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 126–135 (2017)
2. Arbelaez, P., Maire, M., Fowlkes, C., Malik, J.: Contour detection and hierarchical image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **33**(5), 898–916 (2010)
3. Belekos, S.P., Galatsanos, N.P., Katsaggelos, A.K.: Maximum a posteriori video super-resolution using a new multichannel image prior. *IEEE Trans. Image Process.* **19**(6), 1451–1464 (2010)
4. Ben-Ezra, M., Zomet, A., Nayar, S.K.: Video super-resolution using controlled subpixel detector shifts. *IEEE Trans. Pattern Anal. Mach. Intell.* **27**(6), 977–987 (2005)
5. Bevilacqua, M., Roumy, A., Guillemot, C., Alberi-Morel, M.L.: Low-complexity single-image super-resolution based on nonnegative neighbor embedding (2012)
6. Canny, J.: A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **6**, 679–698 (1986)
7. Chang, H., Yeung, D.Y., Xiong, Y.: Super-resolution through neighbor embedding. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004, vol. 1, p. I. IEEE (2004)
8. Dai, T., Cai, J., Zhang, Y., Xia, S.T., Zhang, L.: Second-order attention network for single image super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11065–11074 (2019)
9. Dong, C., Loy, C.C., Tang, X.: Accelerating the super-resolution convolutional neural network. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9906, pp. 391–407. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46475-6_25
10. Duda, R.O., Hart, P.E., et al.: *Pattern Classification and Scene Analysis*, vol. 3. Wiley, New York (1973)
11. Fang, F., Li, J., Zeng, T.: Soft-edge assisted network for single image super-resolution. *IEEE Trans. Image Process.* **29**, 4656–4668 (2020)
12. He, Z., et al.: MRFN: multi-receptive-field network for fast and accurate single image super-resolution. *IEEE Trans. Multimedia* **22**(4), 1042–1054 (2019)
13. Huang, J.B., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5197–5206 (2015)

14. Hyun, C.M., Kim, H.P., Lee, S.M., Lee, S., Seo, J.K.: Deep learning for undersampled MRI reconstruction. *Phys. Med. Biol.* **63**(13), 135007 (2018)
15. Kim, J., Lee, J.K., Lee, K.M.: Accurate image super-resolution using very deep convolutional networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1646–1654 (2016)
16. Lee, D., Yoo, J., Tak, S., Ye, J.C.: Deep residual learning for accelerated MRI using magnitude and phase networks. *IEEE Trans. Biomed. Eng.* **65**(9), 1985–1995 (2018)
17. Li, F., Bai, H., Zhao, L., Zhao, Y.: Dual-streams edge driven encoder-decoder network for image super-resolution. *IEEE Access* **6**, 33421–33431 (2018)
18. Lim, B., Son, S., Kim, H., Nah, S., Mu Lee, K.: Enhanced deep residual networks for single image super-resolution. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 136–144 (2017)
19. Liu, J., Zhang, W., Tang, Y., Tang, J., Wu, G.: Residual feature aggregation network for image super-resolution. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2359–2368 (2020)
20. Liu, Y., Jia, Q., Fan, X., Wang, S., Ma, S., Gao, W.: Cross-SRN: structure-preserving super-resolution network with cross convolution. *IEEE Trans. Circ. Syst. Video Technol.* **32**(8), 4927–4939 (2021)
21. Lu, Z., Li, J., Liu, H., Huang, C., Zhang, L., Zeng, T.: Transformer for single image super-resolution. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 457–466 (2022)
22. Ren, S., Li, J., Tu, T., Peng, Y., Jiang, J.: Towards efficient video detection object super-resolution with deep fusion network for public safety. *Secur. Commun. Netw.* **2021**, 1–14 (2021)
23. Salvador, J., Perez-Pellitero, E.: Naive Bayes super-resolution forest. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 325–333 (2015)
24. Sun, J., Xu, Z., Shum, H.Y.: Image super-resolution using gradient profile prior. In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8. IEEE (2008)
25. Tai, Y.W., Liu, S., Brown, M.S., Lin, S.: Super resolution using edge prior and single image detail synthesis. In: *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2400–2407. IEEE (2010)
26. Tang, J., Zhang, J., Chen, D., Al-Nabhan, N., Huang, C.: Single-frame super-resolution for remote sensing images based on improved deep recursive residual network. *EURASIP J. Image Video Process.* **2021**(1), 1–19 (2021). <https://doi.org/10.1186/s13640-021-00560-8>
27. Timofte, R., De Smet, V., Van Gool, L.: Anchored neighborhood regression for fast example-based super-resolution. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1920–1927 (2013)
28. Wang, X., Yu, K., Dong, C., Loy, C.C.: Recovering realistic texture in image super-resolution by deep spatial feature transform. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 606–615 (2018)
29. Wang, Z., Gao, G., Li, J., Yan, H., Zheng, H., Lu, H.: Lightweight feature dedundancy and self-calibration network for efficient image super-resolution. *ACM Trans. Multimed. Comput. Commun. Appl.* **19**(3), 1–15 (2023)
30. Xie, J., Feris, R.S., Sun, M.T.: Edge-guided single depth image super resolution. *IEEE Trans. Image Process.* **25**(1), 428–438 (2015)
31. Yang, J., Wright, J., Huang, T.S., Ma, Y.: Image super-resolution via sparse representation. *IEEE Trans. Image Process.* **19**(11), 2861–2873 (2010)

32. Yang, W., et al.: Deep edge guided recurrent residual learning for image super-resolution. *IEEE Trans. Image Process.* **26**(12), 5895–5907 (2017)
33. Yoon, Y., Jeon, H.G., Yoo, D., Lee, J.Y., So Kweon, I.: Learning a deep convolutional network for light-field image super-resolution. In: *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 24–32 (2015)
34. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: Cohen, A., et al. (eds.) *Curves and Surfaces 2010*. LNCS, vol. 6920, pp. 711–730. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-27413-8_47
35. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2472–2481 (2018)
36. Zhou, Q., Chen, S., Liu, J., Tang, X.: Edge-preserving single image super-resolution. In: *Proceedings of the 19th ACM International Conference on Multimedia*, pp. 1037–1040 (2011)