Research on WGAN-based Image Super-resolution Reconstruction Method

Xinying Chen, Shuo Lv, and Chunlin Qian

***Abstract*—Nowadays, obtaining information from images is one of the main ways in which people obtain information. However, the images are affected by the acquisition of hardware equipment and transmission technology, which may result in the loss of certain data and the resolution to be reduced. How to recover low-resolution images into high-resolution images has grown to be a popular area of research for image processing. Deep learning techniques can discover more expressive features through adaptive learning from the dataset. However, there are problems with too many deep network parameters, blurred reconstructed images, and obvious human traces.**

To address these issues, this essay suggests the Deep Recursive Residual Subpixel Wasserstein Generative Adversarial Network (DRRSWGAN) using Deep Recursive Residual Subpixel Network (DRRSN) in the generative network to solve the problems of a large number of network parameters, relatively smooth reconstructed images and the presence of human traces. The Wasserstein Generative Adversarial Network (WGAN) is used in the discriminative network to improve the discriminative network of the GAN for the problem of training instability. Removing the Sigmod layer from the network, optimizing the loss function, and using the RMSProp algorithm for gradient optimization, allows the network to be more stable during training and better image reconstruction.

The outcomes of the trial indicate this method improves both PSNR and SSIM metrics, and the image reconstruction results are better in terms of subjective perception.

***Index Terms*—Image reconstruction, Convolutional neural network, Generator network, Super-resolution.**

I. Introduction

IMAGES are the main way for people to obtain information. Therefore, how to improve image information richness, increase image resolution, and give people access to clearer images is a hot research topic nowadays. Generally speaking, the way to improve image resolution is to upgrade existing acquisition equipment. However, there are many difficulties in upgrading image acquisition equipment to the status of the art today, such as high development costs and long cycles. Therefore, a specialized technique is used to reconstruct the image resolution of low-resolution images from an algorithm perspective to obtain pictures with excellent resolution.

Manuscript received November 21, 2022; revised April 19, 2023. This work was supported in part by the Liaoning Provincial Science and Technology Department under Grant 1655706734383; in part by the Liaoning Natural Science Foundation Program Grant 2019-MS-036; and in part by the Basic Scientific Research Projects of Colleges and Universities of Liaoning Provincial Department of Education under Grant LJKMZ20220826.

X. Y. Chen is an associate professor at School of Computer and Communication Engineering, Dalian Jiaotong University, Dalian, LiaoNing, 116021, China. (e-mail: chenxy1979@163.com).

S. Lv is a postgraduate student at School of Computer and Communication Engineering, Dalian Jiaotong University, Dalian, LiaoNing, 116021, China. (e-mail: lvshuo4262506@163.com).

C. L. Qian is a postgraduate student at School of Computer and Communication Engineering, Dalian Jiaotong University, Dalian, LiaoNing, 116021, China. (e-mail: 1061128982@qq.com).

Super-resolution image problem was first introduced by Harris [1], whose main objective was to reconstruct a low-resolution image to obtain a high-resolution image. Before the twentieth century, researchers usually used interpolation to address the issue of excessively high resolution images, for instance bilinear interpolation [2]. However, this method depends on the assumption of continuity between images, and there is no additional information added to the reconstructed image. This leads to blurrier image reconstructions and poor perception of the reconstructed images. After the twenty-first century, researchers have proposed methods for image super-resolution reconstruction determined by machine learning. Freeman et al. [3] pioneered the introduction of approaches for machine learning to the field of image super-resolution, such as neighborhood embedding [4] and local linear regression [5]. All these techniques record the similar local structures corresponding between pictures of high and low resolution, and then input the data into the dataset. In the dataset, images with a similar structure to it are queried and reconstructed by analyzing the low-resolution image data.

Deep learning has quickly developed as a result, some researchers have applied deep learning to image super-resolution reconstruction tasks. In 2014, Dong et al. [6] proposed SRCNN networks (Super-Resolution Convolutional Neural Networks), Which used convolutional neural net works to rebuild images with super resolution. In terms of the network depth structure, Kim et al. [7] proposed the VDSR (Very Deep Super-Resolution) method. Kim believed that the greater the depth of the network layers, the better the image feature representation ability could be extracted. Experimentally, it was found that although the deeper network could achieve good image reconstruction, the convergence speed of the network was slow. Subsequently, Kim uses gradient truncation by introducing residual learning to improve the efficiency of network training. It reduces resource consumption and improves the results of the image super-resolution reconstruction task. Zhang et al. [8] proposed the RCAN (Residual Channel Attention Networks) method, which combines residual learning and residual module. This method has better image reconstruction results.

Some researchers have found that the use of deep networks, while allowing for richer parameters, also leads to over-fitting. Therefore, researchers have proposed recursive networks, which can reuse the parameters in the network to improve the reconstruction efficiency of images. Kim et al. [9] proposed the DRCN (Deep Recursive Convolutional Networks) method, which uses recursive ideas to construct recursive blocks and increase the perceptual field of the network. The training difficulty of the network is reduced, and a better image reconstruction effect is achieved using relatively few network parameters. Inspired by the DRCN method, Tai et al. [10] proposed the DRRN (Deep Recursive

Residual Network) method. A residual module was added to the recursive module, and the network was trained to learn using local residuals and global residuals to improve the efficiency of completing the image super-resolution reconstruction task.

Traditional image super-resolution reconstruction methods usually use a pixel-by-pixel loss function. It makes the reconstructed image higher in terms of the Peak Signal to Noise Ratio (PSNR) evaluation index, but often results in blurred reconstructed images and poor subjective perception by the human eye. To solve this problem, Ledig et al. [11] proposed SRGAN (Super-Resolution Generative Adversarial Network), which inputs low-resolution images directly into the generative network, and the generative network outputs the reconstructed images. The rebuilt picture and the real image are input to the discriminative network at the same time, and the discriminative network determines whether or if the created image is the true image. Qingliang Zeng et al. [12] proposed the RSRGAN network, using the ResNeXt network as a generative network and the idea of WGAN to improve the SRGAN. It reduces the network complexity and eliminates the issues with unsteady GAN training and low training rate.

The approaches for reconstructing super-resolution images based on deep learning described above have obtained several outcomes, but also suffer from a few problems:

1) Image extraction of characteristics using the depth network works well, but has a large number of network parameters that are very computationally intensive in image super-resolution reconstruction and serious memory consumption.

2) The reconstructed images lack detailed information and have obvious human traces.

3) The method for super-resolution picture reconstruction using generative adversarial networks suffers from training difficulties and instability. Both the PSNR and SSIM metrics of the reconstructive pictures are low.

This paper focuses on approaches for reconstructing images with high resolution that are based on deep learning, and proposes the Deep Recursive Residual Sub-pixel Wasserstein Generative Adversarial Network (DRRSWGAN) to solve the issues with the image super-resolution reconstruction techniques discussed above. Improvements were made to the generator and discriminator networks to improve the PSNR and SSIM metrics of the reconstructive pictures while providing a better subjective perception of the reconstructed images.

The following is a summary of our contributions:

(1) In the generator network, Deep Recursive Residual Network (DRRN) is proposed based on a Deep Recursive Residual Sub-pixel Network (DRRSN). It reduces the complexity of the network by residual learning and extracts deep image features. And it can resolve the gradient issue that is the root of the loss of deep network layers by combining recursive ideas and reusing parameters between network layers. The image feature map is expanded and reorganized using sub-pixel convolution to improve the PSNR and SSIM indices of the reconstructed image. A texture loss function is introduced to enable the generated images to have richer detailed information.

(2) In the discriminator network, the idea of WGAN is combined with the RMSProp optimization algorithm for gradient optimization. The Sigmod layer is removed from the discriminator network with the loss functions of the generator and discriminator networks improved. The weight update parameters in the discriminator network are intercepted to a fixed range to resolve the issue of generator adversarial networks' unstable and demanding training.

II. Related Work

The deep learning-based image super-resolution reconstruction method uses convolutional neural networks to convert low-resolution pictures into their equivalent high-resolution pictures. This method predicts low-resolution images and generates some of the detailed information that high-resolution images have, leading to picture reconstruction with extremely high resolution. Many widely used approaches for super-resolution picture reconstruction are based on deep learning, such as SRCNN [6], DRRN [10], VDSR [7], SRGAN [11], SRRESNET [11], etc.

A. Deep convolutional networks for image super-resolution reconstruction

SRCNN proposes a deep learning-based approach to image super-resolution by performing an end-to-end mapping of the same image in the low-resolution case and the high-resolution case, using a convolutional neural network to train this mapping relationship. Compared to traditional image super-resolution methods, SRCNN improves both image reconstruction quality and image reconstruction speed. The SRCNN structure is shown in Fig.1.

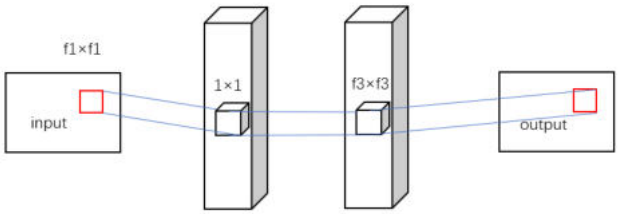

Fig. 1. Structure of SRCNN

The SRCNN is split into three steps: The input image is first subjected to extracting features, and the extracted features are transferred to the next layer. A non-linear mapping is then added by expanding the quantity of layers in the network, and these features are combined and reconstructed by a convolutional layer. A high-resolution image is typically generated and the high-resolution features are averaged. Finally, the resulting high-resolution image is output.

SRCNN still has a lot of flaws, however. For instance, amount of network layers is insufficiently deep, and the picture features are not extracted well. If network layers are added, there will be fake traces in the reconstructed images and a very substantial loss of resources during training. Researchers have been interested in figuring out how to extract deep image features while lowering network complexity, safeguarding image data, minimizing human traces, and regulating network parameters.

B. Generative adversarial networks for image super-resolution reconstruction

With the emergence of rebuilding techniques with high resolution for images based on deep learning, some researchers have found that although these methods can reconstruct low-resolution images, the reconstructed images suffer from loss of texture details. While the SRCNN method is capable of reconstructing low-resolution images, the reconstructed images are too smooth and not ideal for subjective human eye vision. Therefore, Ledig et al. [11] presented a reconstruction of an image with extremely high resolution method (SRGAN) for generative adversarial networks. The idea of generative adversarial networks was used by alternating training of generative and discriminative networks to increase the rebuilt pictures' realism. The SRGAN structure is shown in Fig.2 and Fig.3.

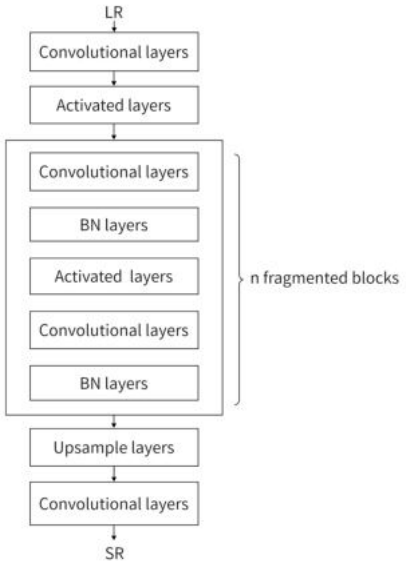

Fig. 2. Structure of generator network

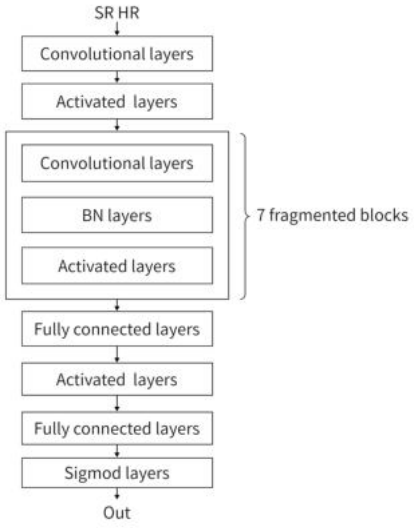

Fig. 3. Structure of discriminator network

SRGAN contains two main networks, the generator network and the discriminator network, as can be seen in Fig.2 and Fig.3. SRGAN employs a deep residual network with hopping connections in the generator network. It guarantees effective gradient information flow within the network and increases the network's resilience. The generator network receives the low-resolution image as an input, and it first performs a convolution operation to separate out its shallow features. Following deep feature extraction in the residual module, combining these methods yields a high-resolution picture and enlarging the image features. The created image is compared to the original high-resolution image in the discriminator network. By extracting features from the two images, the likelihood that the generated picture is owned by the real image is determined.

1) Generator network: The generator network of SRGAN uses the idea of residual networks and mapping from beginning to conclusion. The input low-resolution image is convolved in the first layer for image feature extraction. After 16 residual blocks for deep extraction of image features, the image is scaled up to a set magnification in the upsampling layer. Finally, the image is output.

2) Discriminator network: In the discriminator network, the generated image and the original image are fed into the first convolutional layer for feature extraction. The Leakyrelu activation function is used to avoid the maximum pooling of the whole network. After that, the image information enters into 7 convolutional layers. The size of the convolution kernel is set to 3×3 and the number of feature channels is doubled for every two convolution layers. Each time the number of features is doubled, string convolution is used to lower the image's resolution. The two fully connected layers and the Sigmod layer are used to determine whether the generated image is a real image.

SRGAN improves the problem of overly smooth reconstructed images with little texture detail in image super-resolution reconstruction. Although the subjective perception of SRGAN reconstructed images is improved, the reconstructed image PSNR and SSIM metrics are relatively small. Moreover, the training of SRGAN is very unstable and will lead to training failure in severe cases. The generator network of SRGAN uses residual blocks to reduce the training difficulty of the deep network, but the number of parameters in the network needs to be reduced. Moreover, after the deep network, some information is lost in the image.

III. Method

Deep learning-based image super-resolution reconstruction methods have some problems, and this paper proposes the Deep Recursive Residual Subpixel Wasserstein Generative Adversarial Network (DRRSWGAN). The method consists of a generator network and a discriminator network with recursion and residuals. The number of parameters in the network is controlled while extracting the deep features of the image. The image features are combined and amplified using sub-pixel convolution. The idea of WGAN is introduced to make the image super-resolution method of generative adversarial networks more stable in training and better in reconstructing images.

A. Deep recursive residual subpixel Wasserstein generative adversarial network process

Sub-pixel convolution is combined with Deep Recursive Residual Network (DRRN) [13] in the generator network

to form a Deep Recursive Residual Sub-Pixel Network (DRRSN). The idea of recursive residuals and sub-pixel convolution is used to further extract image features and reduce the number of network parameters. The image information is enriched and human traces in the reconstructed images are reduced. The generative adversarial network is optimized using WGAN ideas in the discriminator network to improve the stability of the network training and enrich the diversity of the generated images, where DRRSN is used as the generator network to form the Deep Recursive Residual Sub-Pixel Wasserstein Generative Adversarial Network (DRRSWGAN). The DRRSWGAN method flow is shown in Fig.4.

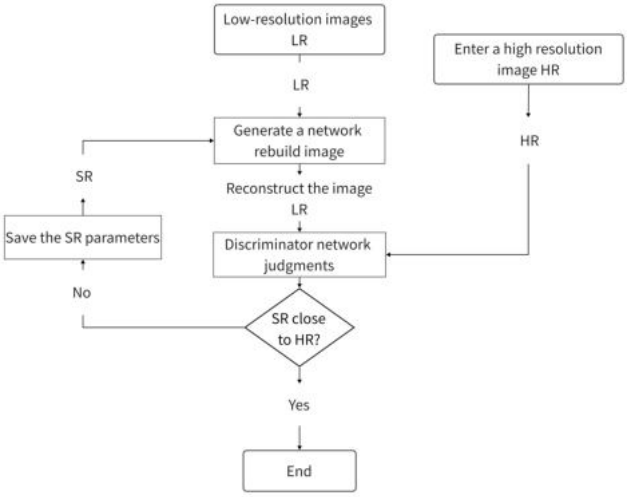

Fig. 4. Flow chart of DRRSWGAN

The specific process steps of the DRRSWGAN method are as follows.

(1) The low-resolution image corresponding to the high-resolution image is input to the generator network (DRRSN), and the first convolutional layer in the generative network extracts the image features.

(2) In the recursive block (which contains a series of residual blocks), image features are extracted deeply, and local residual learning is performed to preserve image information and reduce the difficulty of network training through global learning. The weight parameters within the same residual block are shared to reduce the network parameters.

(3) As soon as the picture exits the recursive residual layer, the image features are scaled up by a sub-pixel convolution layer. The reconstructed image is then output through a final convolution layer.

(4) The generated high-resolution image and the original high-resolution image are fed together into the discriminator network for judgment. If the difference between the two is significant, then the generator network will continue to be trained until the difference between the two is minimal.

B. Deep recursive residual subpixel Wasserstein generative adversarial network architecture

1) Generator network (deep recursive residual sub-pixel networks): In this paper, we use DRRSN as the generator network and remove the BN layer from the residual unit and add a sub-pixel convolution layer to the network. Recursive blocks, residual blocks, and sub-pixel convolution are used in the generator network. Allowing the generator network to fully extract image features while also controlling the parameters in the generator network reduces the resources consumed by the training network. The use of sub-pixel convolution allows for better scaling of the image and removal of artificial traces from the reconstructed image. Generator network organization is shown in Fig.5.

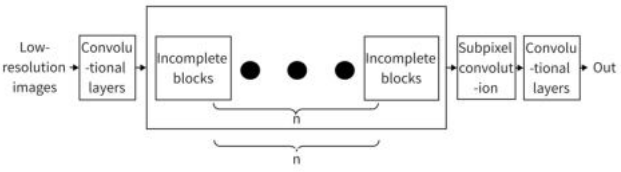

Fig. 5. Structure of generator network

As can be seen, by the structure of the generative grid in Fig.5, the input low-resolution image first undergoes the first layer of convolution for image shallow feature extraction. The input image has a height of M and a width of N and consists of three channels R, G, and B. The image is usually represented by a matrix of $H \times W \times 3$ (H and W are the positions of the image pixels). After the first layer of convolution, the representation of the input image is changed to $H \times W \times C$ (C is the number of channels). After that, the image goes into n recursive blocks and the residual blocks perform deep feature extraction on the image. The image features are enlarged and reconstructed using a sub-pixel convolutional layer, and finally a high-resolution image is generated using a convolutional layer, which is activated using the PRelu function in the generator network.

A residual block in Fig.6 consists of two residual units, and there are multiple residual blocks in a recursive block. By introducing recursive learning into the residual branch, different residual units in a general residual network receive different data. In a DRRSN with residual units, the input of residual units in the same branch is the same, which can further facilitate network training and learning. Global residual learning reduces the difficulty of network training. The generator network uses recursive modules, residual modules and sub-pixel convolution to extract deeper features of the image. Reducing the training burden of the generator network and saving system resources allows the network to be trained more easily and to further improve the results of image reconstruction.

(1) Residual units

The residual unit's structure is depicted in Fig.6. Each residual cell consists of an activation layer and a weight layer, as shown in equation (1).

$$\hat{x} = C(x) = \sigma(F(x, W) + h(x)) \tag{1}$$

where $C(x)$ is the output, σ is the activation function, $F(x, W)$ is the residual mapping, W is the weight value and $h(x)$ is the input. In generative networks, like DRRN, the activation does not come after but before the convolutional layer, called pre-activation. This can make training easier and the residual unit formula for pre-activation is (2).

$$H^r = F\left(H^{r-1}, W^r\right) + H^{r-1} \tag{2}$$

where r is the number of residual units and F is the residual function. H^{r-1} and H^r are the input and output of the rth residual cell, respectively. To make the input to the residual unit in the recursive block the same for all branches, the H^{r-1} is changed to the result of the first convolutional layer C and the weights W are also fixed and shared among the residual units. As a result, the network may learn incredibly complicated properties. The use of branching path learning helps with a gradient backpropagation, facilitating network learning and making it less prone to overfitting.

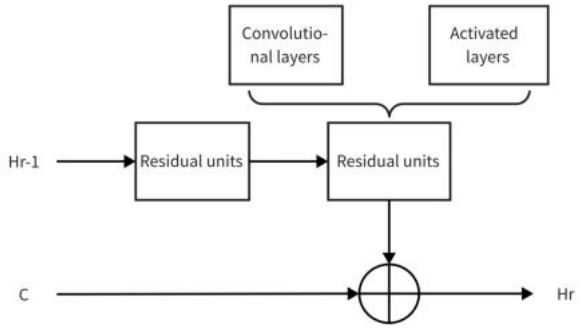

Fig. 6. Structure of a residual element

(2) Design of the BN layer

The residual unit is present in the BN layer, which contributes significantly to convolutional warp networks. It can speed up the network training and has a regularization effect, prevents network overfitting and improves network training accuracy. However, the BN layer also has several problems.

1) In the image super-resolution reconstruction method, the image information may be lost due to the batch normalization operation of the BN layer.

2) In image super-resolution reconstruction methods, it is rare for the network to be over-trained, so there is no need to regularize the network.

3) In the image super-resolution reconstruction method, as the training data distribution is not the same for each batch, adding a BN layer causes the network to re-adapt to the new data distribution in each training session. It has a significant impact on the training speed of the network.

For these reasons, the BN layer is removed from the residual units of the generator network, which speeds up network training and preserves the information in the images, facilitating the removal of artifacts and improving generalization.

(3) Recursive modules

In recursive blocks, the parameters of residual units within the same residual block are shared. Recursive blocks are good for controlling how many variables there are at the network layer and saving network computation costs. The structure of the recursive blocks is shown in Fig.7.

In Fig.8, the number of recursive blocks can be changed and the number of residual blocks in each recursive block is not unique. The nth residual cell can be represented by equation (3) as follows.

$$H_n^r = Q\left(H^{r-1}\right) = F\left(H_n^{r-1}, W_n\right) + H_n^0 \tag{3}$$

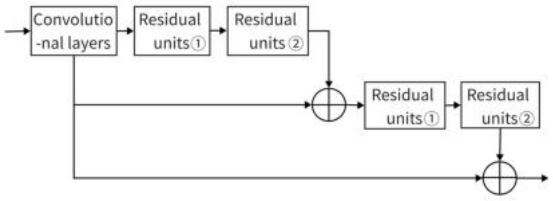

Fig. 7. Structure of recursive blocks

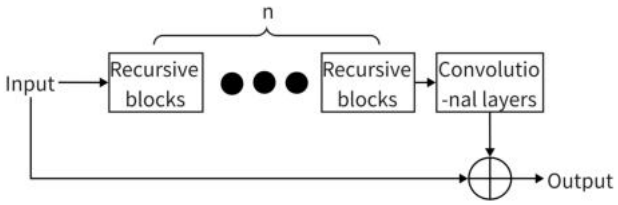

Fig. 8. Network composed of several recursive blocks

where r is the number of residual units, and F is the residual function. H_n^{r-1} and H_n^r are the input and output of the rth residual unit, respectively. H_n^0 is the result of the first layer of convolution, which in turn leads to the output of the nth recursive block, as shown in equation (4).

$$x_n = H_n^r = Q^r\left(f_n\left(x_{n-1}\right)\right) = Q\left(Q\left(\ldots\left(Q\left(f_n\left(x_{n-1}\right)\right)\right)\right)\right) \tag{4}$$

Equation (4) performs a collapse operation on Q^r . The folding operation allows the values of each element to be added cumulatively, which is useful for image feature reconstruction. Afterward, a convolutional layer allows residual learning to be performed on the high-resolution and low-resolution pictures, adding the learning results to the global mapping.

2) Discriminator network: The existing SRGAN method has achieved good results in image super-resolution reconstruction though. However, SRGAN reconstructed images are relatively low in PSNR and SSIM evaluation metrics. There are still some problems with the training of SRGAN. The reason is that the effective gradient descent of the GAN only occurs in the case of convex functions, which do not update well in parallel with each other if both the generator and discriminator networks are neural networks. GAN training is difficult because there is no criterion to judge how well the network is trained during the training process. During training, the GAN is vulnerable to crashes, and the generator network could deteriorate as training goes on. As a result, the generator network's ability to effectively improve the images it produces is compromised, and it stops learning. The discriminator network will make a mistake during a network crash during training, stopping the training process. To solve these problems, the idea of WGAN was introduced to make changes to the discriminator network and combined with the generator network to construct a DRRSWGAN method for image super-resolution reconstruction.

(1) WGAN

Wasserstein GAN [14], abbreviated as WGAN, was proposed in response to a series of problems in GAN. Firstly, the Wasserstein distance is introduced, which is defined as shown in equation (5).

$$W\left(P_r, P_g\right) = \inf_{\gamma \sim \Pi\left(P_r, P_g\right)} E_{(x,y)\sim\gamma}[\|x-y\|] \quad (5)$$

Where $\Pi\left(P_r, P_g\right)$ is the joint distribution of the various outcomes following the combination of phases P_r and P_g. For each joint distribution γ, the generating sample y and the true sample x can be obtained by sampling from $(x, y) \sim \gamma$, and the distance model between the two is calculated as $\|x-y\|$. This leads to the calculation of the sample pair distance expectation under the joint distribution γ as $E_{(x,y)\sim\gamma}[\|x-y\|]$. It is hoped that the lower bound taken on the Wasserstein distance will be used to establish the anticipated value in all potential joint distributions, which is the minimum distance from P_r to P_g. One benefit is the Wasserstein distance over the KL scatter and JS scatter problems in GAN is that if the two distributions do not cross and overlap, the distance between them can also be expressed. Both the KL scatter and JS scatter are abrupt and both are always exceptionally large or small. Both KL scatter and JS scatter are abruptly variable, always exceptionally large or small. In contrast, the Wasserstein distance is smooth. When applying gradient descent methods to optimize parameters, Wasserstein distance provides the gradient. KL scatter and JS scatter fail to provide an effective gradient. If two distributions do not overlap in a high-dimensional space, the Wasserstein distance will provide meaningful gradients, while the KL scatter and JS scatter will not.

WGAN is the introduction of the Wasserstein distance concept into GAN, which may be defined as the loss of the network, thus allowing the network to obtain meaningful gradients. But $W\left(P_r, P_g\right)$ is not directly provable via equation (5). The WGAN authors perform a theoretical derivation to derive the new expression, as shown in equation (6).

$$W\left(P_r, P_g\right) = \frac{1}{K} \sup_{\|f\|_L \leq K} E_{x\sim P_r}[f(x)] - E_{x\sim P_g}[f(x)] \quad (6)$$

There exists a Lipschitz continuum in equation (6), which is a continuous function f. The existence of a constant $K \geq 0$ is required, and it is required that any two elements $x1$ and $x2$ in the domain of definition satisfy equation (7).

$$|f(x1) - f(x2)| \leq K|x1 - x2| \quad (7)$$

At this point the Lipschitz constant for the function f is K. If the domain of the function f is the set of real numbers, then all the above conditions are equivalent to the fact that the absolute value of the derivative function of f does not exceed K. If the derivative function has no upper bound, then it is not Lipschitz continuous because this condition limits the most localized change in a continuous function. Equation (7) means that the Lipschitz$\|f\|L$ of the function f in the case of not exceeding the constant K, satisfying the above condition f can be taken to the upper bound of $E_{x\sim P_r}\left[f(x)\right] - E_{x\sim P_g}\left[f(x)\right]$ in dividing by the constant K, which will be introduced into the neural network becomes the equation (8).

$$K \cdot W\left(P_r, P_g\right) \approx \max_{w|f_w|l\leq K} E_{x\sim P_r}\left[f_w(x)\right] - E_{x\sim P_g}\left[f_w(x)\right] \quad (8)$$

In this equation, w is the parameter of the neural network. Where the magnitude of the constant K does not influence the direction of the gradient and simply causes the gradient to be K-fold expanded. Also, a range interception $[-c, c]$ is performed for the parameters in the neural network. Therefore, the derivative of the input data will not exceed a certain range, and the function f will not vary beyond this range. In turn, a discriminator network is obtained by limiting w to a range while the last layer is not linearly activated. The equation is shown in (9).

$$L = E_{x\sim P_r}\left[f_w(x)\right] - E_{x\sim P_g}\left[f_w(x)\right] \quad (9)$$

L is the Wasserstein distance between the generated and true distributions. WGAN does not use the Sigmoid layer in the last layer of the discriminator network. Due to the approximately fitted Wasserstein distance, WGAN turns the dichotomous classification task in GAN into a regression task. The generator network needs to minimize the Wasserstein distance and there is no need to worry about vanishing gradients in the generator network. The generator network loss and the discriminator network loss of WGAN are represented by equations (10) and (11) respectively.

$$lossG = -E_{x\sim P_g}\left[f_x(x)\right] \quad (10)$$

$$lossD = E_{x\sim P_g}\left[f_x(x)\right] - E_{x\sim P_r}\left[f_x(x)\right] \quad (11)$$

Therefore, the smaller the Wasserstein distance, the better the network is trained. WGAN solves the problem of unstable GAN training. It does not require extra effort to be invested in balancing the generator and discriminator networks during the training process. The collapse mode problem rarely occurs in WGAN, ensuring the diversity of the generated samples. WGAN provides a metric to indicate the effectiveness of training, which can be used to determine how well the network has been trained. A smaller metric means that the network is better trained and the quality of the reconstructed images from the generator network is higher. Inspired by WGAN, the structure of the discriminator network is shown in Fig.9.

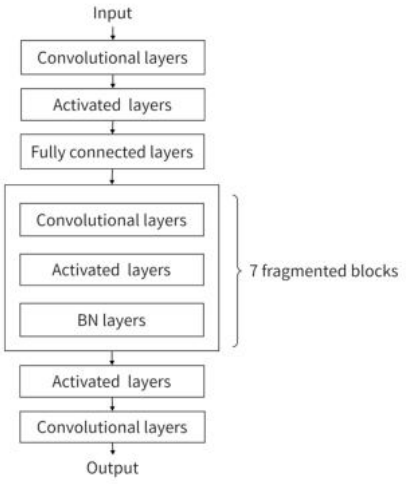

Fig. 9. Structure of discriminator network

It is seen that, following the idea of WGAN, in the discriminator network at the Sigmod is taken off of the final

layer. This is achieved by turning the problem of binary classification in the discriminator network into a regression problem in WGAN. From the problem of determining the probability that the generated image belongs to the real image to the problem of the distance between the generated image and the real image. The discriminator network consists of eight convolutional layers, with the number of channels doubling after every two convolutional layers. The discriminator network examines the difference between the generated and original images, analyses the differences between them, and outputs a judgment. If the original image is better than the generated image, the discriminator network sends the result to the generator network, which continues to train, increasing the similarity between the generated image and the source image. The training methodology for DRRSWGAN is shown in Algorithm 1.

Algorithm 1 DRRSWGAN training method

Start N iterations
  Start of discriminator network c iterations
    m high-resolution images I_i^{HR} are randomly acquired in the training set, $i = 1, 2, 3, ..., m$.
    m corresponding reconstructed images I_i^{SR} are acquired in the generator network, $i = 1, 2, 3, ..., m$.
    Calculate the loss $lossD$.
    Update the discriminator network parameters θD using RMSProp.$\theta D \leftarrow \theta D + \mu \cdot RMSProp(\theta D, lossD)$
    Intercepting discriminator network parameters.θD : $\theta D \leftarrow \theta D + clip(\theta D, -c, c)$
  End of c iterations
  M corresponding low-resolution images I_i^{LR} are randomly acquired in the training set, $i = 1, 2, 3, ..., m$.
  Calculate the loss $lossG$
  Update the discriminator network parameters θG using RMSProp. $\theta G \leftarrow \theta G + \mu \cdot RMSProp(\theta G, lossG)$
End of N iterations

3) loss function: SRGAN improves the realism of the reconstructed image by improving the loss function, but there are still some gaps between the reconstructed image and the real image. It is mainly reflected in the image details that still have blurring or artifacts. Inspired by ESRGAN [15], texture loss was added to the loss function of the generator network in DRRSWGAN [16]. Texture loss is a loss function optimized by a perceptual function and has a high sensitivity to the transfer of image styles. The usage of texture loss can increase how well reconstructed images work. The generator network loss function of DRRSWGAN consists of content loss, adversarial loss, and texture loss, which is formulated as shown in the composition equation (12).

$$L^{SR} = L_{MSE}^{SR} + 10^{-3} L_{Gen}^{SR} + 10^{-3} L_T \tag{12}$$

Among them, content loss uses the MSE loss function. MSE improves the PSNR value better and performs a pixel-by-pixel calculation of the reconstructed image and the original high-resolution image. Its equation is shown in (13).

$$L_{MSE}^{SR} = \frac{1}{r^2 WH} \sum_{x=1}^{rW} \sum_{y=1}^{rH} \left(I_{x,y}^{HR} - G_{\theta G}\left(I^{LR}\right)_{x,y} \right)^2 \tag{13}$$

where $G_{\theta G}\left(I^{LR}\right)$ denotes the high-resolution image generated by the generator network and I^{HR} denotes the true high-resolution image. And the adversarial loss allows the discriminator network to make a more realistic judgment, the generated image or the original high-resolution image, and allows the generator network to learn better, as shown in equation (14).

$$L_{Gen}^{SR} = \sum_{n=1}^{N} -D_{\theta D}\left(G_{\theta G}\left(I^{LR}\right)\right) \tag{14}$$

Inspired by WGAN, the log is removed from the adversarial loss. this is done to make the network easier to train and converge. Finally, there is the texture loss function, whose equation is shown in (15).

$$L_T = \left\| G\left(\theta\left(I_{SR}\right)\right) - G\left(\theta\left(I_{HR}\right)\right) \right\|_2^2 \tag{15}$$

I_{SR} and I_{HR} represent the reconstructed image and the original high-resolution image, respectively, and θ denotes the network parameters. The detailed information of the reconstructed images is enriched by introducing texture loss functions. Next is the discriminator of network losses, which is given by (16).

$$L_D = \frac{1}{N} \sum_{n=1}^{N} \left(D_{\theta D}\left(G_{\theta G}\left(I^{SR}\right)\right) - D_{\theta D}\left(I^{HR}\right) \right) \tag{16}$$

Where N denotes the number of images, I^{SR} denotes the reconstructed image, and I^{HR} denotes the original high-resolution image. The loss function in the discriminator network is mainly to judge how the generated image differs from the original. The smaller L_D is, the closer the generated image is to the real image.

IV. Experimental Settings and Analysis

A. Dataset

1) Trainning Set: The experimental training set adopts the mainstream dataset of current image super-resolution reconstruction, and the training set consists of DIV2K [17] and Flickr2K [18]. DIV2K dataset is very representative in the field of image super-resolution reconstruction, and is mainly used in the image super-resolution reconstruction competitions of NTIRE and PIRM. The DIV2K dataset contains many RGB images, with 900 high-resolution images with 2K resolution. 2650 images in Flockr2K, containing landscapes, people, animals, etc. Some of the images in the training set are shown in Fig.10.

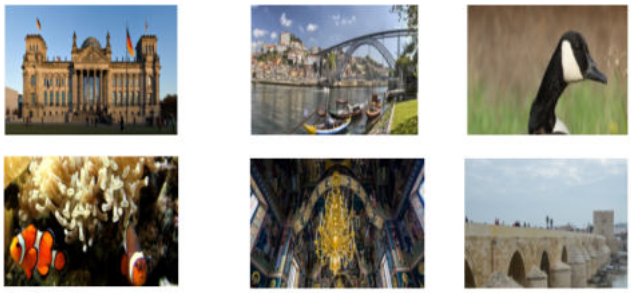

Fig. 10. Partial images in the training set

2) Test Set: The three most popular benchmark datasets were used in the test set, namely Set5 [19], Set14 [20], and BSD100 [21]. With five images in the Set5 dataset, the Set5 dataset is the most widely used test set for image super-resolution reconstruction, as shown in Fig.11. The Set14 dataset contains more images compared to the Set5 dataset, and the resolution of the images in the dataset is higher. There are 100 images in the BSD100 dataset.

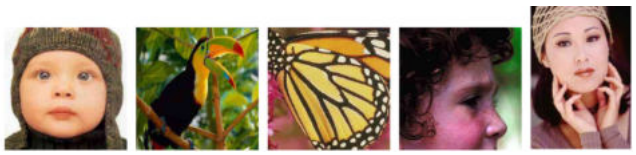

Fig. 11. Images in Set5 dataset

B. Experiments

1) Parameter settings: To ensure that the experimental data are convincing, the image super-resolution reconstruction method and other methods in this paper experiment on the same software and hardware. DRRSWGAN uses the RMSProp optimization algorithm recommended by WGAN. One is added by the RMSProp optimization technique. The issue of early end-of-network training is well-solved by the RMSProp optimization algorithm as compared to other optimization techniques. Smooth objectives are likewise well-suited for the RMSProp optimization technique. The RMSProp optimization algorithm is also well suited to handle smooth objectives. The learning rate is set to 10^{-4} and the momentum parameter is 0.9. At half an epoch, the network's learning rate decreases. At [-0.01, 0.01], the discriminative network update parameters are also intercepted. The network ran roughly 460,000 iterations with a 130-round epoch. The network performed approximately 460,000 iterations with an epoch round of 130. The generator network parameters are set as shown in Table 1.

TABLE I
GENERATOR NETWORK PARAMETERS

Method	DRRSN
Number of residual blocks	20
Number of recursive blocks	1
Training image block size	96
Number of feature channels	64
Whether to use the BN layer	No
Batch size	256
Size of convolutional kernels in the first and last layers	9*9
Intermediate network layer convolutional kernel size	3*3

The discriminator network parameters are shown in Table 2.

2) Training process: (1) Data loading

Before the image super-resolution reconstruction, the training data must be loaded and the data list needs to be generated. First, all the images in the training set are cropped, and the cropping position of the images is random. After cropping an image sub-module of 96×96 size is obtained and this sub-module is used as the original high-resolution image. After that, this high-resolution image is downsampled and reduced to 24×24, and this image is used as the input of the generator network, which is the low-resolution image corresponding to the high-resolution image. This operation allows the network to retrieve the training data more easily during training.

TABLE II
DISCRIMINATOR NETWORK PARAMETERS

Method	DRRSWGAN
Training image block size	96
Number of feature channels	64
Whether to use the BN layer or not	Yes
Batch size	64
Optimization algorithm	RMSProp

(2) Reconstructing images

1) Generator network

After the training images have been processed, the low-resolution images are reconstructed. Firstly, the first convolutional layer is passed through. The number of input channels in the first convolutional layer is 3 and the number of output channels is 64. To enable better extraction of image features, the convolution kernel size of the first convolution layer is set to 9×9 and activated by the PRelu activation function.

After the convolution layer, the image of size 24×24 goes back into the residual recursion module. The recursive module consists of several residual blocks, and within the residual block, there are two residual units. However, different residual blocks have the same residual unit. For example, two residual modules $q1$ and $q2$. these two residual modules each have two residual units, labeled $c1$, and $c2$. The size of the convolution kernel in the residual layer is set to 3×3. the number of input channels and the number of output channels is set to 64. The weights in the same residual branch are shared by a recursive operation. The residual operation extracts the image features.

After that, the image information is passed to the sub-pixel convolution layer. The subpixel convolution layer extracts and expands the features of the image, expanding the size of the low-resolution image from 24×24 to 96×96. The size of the convolution kernel of the subpixel convolution layer is 3×3. The number of input channels and output channels is both 64.

After the image has passed through multiple layers, it comes to the last layer of the network, which is mostly employed to reconstruct the extracted image features. The reconstructed image is converted into the same format as the low-resolution image, except that the size of the image becomes 96×96.

2) Discriminator networks

The generated high-resolution image is fed in at the same time as the original high-resolution image is fed into the discriminator network. All the convolutional layers in the discriminator network are set to have a convolutional kernel size of 3×3, and the number of channels in the first and second layers is set to 64. The number of input channels in the first layer is 3. The discriminator network has 8 convolutional layers, and the number of channels is doubled after every two convolutional layers. The disparity between

the high-resolution image that was created and the original is determined by the discriminator network. If the gap is too large, the generator network continues to learn and makes the created image's quality increases until it reaches a point where it resembles the original high-resolution image.

3) Evaluation metrics: Two main evaluation metrics are used for image evaluation, namely PSNR, and SSIM, which are very important evaluation criteria in image super-resolution.

(1) PSNR

PSNR, known as Peak Signal to Noise Ratio [22] in dB, is an objective criterion to assess image quality. After performing image super-resolution task reweighting, there is a difference between the reconstructed image and the original image. And PSNR is to evaluate the difference between the reconstructed image and the real image, which is defined as shown in equation (17).

$$PSNR = 10 \cdot \log_{10}\left(\frac{\mathrm{MAX}_{\mathrm{I}}^2}{\mathrm{MSE}}\right) \tag{17}$$

where MAX is the maximum pixel value in the image and MSE as shown in equation (18):

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j)]^2 \tag{18}$$

m,n is the image size. K is the noise image. I is the noise-free image. i and j are the image pixel positions. The smaller the MSE value, the larger the PSNR value. And the larger the PSNR, the better the image quality.

(2) SSIM

SSIM, fully known as Structural Similarity [23], is a measure of the degree of similarity between images. Estimates of luminance are expressed as mean values and estimates of contrast are expressed as standard deviation The degree of structural similarity is expressed by the covariance. It is defined as shown in equation (19).

$$SSIM(x,y) = \frac{(2u_x u_y + c_1)(2\sigma_{xy} + c_2)}{(u_x^2 + u_y^2 + c_1)(\sigma_x^2 + u\sigma^2 + c_2)} \tag{19}$$

where u_x is the mean of x, u_y is the mean of y, σ_x^2 is the variance of x, σ_y^2 is the variance of y, and σ_{xy} is the covariance of x and y. $C_1 = (k1L)^2$, $C_2 = (k2L)^2$, $C_1$and C_2 are constants. L is the dynamic range of pixel values. $k1$ and $k2$ are constants 0.01, 0.03, respectively. Structural similarity takes values in the range [0, 1]. The more similar the reconstructed image is to the real image, the closer the SSIM is to 1.

C. Experimental analysis

1) Number of residual blocks: The image's feature extraction using the deep network can be done effectively. The residual blocks increase accuracy while lightening the deep network's computational load. The DRRN research demonstrates that the reconstructed image effect is relatively unaffected by the number of recursive blocks. As a result, in this study, the number of recursive blocks is fixed to 1. For PSNR and SSIM metrics for various numbers of residual blocks, the method presented in this study (DRRSN) is contrasted with the DRRN method. On 9 and 20, the number of residual blocks is compared. The test sets have an amplification factor of 4, and they are Set5, Set14, and BSD100. In Table 3, PSNR comparisons are displayed.

Table 3 shows that for the same number of residual blocks in the Set5 dataset, when the image magnification factor is 4, the DRRSN algorithm performs better than the DRRN algorithm, whether the residual blocks are 9 or 20. The experimental findings for the number of residual blocks set to 9 were all inferior to the results for the identical algorithm when the number of residual blocks was set to 20. The method in this paper (DRRSN) is more effective than other comparison algorithms when the number of residual blocks is 20. In the Set14 dataset, DRRSN has a PSNR difference of 0.11 dB over DRRN. In the BSD100 dataset, DRRSN is 0.091dB higher than DRRN in PSNR. As can be observed, the reconstructed image has a greater PSNR as the number of network layers is increased.

The SSIM comparison is shown in Table 4.

As shown in Table 4, when the Set5 dataset's number of residual blocks is nine, DRRSN performs better than DRRN in terms of SSIM when the image magnification factor is four. DRRSN is 0.008 greater than DRRN in SSIM in the Set5 dataset when the number of residual blocks is 20, even though the SSIM measure of DRRSN is not as excellent as DRRN in the Set14, BSD100 dataset. DRRSN is 0.004 greater than DRRN at SSIM in the Set14 dataset. DRRSN is 0.03 greater than DRRN at SSIM in the BSD100 dataset.

According to the aforementioned experimental findings, the DRRSN approach works superior to the DRRN method for reconstructed pictures with a residual block count of 20 in terms of PSNR and SSIM metrics.

2) Network convergence: There are many problems with deep networks for training, such as large LOSS fluctuations, uncountable network parameters, gradient disappearance, and gradient explosion. The training process of DRRSN, in this essay, the effectiveness of the other four strategies is contrasted, and the training process of the discriminatory network of DRRSWGAN in this paper is compared with that of the discriminatory network of SRGAN. Comparative analysis's findings are shown in the following multi-graphs.

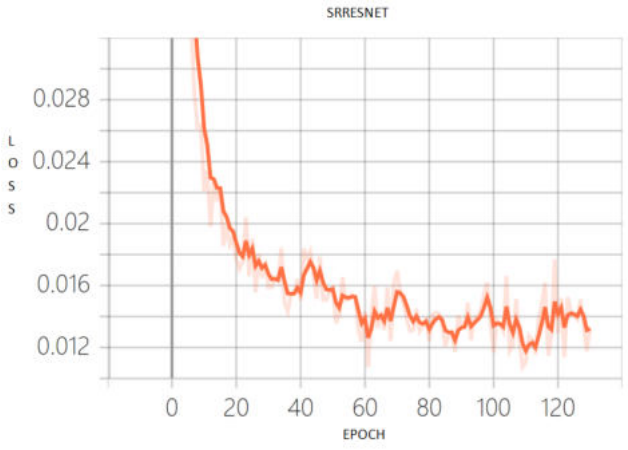

Fig. 12. SRRESNET training process

In the above multiplot, the horizontal coordinate is EPOCH, which is the number of training rounds. The vertical

TABLE III
PEAK SIGNAL TO NOISE RATIO (PSNR) COMPARISON

Test set	Magnification	DRRN(9)	DRRN(20)	DRRSN(9)	DRRSN(20)
Set5	×4	28.900dB	28.931dB	29.153dB	29.282dB
Set14	×4	26.503dB	26.532dB	26.613dB	26.758dB
BSD100	×4	26.266dB	26.285dB	26.357dB	26.436dB

TABLE IV
STRUCTURAL SIMILARITY (SSIM) COMPARISON

Test set	Magnification	DRRN(9)	DRRN(20)	DRRSN(9)	DRRSN(20)
Set5	×4	0.833	0.833	0.835	0.841
Set14	×4	0.744	0.745	0.743	0.749
BSD100	×4	0.713	0.713	0.710	0.716

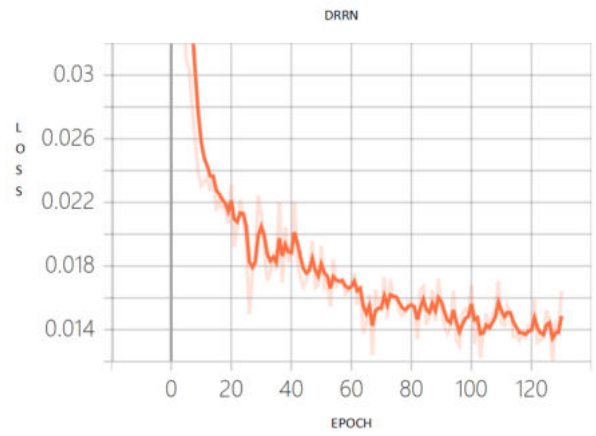

Fig. 13. DRRN training process

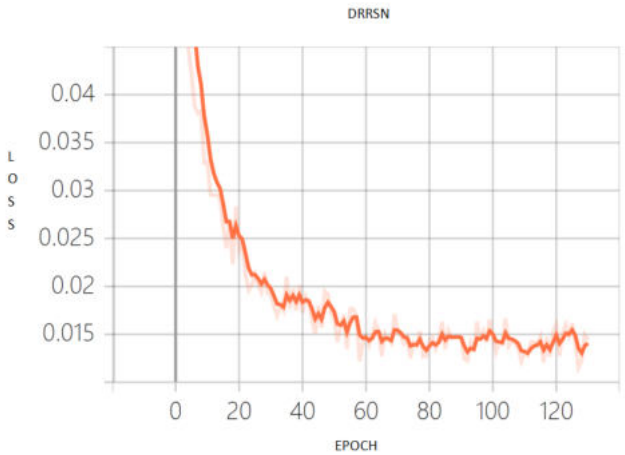

Fig. 14. DRRSN training process

coordinate is LOSS, which is the difference between the generated and real images. Comparing the multiple plots above shows that DRRSN is more stable when the network is trained. The efficiency of the DRRSN approach is demonstrated by the low LOSS variations compared to the significant LOSS fluctuations of the other three methods.

The stability of generative adversarial networks is mainly reflected in the discriminator network. The training fluctuation magnitude of the discriminator network directly affects the stability of the whole network. In this paper, the training process of the discriminator network of DRRSWGAN and SRGAN is compared. The training process of the discriminator network of DRRSWGAN is shown in Fig.16. The training process of the discriminator network of SRGAN is shown in Fig.17.

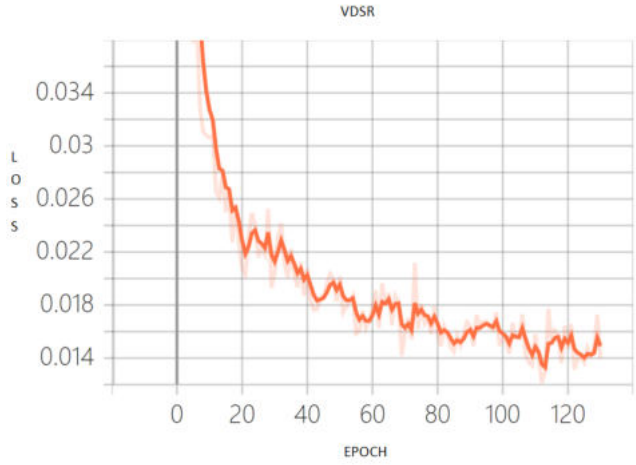

Fig. 15. VDSR training process

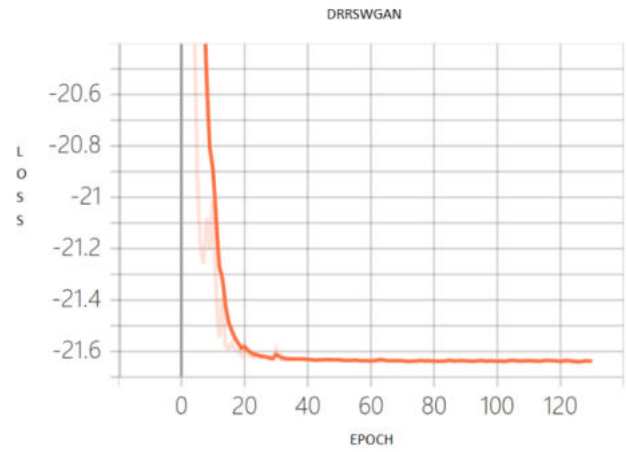

Fig. 16. DRRSWGAN training process

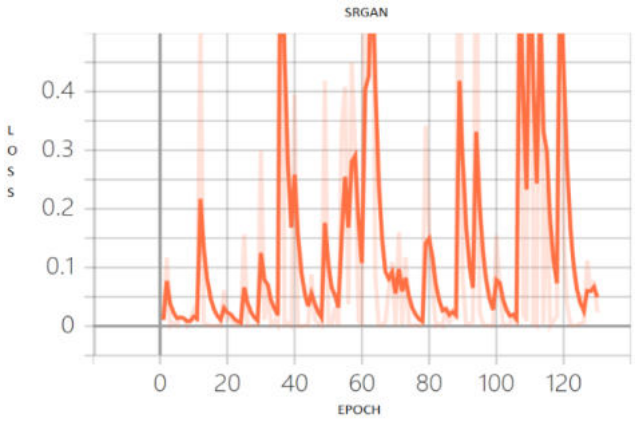

Fig. 17. SRGAN training process

In Fig.16 and Fig.17, the horizontal coordinate is EPOCH, which is the number of training rounds. The vertical coordinate is the LOSS, which is the disparity between the generated image and actual images. From Fig.17, the discriminator network training of SRGAN is very unstable and fluctuating. This fluctuation continues throughout the training. However, the discriminator network training of DRRSWGAN in Fig.16 is very stable and reaches convergence at close to 40 rounds of training. It should be noted that the LOSS values in the figure do not represent the final LOSS values of the generated images, because the LOSS in the figure is only the LOSS in the discriminator network.

3) Experimental results: DRRSN and DRRSWGAN are compared experimentally with VDSR, SRRSNET, DRRN, and SRGAN methods. The different methods were evaluated and compared on PSNR and SSIM of the reconstructed images in different test sets. The results of the PSNR comparison for an image magnification factor of 4 are shown in Table 5.

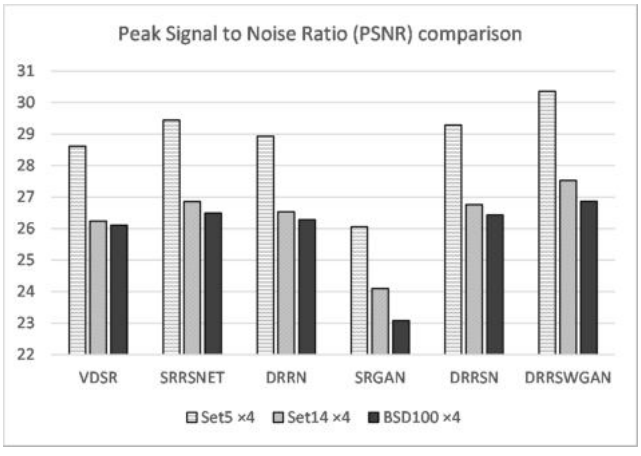

Fig. 18. Peak Signal to Noise Ratio (PSNR) comparison

Table 5 and Figure 18 display the peak signal-to-noise ratio (PSNR) comparative data results. With data obtained from the processing of the original algorithm, we compared the outcomes of the updated data algorithm. In the Set5×4 dataset, the DRRSWGAN algorithm achieves a PSNR value of 30.353dB, surpassing the maximum PSNR of existing algorithms and improving by 0.907dB compared to the SRRSNET algorithm. The DRRSWGAN algorithm produces PSNR values of 27.532dB and 26.862dB in the Set14 and BSD100 datasets, respectively, which significantly boost stability and the signal-to-noise ratio. The DRRSWGAN method's PSNR has been observed to be much higher in all three test sets when compared to other algorithms, proving the viability and benefits of our enhanced approach.

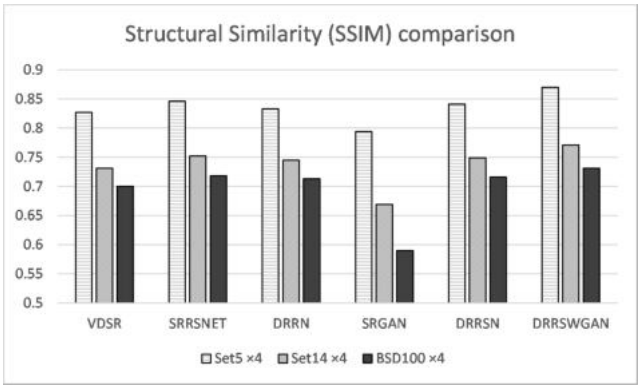

Fig. 19. Structural Similarity (SSIM) comparison

Findings from the structural similarity (SSIM) comparison data exhibited in Table 6 and Figure 19. We have compared the results of the improved data algorithm with the data obtained from the processing of the existing algorithm. In the Set5×4 dataset, the DRRSWGAN algorithm achieves an SSIM value of 0.870, which exceeds the maximum SSIM of currently available algorithms. The improvement over the previous algorithms DRRN and DRRSN are 0.037 and 0.029 respectively, which is a more significant improvement. In contrast, the DRRSWGAN algorithm achieves SSIM values of 0.771 and 0.731 in the Set14×4 dataset and the BSD100 dataset, which are significantly competitive in terms of structural similarity (SSIM) values. It is obvious that the SSIM of the DRRSWGAN algorithm is markedly enhanced in all three test sets compared to other algorithms, proving the feasibility and advantages of the DRRSWGAN algorithm.

The comparison of reconstructed image effects is also one of the evaluations of image super-resolution reconstruction methods. In this paper, images are selected in Set5, Set14, and BSD100, and the training the outcomes of several techniques at an image magnification factor of 4 are reconstructed in images.

A comparison of the reconstructed image effects is shown in Fig. 20 and Fig.21. In Fig.20, in the left image, the butterfly wing part is intercepted and enlarged, and the reconstructed image of this paper is clearer and richer in details compared with other methods; In the middle image, the ear part of the character is intercepted and enlarged, and the method of this paper has fewer artifacts and better subjective effect for human eyes; In the right image, the pink text part is intercepted and enlarged, and the method of this paper shows clearer letters compared with other algorithms.

The image on the left in Fig. 21 demonstrates how the coral portion of the image is intercepted and expanded to highlight that the reconstructed image of the DRRSWGAN also contains some artifacts, but they are significantly reduced and the image is more detailed when compared to other approaches.It can be observed that DRRSWGAN has a better reconstruction of the window border in the image on

TABLE V
PEAK SIGNAL TO NOISE RATIO (PSNR) COMPARISON

Test set	Magnification	VDSR	SRRSNET	DRRN	SRGAN	DRRSN	DRRSWGAN
Set5	×4	28.618dB	29.446dB	28.931dB	26.059dB	29.282dB	30.353dB
Set14	×4	26.242dB	26.854dB	26.532dB	24.100dB	26.758dB	27.532dB
BSD100	×4	26.113dB	26.495dB	26.285dB	23.077dB	26.436dB	26.862dB

TABLE VI
STRUCTURAL SIMILARITY (SSIM) COMPARISON

Test set	Magnification	VDSR	SRRSNET	DRRN	SRGAN	DRRSN	DRRSWGAN
Set5	×4	0.827	0.846	0.833	0.794	0.841	0.870
Set14	×4	0.731	0.752	0.745	0.669	0.749	0.771
BSD100	×4	0.700	0.718	0.713	0.590	0.716	0.731

Fig. 20. Reconstruction effect comparison(1)

the right, which intercepts and enlarges the window section on the right side of the image.

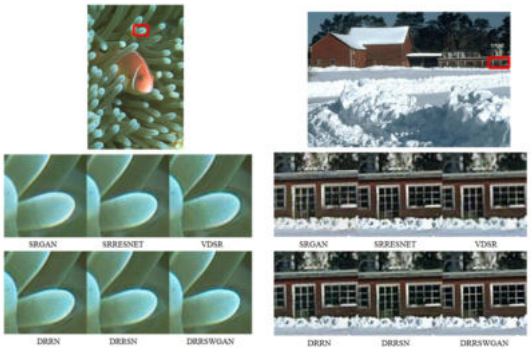

Fig. 21. Reconstruction effect comparison(2)

In summary, the DRRSN and DRRSWGAN methods recommended in this article have a more obvious improvement in comment metrics and image rendering effect compared with other algorithms. So, it can be known that our improvements are effective.

V. CONCLUSION

Images are affected by a variety of factors during the acquisition process and are susceptible to noise contamination during transmission, resulting in blurred images that fail to provide accurate information from the images. In this paper, we introduce two methods of image super-resolution reconstruction based on deep learning. Although these methods have achieved a series of results, there are still some problems: how to perform deep image feature extraction; how to fewer parameters are used in the deep network; and how to make the quality of the generated image better. To deal with these issues, this paper proposes the Deep Recursive Residual Subpixel Wasserstein Generative Adversarial Network (DRRSWGAN) and the main work is as follows.

(1) A deep recursive residual network (DRRN) is used in the generator network and sub-pixel convolution is introduced to form a deep recursive residual sub-pixel network (DRRSN). The BN layer within the residual unit is removed to protect the image information while extracting deep image features.

(2) The idea of WGAN is introduced in the discriminator network, and the loss function in the network is optimized. The gradient optimization of the network loss is performed using the RMSProp optimization algorithm to remove the Sigmod layer in the discriminator network. The parameter updates in the network are also truncated to improve the network training stability.

The outcomes of the trial indicate the reconstructed images of this paper method have been improved in PSNR and SSIM indexes in different degrees compared with other methods. The network training is more stable and the subjective effect of the reconstructed images is better. The approach used in this study provides benefits in different comparisons with other methods.

REFERENCES

[1] J. L. Harris, "Diffraction and resolving power," *JOSA*, vol. 54, no. 7, pp. 931–936, 1964.
[2] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 29, no. 6, pp. 1153–1160, 1981.
[3] W. T. Freeman, T. R. Jones, and E. C. Pasztor, "Example-based super-resolution," *Computer Graphics & Applications IEEE*, vol. 22, no. 2, pp. 56–65, 2002.
[4] H. Chang, D. Yeung, and Y. Xiong, "Super-resolution through neighbor embedding," in *Computer Vision and Pattern Recognition*, pp. 275–282, 2004.
[5] R. Timofte, V. De Smet, and L. Van Gool, "Anchored neighborhood regression for fast example-based super-resolution," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1920–1927, 2013.
[6] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European Conference on Computer Vision*, pp. 184–199, 2014.
[7] J. Kim, J. K. Lee, and K. M. Lee, "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.

[8] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 286–301, 2018.

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

[10] 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*, pp. 3147–3155, 2017.

[11] C. Ledig, L. Theis, F. Huszár, 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*, pp. 4681–4690, 2017.

[12] F. Nan, Q. Zeng, Y. Xing, and Y. Qian, "Single image super-resolution reconstruction based on the resnext network," *Multimedia Tools and Applications*, vol. 79, no. 45, pp. 34459–34470, 2020.

[13] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "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.

[14] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *International Conference on Machine Learning*, pp. 214–223, 2017.

[15] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, and C. C. Loy, "Esrgan: Enhanced super-resolution generative adversarial networks," in *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*, vol. 11133, pp. 63–79, 2018.

[16] M. S. Sajjadi, B. Scholkopf, and M. Hirsch, "Enhancenet: Single image super-resolution through automated texture synthesis," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4491–4500, 2017.

[17] E. Agustsson and R. Timofte, "Ntire 2017 challenge on single image super-resolution: Dataset and study," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 126–135, 2017.

[18] R. Timofte, E. Agustsson, L. Van Gool, M.-H. Yang, and L. Zhang, "Ntire 2017 challenge on single image super-resolution: Methods and results," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 114–125, 2017.

[19] M. Bevilacqua, A. Roumy, C. Guillemot, and M. Alberi-Morel, "Low-complexity single-image super-resolution based on nonnegative neighbor embedding," in *British Machine Vision Conference*, pp. 1–10, 2012.

[20] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International Conference on Curves and Surfaces*, pp. 711–730, 2010.

[21] 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 *Proceedings Eighth IEEE International Conference on Computer Vision*, vol. 2, pp. 416–423, 2001.

[22] J. Korhonen and J. You, "Peak signal-to-noise ratio revisited: Is simple beautiful?," in *2012 Fourth International Workshop on Quality of Multimedia Experience*, pp. 37–38, 2012.

[23] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.