ORIGINAL RESEARCH PAPER

Pre-training of gated convolution neural network for remote sensing image super-resolution

Yali Peng1,2 | Xuning Wang1,2 | Junwei Zhang1,2 | Shigang Liu1,2

1 Key Laboratory of Modern Teaching Technology, Ministry of Education, Xi’an 710062, China
2 School of Computer Science, Shaanxi Normal University, Xi’an 710119, China

Correspondence
Shigang Liu, Key Laboratory of Modern Teaching Technology, Ministry of Education, Xi’an 710062, China.
Email: shgliu@snnu.edu.cn

Abstract
Many very deep neural networks are proposed to obtain accurate super-resolution reconstruction of remote sensing images. However, the deeper the network for image SR is, the more difficult it is to train. The low-resolution inputs and features contain abundant low-frequency information and noise, which are treated equally as the high-frequency information across the network. To solve these problems, a novel single-image super-resolution algorithm named pre-training of gated convolution neural network (PGCNN) is proposed for remote sensing images. The proposed PGCNN consists of several residual blocks with long skip connections. Each residual block contains an additional well-designed gated convolution unit, which provides different weights to high-frequency information and low-frequency information to control the transmission of information, making the main network focus on learning high-frequency information. Compared with several state-of-the-art methods, experimental results on the remote sensing datasets (SIRI-WHU, NWPU-RESISC45, RSSCN7 and UC-Merced-Land-Use) show that the proposed PGCNN has the accuracy and visual improvements.

1 INTRODUCTION

At present, deep convolution neural networks can automatically learn advanced features from data, showing great potential in tasks such as image classification [1], speech recognition, face recognition [2–4], target detection, and natural image processing. And compared with the traditional algorithm, deep convolution neural networks show an overwhelming advantage to deal with super-resolution (SR) problems [5, 6]. Since the introduction of convolutional neural networks, more and more excellent variants of convolutional neural networks have been proposed one after another, and have achieved good results in their different applications, such as residual network [7–11], dense connection network [12], recurrent neural network [13], adversarial neural network [14]. In the past several years, some novel methods have also been applied to convolutional neural networks, including attention mechanisms [15–18], multi-supervision [19] etc. These methods and their variants based on convolutional neural networks are mainly used to improve performance and efficiency. It is worth noting that the performance and realization of the network are closely related to its excellent design. The use of skip links and attention mechanisms can effectively and efficiently design a small and efficient network. This provide the ability to design a network without many convolutional layers or large network of neural units, and greatly reduces the design and training costs of the network, hugely improves the performance and applicable value of the network.

2 RELATED WORK

Super-resolution (SR) is an image processing technology that recovers high-resolution images (HR) from single or continuous low-resolution images (LR). In recent years, massive state-of-the-art methods [20–25] based on traditional machine learning strategies are proposed to address the problems such as image enhancement, and these methods have great advantages in super-resolution reconstruction of satellite remote sensing images. In this section, we focus on developing advanced neural network-based methods. The SRCNN [26] proposed by Dong et al has been widely studied as a super-resolution film. Although SRCNN successfully introduced deep learning
techniques into the super-resolution (SR) problem, we found its limitations in three areas. Firstly, it relies on the context of small image regions. Secondly, training is slowly converging. Thirdly, the network is only suitable for a single size. Kim et al proposed a 20-layer CNN model called VDSR [27], which is a significant improvement over SRCNN. However, the VDSR style architecture requires bicubic interpolation images as input, resulting in a large amount of calculation time and memory. Originally, He et al proposed ResNet to solve high-level computer vision problems, such as image classification and detection. SRResNet [28] successfully solved those time and memory problems with good performance, but it did not draw on the advantages of the ResNet architecture. Subsequently, based on the SRResNet architecture, it was optimized through analysis and deletion. Lim and others [8] proposed EDSR and MDSR network architectures.

High-resolution images have richer details than low-resolution images and are suitable for remote sensing target detection and recognition. In recent years, super-resolution methods of single-image and multi-images in the field of remote sensing image processing have been widely proposed. Chavez-Roman and Ponomaryov [29] combined discrete wavelet transform and sparse representation to generate the high-resolution image from a single low-resolution image. Li et al. [30] explored sparse properties in both spectral and spatial domains for hyperspectral images super-resolution. Lei et al. [31] proposed a new single-image super-resolution algorithm named local-global combined networks for remote sensing images based on the deep CNNs.

Most of the above methods only consider parts of the image and ignore environmental information. Remote sensing images usually have higher levels of abstract data and complex spatial distribution, which is very different from natural images. And the incremental depth in the deep CNN framework will affect the ability to transfer feature information. Simple direct or skip connection algorithm based on CNN, poor performance when applied to remote sensing satellite image SR [32, 33]. And such strategies omit the difference between the noise and high-frequency information, not conducive to reconstructing high-quality remote sensing SR images. In order to mine omnidirectional and multi-scale remote sensing images, we use gated convolution layers to reconstruct the details of the images. In addition, we use a pre-training strategy based on transfer learning ideas to further improve the performance of the network. Compared with the most advanced reconstruction methods in recent years, the algorithm performs better in terms of peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) [34].

3 | PRE-TRAINING OF GATED CONVOLUTION NEURAL NETWORK

3.1 | Gated residual block

In the previous CNN-based residual networks, the learning results of the residual units were considered to be valid information, which made the network insufficiently universality in some real and complex scenarios, and caused the network’s learning ability to be interfered to some extent. In order to make the network focus more on high-frequency effective information, we designed a new type of residual block which make up of two convolution layers, a gated convolution layer and some nonlinear activation units named gated residual block (GRB). Figure 1 shows its basic structure. As gated convolution unit has a sigmoid function instead of a linear function, which will slightly increase the amount of calculation. Since the sigmoid function can constrain the output between 0 and 1, which means that the output value of the gated convolution unit represents the probability of a certain kind of data passing through a certain neural unit, and we then multiply this value with the original input, so it can restrict the passability of various data, greatly improving the sensitivity of the network to non-noise data. In order to make a compromise between performance and efficiency, we generally set the gated convolution kernel size to 1 × 1, and we generally set the number of feature extraction layers of each GRB unit to 2. With gated strategy, our gated residual unit can obtain a larger field of view and a higher fitting ability.

3.2 | Single-scale model

There are many ways to design a multi-scale super-resolution reconstruction network. However, in order to more directly verify the effectiveness of pre-training strategy and gated residual block, we have designed a neural network with a single scale to super-resolution reconstruction of remote sensing images. For convolutional networks, the performance of the convolutional network is mainly related to the capacity of the model. The capacity of the network is also called the fitting ability of the network. Generally speaking, the depth of the network and the width of the network are the core factors affecting the capacity of the network. In this case, more and more researchers tend to design a very large network to obtain more outstanding results. However, the huge network further increases the difficulty of training, and also increases the computing cost and time cost of the network, which is very unfriendly in practical applications.

Therefore, we tend to design a smaller and more efficient network. We use a total of 16 gated residual blocks in the network model. Each gated residual block unit consists of 2 feature extraction convolution layers and 1 gated convolution layer. Any one convolution layer contains only 64 neural units. At the end of the network, we added an upsampling layer which consists of a sub-pixel layer and a convolution layer. There are also some small tricks in the model designing, such as jumping links etc. The detailed structure of the network is shown in Figure 2.

Our pre-training of gated convolution neural network (PGCNN) mainly consists four parts: shallow feature extraction module, gated residual group (GRG) deep feature extraction module, upscale module, and reconstruction part. Assuming that the input of the network is \(I_{LR}\) and the output is \(I_{SR}\), the shallow feature extraction layer can be expressed by the following formula:

\[
F_1 = H_{SF}(I_{LR})
\]
where $H_{SF} (\cdot)$ stands for convolution operation, $F_0$ is then used as the input of the gated residual group, which can be expressed as:

$$F_{DF} = H_{GRG} (F_0)$$  \hspace{1cm} (2)$$

where $H_{GRG} (\cdot)$ represents the gated residual group operation, and $F_{DF}$ represents the deep feature information after the shallow feature information passes through the deep feature extraction module. Deep features will be converted into a super convolution feature map. Then, we have,

$$F_{UP} = H_{UP} (F_{DF})$$  \hspace{1cm} (3)$$

where $H_{UP} (\cdot)$ and $F_{UP}$ represent the up-sampling operation and the up-sampled feature map, respectively. These features will finally go through an image reconstruction module to convert the large-scale feature map into SR,

$$I_{SR} = H_{REC} (F_{UP})$$  \hspace{1cm} (4)$$

where $H_{REC} (\cdot)$ stands for image reconstruction module. In summary, PGCNN can be expressed by the following formula:

$$I_{SR} = H_{REC} (F_{UP}) = H_{PGCNN} (I_{LR})$$  \hspace{1cm} (5)$$

Based on the above discussion, we summarize the proposed algorithm as shown in Figure 3.

### 3.3 Up-scale functions

In order to convert LR image features into SR images features, the common strategy is increasing the pixel to match the SR image feature pixel count. There are many high efficiency methods such as nearest neighbour interpolation, bilinear interpolation, and bicubic interpolation in traditional machine learning field. But new up-sampling functions are proposed in recent years in deep learning field. Some model layers such as unpooling layer, convolution layer, deconvolution layer which are used to increase the resolution of LR images features.
FIGURE 3 PGCNN algorithm

Some researchers use pre-upsampling model architecture. They directly load the obtained same resolution with HR image dataset by using the common bicubic interpolation into the network. This is indeed more enhanced than just using bicubic interpolation. However, in subsequent research, some scholars have found that the utilization of post-upsampling is more conducive to better results.

In our network model, we utilize post-upsampling strategy and use a pixel shuffle layer to lift resolution of LR image features. Pixel shuffle is a method of pixel rearrangement. It rearranges the elements of the input feature imaging \( H \times W \times C \times r^2 \) tensor into \( rH \times rW \times C \) tensor (where \( H \) and \( W \) are the height and the width of the feature map, respectively, \( C \) represents the number of characteristic channels, and \( r \) is the magnification of the network).

### 3.4 Loss functions

Learning the end-to-end mapping model requires estimating a plurality of parameters. Constrain the reconstruction image loss with origin HR image is extremely important step for image super-resolution. How to set the loss function significantly affects the model performance. A well-designed loss function can greatly improve the training effect of the network and reduce the training time of the network. Most of the previous works minimize the pixel loss by using pixel-based l1-norm and l2-norm or some multiple loss combining methods. Using l2-norm as the loss function seems to make it easier for the network to achieve higher peak signal-to-noise ratios. However, MSE will excessively punish the specific differences caused by some features in the training data, which cannot make the network capture the details of the training data and high-frequency texture details, which often results in a network with high peak information noise ratio results, but it creates too smooth problems that are inconsistent with human vision. Therefore, in our experiments, we have adopted a simpler and straighter l1-norm as the loss function, which greatly reduces the adverse effect of artificial will on image super-resolution reconstruction. Moreover, this strategy is also very suitable for the problem of super-resolution reconstruction of satellite remote sensing images. It can more objectively restore a lot of realistic detailed information in satellite images and is more faithful to ground truth. But we do not preclude the usage of loss functions of the other formulations. The loss function can be expressed as:

\[
L(\theta) = \frac{1}{N} \sum_{i=1}^{N} \| H_{\text{PGCNN}} (I_{LR}^i) - I_{HR}^i \|_1
\]  

(6)

where \( N \) is the number of training samples, \( H_{\text{PGCNN}} (\cdot) \) represents the mapping function corresponding to PGCNN, \( I_{LR}^i \) and \( I_{HR}^i \) are the LR and HR of the \( i \)-th sample, respectively.

### 4 EXPERIMENTS

#### 4.1 Datasets

DOTA [35] is a newly proposed high-quality satellite image dataset which are mainly collected from Google Earth, satellite JL-1 and satellite GF-2 of the China Centre for Resources.
Satellite Data and Application. This dataset contains 2806 aerial images derived from different sensors and platforms. The image size varies ranging from roughly 800 $\times$ 800 to 4000 $\times$ 4000 pixels. This dataset is divided into three parts: training set, test set, and validation set. In our experiments, we merge the three parts together and select first 800 images as our training set, while the closely followed 60 images are used for validation, and additional 40 images are used in the testing phase.

We also provide four popular (SIRI-WHU [36–38], NWPU-RESISC45 [39], RSSCN7 [40] and UC-Merced [41]) scene classification remote sensing datasets and extract the first one as the test set in each scene of each dataset as the supplementary test dataset. All of the above LR images are obtained by downsampling the original images with a bicubic kernel.

### 4.2 Training details

In training, we use 48 $\times$ 48 picture blocks as data, and use the normalization algorithm to preprocess the DOTA dataset. In each batch, the average RGB value of the input data is subtracted to ensure that the network is more sensitive to the details of the dataset. We randomly use strategies such as horizontal flip, vertical flip, offset, and channel changes to enhance training data to ensure the efficiency and reliability of network training.

The training process is mainly divided into two stages. Firstly, the DIV2K [5] dataset is used to train the network 3 $\times$ 10^5 iterations, we set the initial learning rate to $1 \times 10^{-4}$ and use a strategy that automatically adjusts during the training process to halve the learning rate every 2 $\times$ 10^3 iterations of back propagation, and then the trained model is trained with the DOTA dataset for 6 $\times$ 10^7 iterations, and the learning rate is initialized to 5 $\times$ 10^{-5}. As usual, we use ADAM [42] as the optimizer and set its parameters $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1 \times 10^{-8}$. Our model is implemented on Pytorch with Python3.6 under Linux systems, and used the NVIDIA Tesla V100 graphics processing unit. The training took about 8 h for each up-scaling factor.

### 4.3 Evaluation on DOTA dataset

We compared PGCNN with several classic methods such as SRCNN, SRResNet, and EDSR benchmark called EDSR- in recent years. At the same time, we use the traditional interpolation-based method bicubic as the test benchmark. We use PSNR and SSIM as the evaluation indicators of the test and all tests are conducted in the RGB colour space.

Table 1 shows the experimental results of various methods on the DOTA test set. It can be seen that, compared to the bicubic benchmark, all deep learning-based methods are significantly ahead. On scale $\times 2$, $\times 3$ and $\times 4$, our proposed method has a great advantage compared to the results of SRCNN and SRResNet methods. Compared with EDSR-, our method can still maintain a large advantage. The bold font indicates the best performance.

To further verify the visual superiority of PGCNN’s results, in Figures 4 and 5, we show visual comparisons on scale $\times 4$. The best results are bolded. From the test results of image P0866, we can see that all the methods involved in the comparison are incomplete reconstruct the lines and texture of the runway clearly. On the contrary, our method can reduce the blurry artificial traces and correctly reconstruct more lines and texture information. Similarly, as can be seen from the results of several other images, our method can more and more clearly reconstruct the texture information of the image. This further illustrates the effectiveness of our gating residual unit and the superiority of PGCNN.

### 4.4 Evaluation on multiple datasets

We also compared with other competitive methods on SIRI-WHU, NWPU-RESISC45, RSSCN7 and UC-Merced-Land-Use test datasets. Tables 2, 3 and 4 show the comparison data under scale $2 \times$, scale $3 \times$ and scale $4 \times$, respectively. The results show that the quantitative results obtained by our super-resolution method still have greater advantages, which also proves that our method has good migration and general. It can demonstrate the excellent performance by exploring the detailed information in LR images and reconstructing better SR results. The bold font indicates the best performance.

### 4.5 Effects of pre-training

Since our work uses two more important strategies to enhance the performance of the network, in order to explore the specific impact of different strategies on the experiment, we further conduct corresponding experiments to illustrate the impact of the application of different strategies on the network performance.
FIGURE 4  Visual comparison for 4× SR on P0866 and P0881 of DOTA datasets

TABLE 2  Average results of PSNR (dB), SSIM on the SIRI-WHU, NWPU-RESISC45, RSSCN7 and UC-Merced test datasets with scale factor of 2

| Method   | Scale | SIRI-WHU PSNR | SIRI-WHU SSIM | NWPU-RESISC45 PSNR | NWPU-RESISC45 SSIM | RSSCN7 PSNR | RSSCN7 SSIM | UC-Merced PSNR | UC-Merced SSIM |
|----------|-------|---------------|---------------|--------------------|--------------------|-------------|-------------|----------------|----------------|
| Bicubic  | x2    | 32.709        | 0.903         | 30.066             | 0.877              | 27.824      | 0.858       | 32.869         | 0.916          |
| SRCNN    | x2    | 34.397        | 0.932         | 31.811             | 0.915              | 29.547      | 0.902       | 35.237         | 0.942          |
| SRResNet | x2    | 34.692        | 0.935         | 32.357             | 0.921              | 30.142      | 0.910       | 35.610         | 0.946          |
| EDSR     | x2    | 34.830        | 0.937         | 32.423             | 0.917              | 30.199      | 0.912       | 35.690         | 0.945          |
| PGCNN    | x2    | 34.886        | 0.937         | 32.500             | 0.925              | 30.274      | 0.913       | 35.859         | 0.947          |
FIGURE 5 Visual comparison for 4x SR on P0884 and P0886 of DOTA datasets

TABLE 3 Average results of PSNR (dB), SSIM on the SIRI-WHU, NWPU-RESISC45, RSSCN7 and UC-Merced test datasets with scale factor of 3

| Method   | Scale | SIRI-WHU  | NWPU-RESISC45 | RSSCN7  | UC-Merced |
|----------|-------|-----------|---------------|---------|-----------|
|          |       | PSNR      | SSIM          | PSNR    | SSIM      | PSNR      | SSIM      | PSNR      | SSIM      |
| Bicubic  | x3    | 29.682    | 0.810         | 27.272  | 0.776     | 25.050    | 0.800     | 28.621    | 0.810     |
| SRCNN    | x3    | 30.997    | 0.852         | 28.472  | 0.818     | 26.105    | 0.788     | 30.241    | 0.857     |
| SRResNet | x3    | 31.368    | 0.860         | 28.835  | 0.829     | 26.413    | 0.800     | 30.724    | 0.870     |
| EDSR     | x3    | 31.414    | 0.861         | 28.888  | 0.831     | 26.461    | 0.803     | 30.716    | 0.868     |
| PGCNN    | x3    | 31.440    | 0.862         | 28.991  | 0.834     | 26.587    | 0.808     | 30.890    | 0.872     |
### TABLE 4
Average results of PSNR (dB), SSIM on the SIRI-WHU, NWPU-RESISC45, RSSCN7 and UC-Merced test datasets with scale factor of 4

| Method   | Scale | SIRI-WHU PSNR | SIRI-WHU SSIM | NWPU-RESISC45 PSNR | NWPU-RESISC45 SSIM | RSSCN7 PSNR | RSSCN7 SSIM | UC-Merced PSNR | UC-Merced SSIM |
|----------|-------|---------------|---------------|--------------------|--------------------|-------------|-------------|----------------|----------------|
| Bicubic  | x4    | 28.055        | 0.727         | 25.772             | 0.675              | 23.707      | 0.623       | 26.334         | 0.710          |
| SRCNN    | x4    | 29.119        | 0.775         | 26.734             | 0.731              | 24.401      | 0.688       | 27.632         | 0.761          |
| SRResNet | x4    | 29.549        | 0.790         | 27.125             | 0.749              | 24.784      | 0.711       | 28.166         | 0.775          |
| EDSR     | x4    | 29.574        | 0.790         | 27.118             | 0.750              | 24.755      | 0.711       | 28.121         | 0.777          |
| PGCNN    | x4    | 29.600        | 0.791         | 27.242             | 0.754              | 24.932      | 0.719       | 28.312         | 0.782          |

### TABLE 5
Investigations of pre-training. The best average results of PSNR (dB), SSIM on the SIRI-WHU, NWPU-RESISC45, RSSCN7 and UC-Merced test datasets are observed

| Method | Pre-training | Scale | SIRI-WHU PSNR | SIRI-WHU SSIM | NWPU-RESISC45 PSNR | NWPU-RESISC45 SSIM | RSSCN7 PSNR | RSSCN7 SSIM | UC-Merced PSNR | UC-Merced SSIM |
|--------|--------------|-------|---------------|---------------|--------------------|--------------------|-------------|-------------|----------------|----------------|
| PGCNN  | No           | x2    | 34.830        | 0.937         | 32.423             | 0.917              | 30.199      | 0.912       | 35.690         | 0.945          |
|        | Yes          | x2    | 34.832        | 0.937         | 32.448             | 0.924              | 30.231      | 0.912       | 35.832         | 0.948          |
|        | No           | x3    | 31.414        | 0.861         | 28.888             | 0.831              | 26.461      | 0.803       | 30.716         | 0.868          |
|        | Yes          | x3    | 31.447        | 0.862         | 28.982             | 0.834              | 26.586      | 0.807       | 30.863         | 0.869          |
|        | No           | x4    | 29.574        | 0.790         | 27.118             | 0.750              | 24.755      | 0.711       | 28.121         | 0.777          |
|        | Yes          | x4    | 29.683        | 0.793         | 27.193             | 0.753              | 24.900      | 0.719       | 28.242         | 0.779          |

In the test of the pre-training strategy, we only use the ordinary residual convolution without gating mechanism. The number of residual blocks and the network model structure are consistent with the main experiment in this paper. On this basis, we first use the DIV2K dataset to train the network with $3 \times 10^5$ iteration cycles, where the learning rate is initialized to $1 \times 10^{-4}$, and then the trained network is fine-tuned again using the DOTA dataset, and the learning rate is initialized to $5 \times 10^{-5}$, the experimental results are shown in Table 5. By comparing the results of the network that has not been pre-trained with the DIV2K dataset, it can be seen that the pre-trained network shows better performance after being replaced with the main training dataset for fine-tuning, which also further shows the importance of transfer learning strategy in remote sensing images super-resolution reconstruction.

### 4.6 Effects of using gated convolution

We further demonstrate the impact of adding a gating module to the residual unit on the overall network performance. It is worth noting that in this test, we will not use the DIV2K dataset for pre-training on the network, only the DOTA data for training. The results of the experiment are shown in Table 6.

By comparing the experimental results in the table, we find that the network performance of the added gating module

### TABLE 6
Investigations of using gated convolution. The best average results of PSNR (dB), SSIM on the SIRI-WHU, NWPU-RESISC45, RSSCN7 and UC-Merced test datasets are observed

| Method | With gate module | Scale | SIRI-WHU PSNR | SIRI-WHU SSIM | NWPU-RESISC45 PSNR | NWPU-RESISC45 SSIM | RSSCN7 PSNR | RSSCN7 SSIM | UC-Merced PSNR | UC-Merced SSIM |
|--------|------------------|-------|---------------|---------------|--------------------|--------------------|-------------|-------------|----------------|----------------|
| PGCNN  | No               | x2    | 34.830        | 0.937         | 32.423             | 0.917              | 30.199      | 0.912       | 35.690         | 0.945          |
|        | Yes              | x2    | 34.829        | 0.937         | 32.443             | 0.924              | 30.232      | 0.912       | 35.732         | 0.946          |
|        | No               | x3    | 31.414        | 0.861         | 28.888             | 0.831              | 26.461      | 0.803       | 30.716         | 0.868          |
|        | Yes              | x3    | 31.449        | 0.862         | 28.988             | 0.834              | 26.586      | 0.807       | 30.863         | 0.869          |
|        | No               | x4    | 29.574        | 0.790         | 27.118             | 0.750              | 24.755      | 0.711       | 28.121         | 0.777          |
|        | Yes              | x4    | 29.600        | 0.791         | 27.197             | 0.753              | 24.884      | 0.718       | 28.171         | 0.779          |
performs generally on a small portion of the dataset with low magnification, but in most cases, the gating module shows good results and further enhance the overall performance of the network.

4.7 | Model size analyses

The size of the network model is an important factor in practical applications. We show the comparison of model size and model-related parameters in Table 7. For fair comparison, all the models participating in the comparison use the same number of residual units, and the number of neural units in all convolutional layers is mostly kept at 64.

It can be seen that our PGCNN has a very small amount of parameters, which is slightly more than EDSR-. The reason is that our residual unit adds the GATE module, which brings a small amount of additional parameters. Because the residual unit used by SRResNet contains a batch normalization layer and an additional convolutional layer, its parameter amount is slightly larger than that of PGCNN and EDSR. Overall, our PGCNN has a smaller amount of parameters, and a better trade-off between model size and performance. This also shows that the gated residual unit is desirable in practical applications.

5 | CONCLUSIONS

This paper proposes a new form of convolutional residual unit called gated residual block, and using this residual unit, we design a single-scale super-resolution neural network model for remote sensing images. We firstly use the DIV2K dataset to pre-train the network, and then the DOTA dataset to fine-tune the trained network again. In order to verify the reconstruction effectiveness of gated residual unit and the pre-training strategy for feature migration, we designed two different experiments to show the effect of different designs and training strategies on the results. At the same time, we also used some other public remote sensing datasets such as SIRI-WHU, NWPU-RESISC45, RSSCN7 and UC-Merced-Land-Use to test the quality of the super-resolution reconstruction of the model we built. Compared with some excellent neural network super-resolution models in recent years, our method can reconstruct more texture and detail information, and it also shows good results on other datasets. Our method shows good robustness further demonstrates the superiority and effectiveness of our proposed gating residual unit and feature migration pre-training strategy.

ORCID
Yali Peng https://orcid.org/0000-0002-0055-3081
Shigang Liu https://orcid.org/0000-0003-0162-3595

REFERENCES

1. Liu, S., et al.: Virtual images inspired consolidate collaborative representation-based classification method for face recognition. J. Mod. Opt. 63(12), 1181–1188 (2016)
2. Peng, Y., et al.: Space–Frequency domain based joint dictionary learning and collaborative representation for face recognition. Signal Process. 147, 101–109 (2018)
3. Liu, S., et al.: A novel label learning algorithm for face recognition. Signal Process. 124, 141–146 (2016)
4. He, K., et al.: Regularized negative label relaxation least squares regression for face recognition. Neural Process. Lett. 1–19 (2020)
5. Timofte, R., et al.: Ntire 2017 challenge on single image super-resolution: Methods and results. In: Proceedings of the 30th IEEE Conference on Computer Vision and Pattern Recognition Workshops. Honolulu, pp. 1110–1121 (2017)
6. Glasner, D., Bagon, S., Irani, M.: Super-resolution from a single image. In: 2009 IEEE 12th International Conference on Computer Vision. Kyoto, Japan, pp. 349–356 (2009)
7. Lu, Z., et al.: Fast single image super-resolution via dilated residual networks. IEEE Access 7, 109729–109738 (2018)
8. Lim, B., et al.: Enhanced deep residual networks for single image super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. Honolulu, pp. 1132–1140 (2017)
9. Li, S., et al.: Fast spatio-temporal residual network for video super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Long Beach, CA (2019)
10. Lan, M., et al.: Global context based automatic road segmentation via dilated convolutional neural network. Inf. Sci. 535, 156–171 (2020)
11. He, K., et al.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, pp. 770–778 (2016)
12. Huang, G., et al.: Densely connected convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Honolulu, HI, pp. 2261–2269 (2017)
13. Mikolov, T., et al.: Recurrent neural network based language model. In: Eleventh Annual Conference of the International Speech Communication Association. Makuhari, Japan, pp. 1045–1048 (2010)
14. Ajakan, H., et al.: Domain-adversarial neural networks. J. Stat. 1050 (2014)
15. Zhang, Y., et al.: Image super-resolution using very deep residual channel attention networks. In: Ferrari V., Hebert M., Sminchisescu C., Weiss Y. (eds.) Proceedings of the European Conference on Computer Vision (ECCV), Munich, September 2018. Lecture Notes in Computer Science, vol. 11111, pp. 294–310. Springer, Cham (2018)
16. Wang, N., et al.: Multistage attention network for image inpainting. Pattern Recognit. 106, 107448 (2020)
17. Wang, N., et al.: Musical: Multi-scale image contextual attention learning for inpainting. In: International Joint Conference on Artificial Intelligence. Macao, China, pp. 3748–3754 (2019)
18. Vaswani, A., et al.: Attention is all you need. In: Advances in neural information processing systems. Long Beach, CA, pp. 5998–6008 (2017)
19. Ronneberger, O., et al.: U-Net: Convolutional networks for biomedical image segmentation. In: Navab N., Hornegger J., Wells W., Frangi A. (eds.) Proceedings of the European Conference on Computer Vision (ECCV), Munich, September 2018. Lecture Notes in Computer Science, vol. 9901, pp. 234–244. Springer, Cham (2018)
20. Liu, H., et al.: Crowd evacuation simulation approach based on navigation knowledge and two-layer control mechanism. Inf. Sci. 436–437, 247–267 (2018)
21. Liu, Z., et al.: Discriminative low-rank preserving projection for dimensionality reduction. Appl. Soft Comput. 85, 105768 (2019)
22. Liu, S., Peng, Y.: A local region-based chan–vese model for image segmentation. Pattern Recognit. 45(7), 2769–2779 (2012)
23. Liu, H., et al.: A path planning approach for crowd evacuation in buildings based on improved artificial bee colony algorithm. Appl. Soft Comput. 68, 360–376 (2018)
24. Liu, Z., et al.: Structured optimal graph based sparse feature extraction for semi-supervised learning. Signal Process. 170, 107456 (2020)
25. Song, L., et al.: Unsupervised domain adaptive re-identification: Theory and practice. Pattern Recognit. 102, 107173 (2020)
26. Dong, C., et al.: Learning a deep convolutional network for image super-resolution. In: Fleet D., Pajdla T., Schiele B., Tuytelaars T. (eds.) European Conference on Computer Vision, Zurich, September 2014. Lecture Notes in Computer Science, vol. 8692, pp. 184–199. Springer, Cham (2014)
27. Kim, J., Kwon Lee, J., Lee, K.: Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Las Vegas, NV, pp. 1646-1654 (2016)
28. Ledig, C., 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. Honolulu, HI, pp. 105–114 (2017)
29. Chavez-Roman, H., Ponomaryov, V.: Super resolution image generation using wavelet domain interpolation with edge extraction via a sparse representation. IEEE Geosci. Remote Sens. Lett. 11(10), 1777–1781 (2014)
30. Li, J., et al.: Hyperspectral image super-resolution by spectral mixture analysis and spatial–spectral group sparsity. IEEE Geosci. Remote Sens. Lett. 13(9), 1250–1254 (2016)
31. Lei, S., Shi, Z., Zou, Z.: Super-resolution for remote sensing images via local–global combined network. IEEE Geosci. Remote Sens. Lett. 14(8), 1243–1247 (2017)
32. Zhang, L., Zhang, L., Du, B.: Deep learning for remote sensing data: A technical tutorial on the state of the art. IEEE Geosci. Remote Sens. Mag. 4(2), 22–40 (2016)
33. Zhang, J., et al.: Satellite image super-resolution based on progressive residual deep neural network. J. Appl. Remote Sens. 14(3), 1 (2020)
34. Wang, Z., et al.: Image quality assessment: From error visibility to structural similarity. IEEE Trans. Image Process. 13(4), 600–612 (2004)
35. Xia, G.-S., et al.: Dota: A large-scale dataset for object detection in aerial images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. Salt Lake City, UT, pp. 3974–3983 (2018)
36. Zhu, Q., et al.: Bag-of-visual-words scene classifier with local and global features for high spatial resolution remote sensing imagery. IEEE Geosci. Remote Sens. Lett. 13(6), 747–751 (2016)
37. Zhao, B., et al.: The fisher kernel coding framework for high spatial resolution scene classification. Remote Sens. 8(2), 157 (2016)
38. Zhao, B., et al.: Dirichlet-derived multiple topic scene classification model for high spatial resolution remote sensing imagery. IEEE Trans. Geosci. Remote Sens. 54(4), 2108–2123 (2015)
39. Cheng, G., Han, J., Lu, X.: Remote sensing image scene classification: Benchmark and state of the art. Proc. IEEE 105(10), 1865–1883 (2017)
40. Zou, Q., et al.: Deep learning based feature selection for remote sensing scene classification. IEEE Geosci. Remote Sens. Lett. 12(11), 2321–2325 (2015)
41. Yang, Y., Newssam, S.: Bag-of-visual-words and spatial extensions for land-use classification. In: Proceedings of the 18th SIGSPATIAL International Conference on Advances in Geographic Information Systems. San Jose, CA, pp. 270–279 (2010)
42. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. Computer Science, (2014)