RESEARCH ON IMAGE SUPER-RESOLUTION RECONSTRUCTION ALGORITHM BASED ON GENERATIVE ADVERSARIAL NETWORK

Lei YANG[1], Zhigang LI[2]✉ and Chenxi LIANG[3]

[1]Dept. of Information and Communication Eng., Harbin Engineering University
(Harbin, Heilongjiang, China)
E-mail: yanglei1@hrbeu.edu.cn
[2]Associate Professor, Dept. of Information and Communication Eng., Harbin Engineering University
(Harbin, Heilongjiang, China)
E-mail: lizhigang@hrbeu.edu.cn
[3]Dept. of Information and Communication Eng., Harbin Engineering University
(Harbin, Heilongjiang, China)
E-mail: TrancyLiang@163.com

Due to the limitations of the imaging system hardware conditions and the influence of external environment and other factors, the edges of the collected images are always blurred and the resolution is not high. To solve these problems, an image super-resolution reconstruction method based on generative adversarial network is proposed. Firstly, according to the characteristics of low resolution images such as lack of detailed information and poor perceptual quality, a generation model with recursive network is designed in this paper. The recursive module is used to extract high-frequency information from the feature graph, suppress useless low-frequency information, and enhance the expressive force of the feature, which promotes the reconstruction of image texture details. In addition, a discriminator composed of deep convolutional neural network is proposed in this study to better distinguish the original high-resolution image from the reconstructed image. At the same time, the pre-trained VGG-19 network is used to calculate the image perception loss before the activation function to help restore the texture details of the image to obtain the high-resolution image. In summary, the experimental results show that the proposed method is superior to other typical methods in image reconstruction performance, and achieves higher image visual quality.

Key Words : *Super-resolution reconstruction, generative adversarial network, recursive module*

1. INTRODUCTION

As is known to all, image is an important way for human to obtain visual information, so image processing technology has broad application prospect and application value. Image super-resolution reconstruction (SR) is a technique to obtain high-resolution (HR) images from one or more low-resolution (LR) images by overcoming the limitations of the hardware conditions of the imaging system and the influence of external environment. In recent years, SR technology has been widely studied and applied in medical imaging, remote sensing image processing, video surveillance and other fields.

SR technology has three kinds of research methods, namely interpolation method, reconstruction method, learning method. There are three SR methods based on interpolation, which are the nearest interpolation method, bilinear interpolation method and bicubic interpolation method. The reconstruction effect of this kind of method is low, and the image after reconstruction is relatively blurred. There are three SR methods based on reconstruction, which are maximum posterior probability method, convex set projection method and iterative back projection method. Such methods cannot deal with complex structures in images. The learning-based method learns the high-frequency details lost in the LR images from a set of low resolution images and HR images. A large number of experiments show its strong ability of image super-resolution, but for inappropriate training samples, it will produce obvious artifacts and unnecessary noise in the composite images.

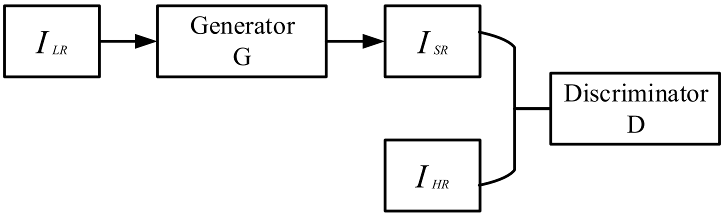

Fig.1 Basic framework of GAN.

2. RELATED WORK

In recent years, with the development of machine learning technology, generative adversarial network[1] (GAN) has been widely used. GAN is mainly composed of generator (G) and discriminator (D). G receives a low-resolution image I_{LR} and outputs a fake high-resolution image I_{SR}. D is essentially a binary classifier, which receives the original high-resolution image I_{HR} and the I_{SR}, and outputs the result of judgment, that is, the true or false of the sample. The basic framework of GAN is shown in **Fig.1**. In Fig.1, the generation model is represented by G and the discriminant model by D. The two models take zero-sum game theory as the core idea source of network training and skillfully use the antagonistic thought training network. For the G, make $D(G(I_{LR}))$ as 1 as possible, for the D, make $D(G(I_{LR}))$ as 0 as possible, and for $D(I_{HR})$ as 1 as possible. I_{SR} is expressed as:

$$I_{SR} = G(I_{LR}) \tag{1}$$

3. PROPOSED METHOD

GAN is based on general convolution. Since the input low-frequency information and high-frequency information of the neural network are treated equally in the channel, the important characteristic information is ignored. Therefore, we have proposed a new SR method combining with the attention mechanism on this basis.

(1) Design of the generator

In SR, the Batch Normalization[2] (BN) layer tends to destroy the image spatial information and weaken the network generalization ability. Therefore, we have removed the BN layer. Image texture details are particularly important for generating high-resolution images. In order to further extract important texture information needed for image reconstruction and suppress useless information, we have introduced recursive mechanism[3] into G. G is shown in the **Fig.2**.

In the figure, the input of the network is I_{LR} and the output is I_{SR}. Firstly, G uses a layer of convolution to extract the shallow features of low-resolution images, which is accompanied by the PReLU activation function to convert linear features into nonlinear features. Then, the extracted shallow feature map is sent to the residual channel attention block[4] (RCAB) as a new input. The RCAB is shown in **Fig.3**.

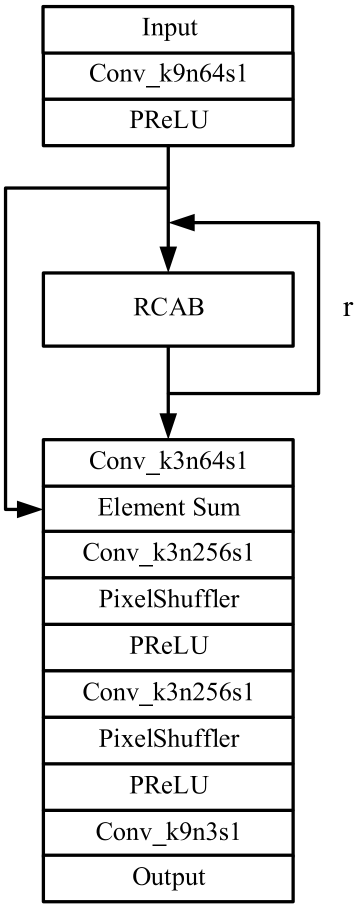

Fig.2 Generator structure: Conv represents convolution layer, k is the size of the convolution kernel, n is the number of convolution kernels, s is the convolution step, PReLU represents the activation layer, PixelShuffler represents subpixel convolution layer, and r is the number of recursions.

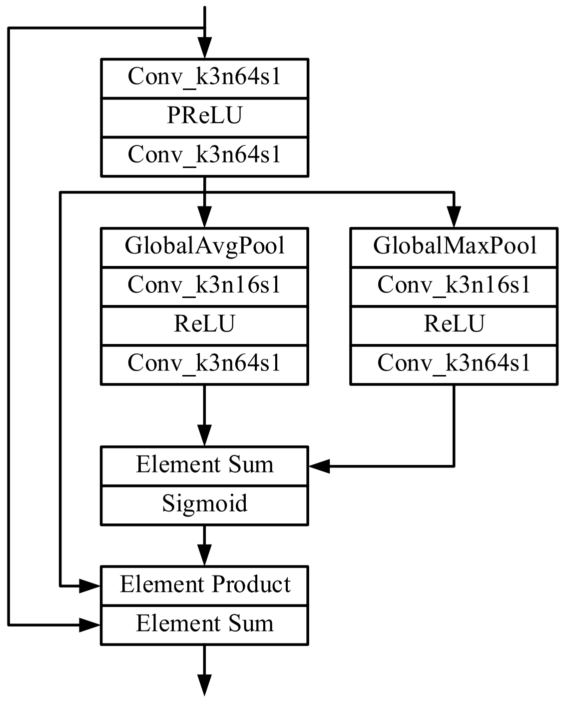

Fig.3 RCAB structure: Sigmoid represents activation layer, GlobalAvgPool represents global average pooling layer, GlobalMaxPool represents global maximum pooling layer, Element Sum represents matrix addition, Element Product represents matrix multiplication.

Finally, the feature map is up-sampled with an amplification factor of 4 through two up-sampling modules and one convolution layer, and I_{SR} is obtained. The up-sampling module is implemented by sub-pixel convolution layer.

(2) Design of the discriminator

In addition to improving G, we also have made targeted improvements to D. The improved D is shown in the **Fig.4**. In this figure, there are two types of input, respectively from I_{SR} and I_{HR}. The output is the probability that the high resolution image is true or false. We have used Leaky_ReLU as the activation function, where $\lambda = 0.2$. In order to reduce the computational load of the whole network, the global average pooling layer[5] is used to replace the full connection layer. Finally, the probability of the high resolution image being true or false is obtained through Sigmoid activation function.

(3) Loss function

In the process of network training, the loss function is of great significance to guide the training of network model. The overall loss function in this paper includes the following parts.

a) Pixel-wise loss

In order to measure the similarity between I_{SR}

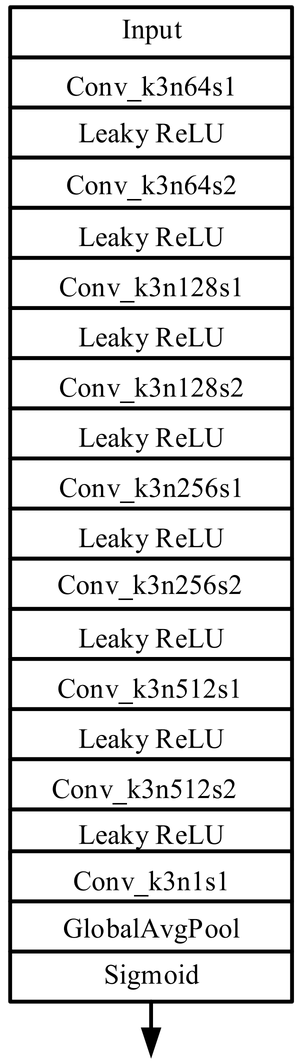

Fig.4 Discriminator structure: Leaky_ReLU represents activation layer.

and I_{HR}, the mean square error (MSE) of the pixels is calculated. The aim is to minimize the difference between I_{SR} and I_{HR} in the pixel domain, which is known as pixel-wise loss. This loss can be expressed as:

$$L_{\text{pixel}} = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} \left(\left(I_{HR} \right)_{x,y} - G\left(I_{LR} \right)_{x,y} \right)^2 \quad (2)$$

where $W \times H$ is the size of the image, and (x, y) is the pixel value in the image.

b) Perceptual loss

In order to compare the advanced perceptual and semantic differences between I_{SR} and I_{HR}, the fully trained VGG-19[6] model is used to extract the features of the input image, and then the feature map before activation is used to calculate the Euclidean distance in the feature space, which is called the perceptual loss. This loss can be expressed as:

$$L_{\text{per}} = \frac{1}{W \times H} \sum_{x=1}^{W} \sum_{y=1}^{H} \Big(\phi_i \big(I_{HR} \big)_{x,y} - \phi_i \Big(G \big(I_{LR} \big)_{x,y} \Big) \Big)^2 \tag{3}$$

where ϕ_i represents the feature values of the before activated layer after the i-th convolutional layer of VGG-19.

c) Adversarial loss

On the basis of pixel-wise loss and perceptual loss, combined with adversarial loss[7], G and D can compete with each other, so that D can extract potential patterns that are difficult to learn from the real reference image, and force the generator to adjust the model, so that the generator can produce realistic high-resolution images.

G's adversarial loss is as follows:

$$L_{G\_ad} = \sum_{n=1}^{N} -\log D\big(G\big(I_{LR}\big)\big) \tag{4}$$

D's confrontation loss is as follows:

$$L_{D\_ad} = -\sum_{n=1}^{N} \Big(\log\Big(1 - D\big(G\big(I_{LR}\big)\big)\Big) + \log D\big(I_{HR}\big) \Big) \tag{5}$$

In our method, the weighted sum function that is obtained by linearly combining three loss functions, namely, the pixel-wise loss, perceptual loss, and adversarial loss functions, is used as the global loss function of our method. It can be expressed as follows:

$$L_G = L_{\text{pixel}} + \alpha L_{\text{per}} + \beta L_{G\_ad} \tag{6}$$

$$L_D = L_{D\_ad} \tag{7}$$

where α and β are the linear combination weights of the corresponding loss functions L_{per} and L_{G_ad}, respectively, in the target function.

(4) Evaluation indicators

In order to accurately evaluate the reconstruction effect of the super-resolution reconstruction algorithm, we mainly adopt two evaluation indicators, namely PSNR and SSIM. PSNR is expressed as:

$$PSNR = 10 \log_{10} \left(\frac{x_{\max}^2}{L_{\text{pixel}}} \right) \tag{8}$$

where $x_{\max}$ is the maximum pixel value in I_{HR}, L_{pixel} is the pixel-wise loss of I_{SR} and I_{HR}.

SSIM is expressed as:

$$SSIM = \frac{\big(2\mu_x \mu_{\hat{x}} + c_1\big)\big(\sigma_{x\hat{x}} + c_2\big)}{\big(\mu_x^2 + \mu_{\hat{x}}^2 + c_1\big)\big(\sigma_x^2 + \sigma_{\hat{x}}^2 + c_2\big)} \tag{9}$$

where μ_x and $\mu_{\hat{x}}$ are the pixel mean values of I_{HR} and I_{SR}, and σ_x and $\sigma_{\hat{x}}$ are the variances, $\sigma_{x\hat{x}}$ is the covariance of I_{HR} and I_{SR}, c_1 and c_2 are constants.

4. EXPERIMENT

(1) Data and similarity measures

We used 800 training images from DIV2K dataset as training set. A training set of 80000 images was obtained by data augmentation method. We performed testing three widely used benchmark datasets Set5 and Set14 and BSD100. All experiments are performed with a scale factor of 4× between LR images and HR images. For fair comparison, all reported PSNR[dB][8] and SSIM measures were calculated on the y-channel of center-cropped, removal of a 4-pixel wide strip from each border, super-resolved images for the reference methods, including bicubic interpolation, SRCNN[9] and SRGAN.

(2) Training details and parameters

The platform of the experiments is a Windows 10 operating system with an Intel 2.9GHz i7-10700 CPU with 8 G memory that is configured with an NVIDIA GTX 1070 GPU, and we trained the model under the GPU-based Pytorch deep learning framework, and the total training time is 45 hours. We obtained the LR images by down-sampling the HR images (RGB, C = 3) using bilinear kernel with down-sampling factor r = 4. For each mini-batch we cropped 16 random 128 × 128 HR sub-images of distinct training images. Note that we can apply the generator model to images of arbitrary size as it is fully convolutional. For optimization we used Adam with β_1. r is 13 recursion in G. In the calculation of the loss function of G, α is 10^{-3} and β is 6×10^{-3}.

(3) Result and analysis

We conducted comparative tests on three test sets, Set5, Set14 and BSD100, and compared the perform-

Table 1 Comparison of bicubic interpolation, SRCNN, SRGAN and ours model on benchmark data.[4× upscaling].

	Bicubic	SRCNN	SRGAN	Ours
Set5	27.58dB/0.79	29.41dB/0.83	30.41dB/0.86	31.09dB/0.87
Set14	25.54dB/0.68	26.82dB/0.73	27.48dB/0.74	27.69dB/0.74
BSD100	25.55dB/0.65	26.46dB/0.70	26.67dB/0.70	26.73dB/0.71

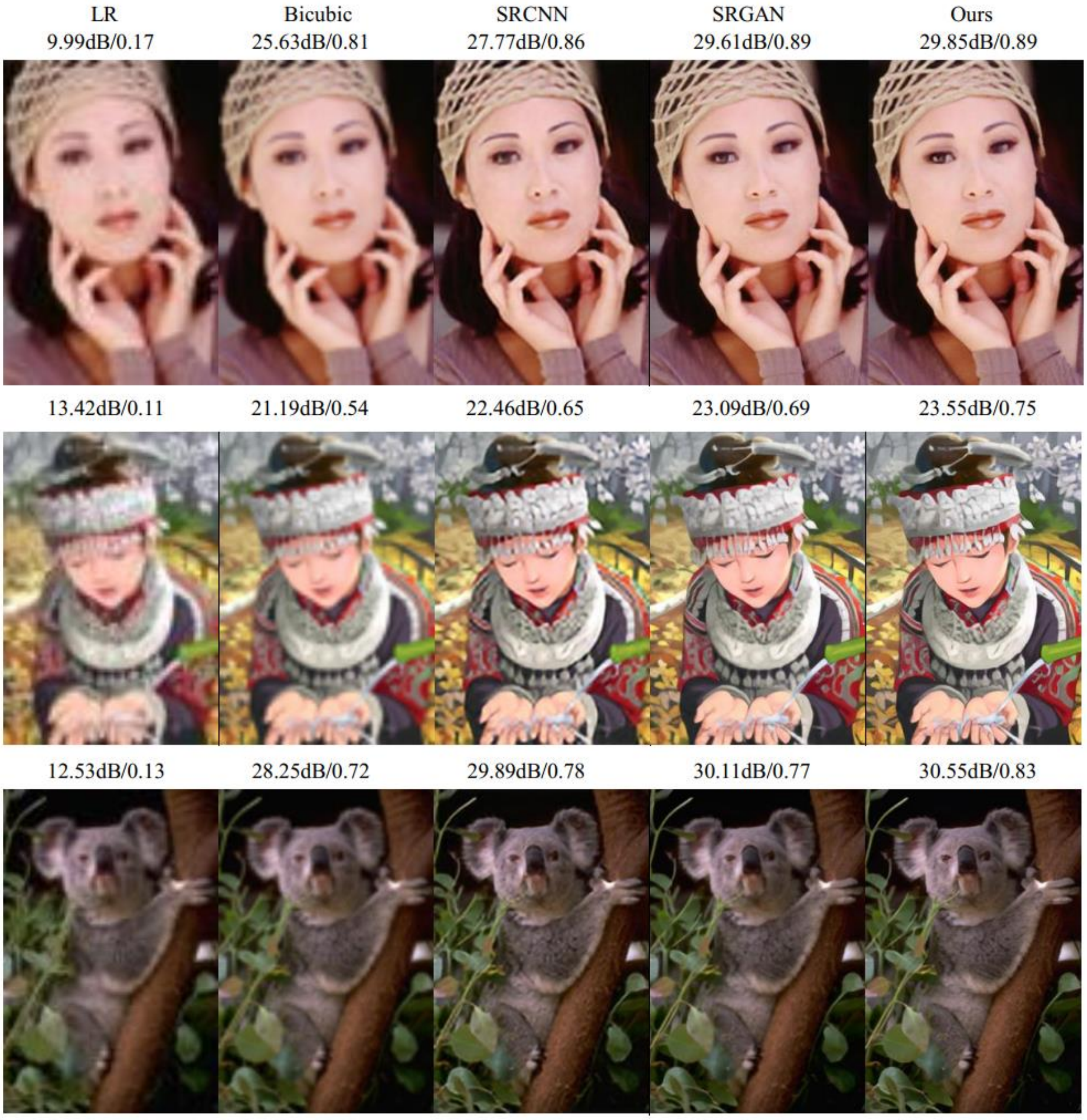

Fig.5 From left to right: reconstructed low resolution images LR, bicubic interpolation,SRCNN, SRGAN and ours. From top to bottom: the test results of Set5, Set14, and BSD100 datasets. The corresponding PSNR and SSIM are shown at the top of the image. [4×upscaling]

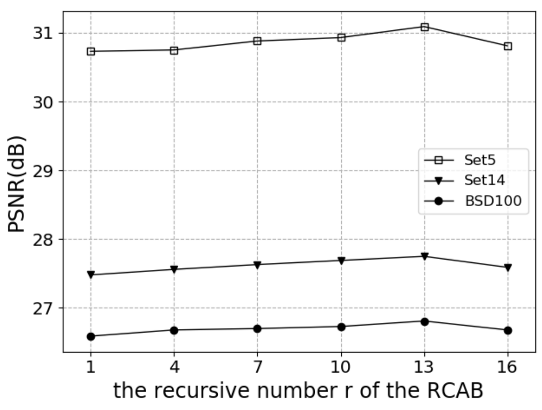

Fig.6 The recursion number r of the RCAB module and the performance of scale factor 4 on the Set5, Set14 and BSD100 datasets.

ance of bicubic interpolation, SRCNN and SRGAN. The quantitative results were shown in **Table 1**. It can be seen that the evaluation indicators of our model are relatively high.

Next, we selected one image from each set of Set5, Set14 and BSD100 for comparison experiment. The algorithms for comparison include bicubic interpolation, SRCNN, SRGAN and our model. This is shown in **Fig.5**. It can be seen that the reconstruction effect of our model is relatively clear.

As the number of recursions increases, the gradient disappearance of the network became increasingly severe. Therefore, we investigated the relationship between the recursion number r of RCAB module and model reconstruction performance, and selected different recursion numbers r and ensured that other conditions were the same to do comparative experiments.The performance evolution of the model with various values of r and a scale factor of 4 on datasets Set5, Set14 and BSD100 is presented in **Fig.6**. According to the experimental results, when the recursion number is 13, the model reconstruction effect is the best.

5. CONCLUSIONS

In this paper, we have proposed a new image super-resolution method, which uses a recursive residual channel attention block based on generative adversarial network to generate high-resolution images. It not only adaptively adjusts the feature channel information to enhance the expressive force of features, but also reduces the generation of pseudo textures to improve the perceptual quality of images. For the loss function, feature values of the VGG-19 network before activation are used to constrain the perceptual loss, richer features can be obtained, making the perceptual loss value more convincing. It can be seen from the experiment that the reconstructed image of our method is relatively clearer, and its evaluation value is relatively high when evaluated by PSNR and SSIM, which are widely used. Next, enhancing the discriminant ability will be the focus of our next research.

REFERENCES

1) Goodfellow I. J., Pouget-Abadie J., Mirza M. : Generative Adversarial Networks, *Advances in Neural Information Processing Systems*, No. 3, pp. 2672-2680, 2014.
2) Wang X., Yu K., Wu S. : Enhanced Super-Resolution Generative Adversarial Networks, *Springer*, pp. 63-79, 2018.
3) Kim J., Lee J. K., Lee K. M. : Deeply-Recursive Convolutional Network for Image Super-Resolution, *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.
4) Zhang Y., Li K., Li K. : Image Super-Resolution Using Very Deep Residual Channel Attention Networks, *Springer*, pp. 294-310, 2018.
5) Lin M., Chen Q., Yan S. : Network In Network, *Computer Science*, 2013.
6) Simonyan K., Zisserman A. : Very Deep Convolutional Networks for Large-Scale Image Recognition, *Computer Science*, 2014.
7) Ledig C., Theis L., F. Huszar. : Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network, *IEEE Computer Society*, 2016.
8) Gupta P., Srivastava P., Bhardwaj S. : A modified PSNR metric based on HVS for quality assessment of color images, *2011 International Conference on Communication and Industrial Application*, 2012.
9) Dong C., Loy C. C., He K. : Image Super-Resolution Using Deep Convolutional Networks, *IEEE Trans Pattern Anal Mach Intell*, Vol. 38, No. 2, pp. 295-307, 2016.