Research on Image Super-resolution Reconstruction Method Based on Improved SRCNN

Zhongcai-Huo, Zhongdong-Wu*, Weifu-Xu

1School of Electronic and Information Engineering, Lanzhou Jiaotong University, Lanzhou, Gansu, 730070, China

*Corresponding author’s e-mail: WUZHD@mail.lzjtu.cn

Abstract. Existing image convolutional neural network based on image super-resolution algorithm has a problem of image texture blurred to improve. In this paper, we first analyze the factors of reconstructed image quality, then use the parametric rectified linear unit (PRLU) to solve the problem of over-compression in the original network. And combining the existing network models and image processing algorithms adjusts the parameters of the neural network. The network model of 9-1-5 in Super-Resolution Convolutional Neural Network (SRCNN) is improved to a three-layer model of 5-3-5. The number of convolution cores in the first and second layers is adjusted to 32, and the magnification is increased by four times from three times. Finally, simulation experiments are carried out on DIV2K dataset in NTIRE2017. Experimental results show that the proposed algorithm achieves good super-resolution results, and the subjective visual effect and objective evaluation indices are both improved obviously.

The image super-resolution technology has great application demand in the fields of video surveillance, virtual scene restoration and so on. There is still a great research prospect for the image super-resolution technology.

1. Introduction

Image super-resolution [1] is a technology for generating high-resolution images with many details and clear details corresponding to low-resolution images with poor quality [2,3]. It has great application prospect and application demand in computer vision and other fields that need to provide high-resolution images. Currently, commonly used methods are generally divided into interpolation method, reconstruction method and learning method. Among them, the method of super-resolution reconstruction using interpolation method has the advantages of low algorithm complexity and fast reconstruction speed that other methods do not have, but the image reconstruction effect is not good. Although the ultra-resolution reconstruction method can reconstruct the image with clear edges, the image after reconstruction is poor in detail richness. With the development of the artificial neural network, deep learning method has the advantage of richer and truer image details after reconstruction in the field of image processing. By learning the feature relations between two groups of images with the same content but different resolution, and applying the feature information obtained through learning to the reconstruction process, high-resolution images with rich details can be reconstructed [4].

This article by Changsheng Hu and other literature [5,6], aims to modify the Network structure, join the pooling level and adjust the Network parameters. SRCNN is used in the 9-1-5 Network model, the improved Network for the 5-3-5 three-layer model, and adjusts the size to improve performance of each layer of convolution kernels set $n_1 = 32$, $n_2 = 32$, and magnifies of super-resolution images from 3
times increased to 4 times, consisting of a series of experimental results. It has shown that the improved algorithm achieves better super-resolution effect.

2. SRCNN algorithm improvement

2.1. The convolution layer
For the algorithm model of the convolutional neural network, the key point is to consider the effect of convolution kernel size, quantity and network learning rate on model efficiency and effect.

2.1.1. A subsubsection.
The network structure used in SRCNN is 9-1-5. In order to balance the performance and speed, the improved network is a three-layer model of 5-3-5, that is the convolution kernel size of the three convolutional layers of the network is \( f_1 = 5, f_2 = 3, f_3 = 5 \) respectively.

In primitive SRCNN network, the first layer of convolution kernels size 9x9, but using a larger convolution check image matrix convolution operation will be extracted to a large number of dense and similar characteristics, parameters caused by redundancy, through experimental verification shows that the first layer using the size of a 5x5 convolution kernels is enough for the cover image characteristics, at the same time in the first layer network using 5x5 convolution kernels can significantly reduce the number of parameters, and improve the efficiency of the network.

The size of the second layer convolution kernel determines the effect of image super-resolution reconstruction. The larger the convolution kernel is, the better the reconstruction effect will be. but at the same time, the computational amount will also increase. In order to improve the super-resolution effect of the network without making the computational amount too large, the size of the second layer convolution kernel is chosen to be 3x3.

For the third layer network, reducing the size of the convolution kernel of the third layer will affect the image quality after reconstruction, but increasing the convolution kernel of the third layer will increase the computation amount. Considering the calculation and reconstruction quality, the 5-3-5 model is finally selected.

2.1.2. Convolution kernel number
In SRCNN’s 9-1-5 model, \( n_1 = 64, n_2 = 32 \). The improved algorithm tests the sum of different sizes under the 5-3-5 model, and finally selects the convolution kernel number \( n_1 = 32, n_2 = 32 \). For the 5-3-5 model of the improved algorithm, the SET5 test set is tested with the same learning rate and different convolution kernel number. Taking Butterfly as an example, the learning rate is \( 10^{-4} \), and the peak signal-to-noise ratio of the comparison image is obtained, as shown in table 1.

| Convolution kernel number | PSNR     |
|---------------------------|----------|
| \( n_1 = 64 \), \( n_2 = 32 \) | 27.94dB   |
| \( n_1 = 32 \), \( n_2 = 32 \) | **28.10dB** |
| \( n_1 = 32 \), \( n_2 = 64 \) | 27.57dB   |

2.1.3. Vector
The weight should be updated constantly during network training. If the learning rate is too small, then the slow gradient updating will lead to slow convergence, and if the learning rate is too high, the loss function will oscillate. SRCNN has a small learning rate of \( 10^{-5} \) in the third layer, and \( 10^{-4} \) in the other two layers. Therefore, the initial learning rate was uniformly set as \( 10^{-4} \) in the experiment.
2.2. The activation function
In the original SRCNN, linear correction unit is used as the activation function. It is easy to see from the function expression that when the extracted image feature points fall into the first quadrant, they can be recognized and extracted by the activation function, but the feature points in other quadrants cannot be recognized. An improved linear correction unit with parameters is proposed.

The expression of linear correction unit is as follows:

\[ f(x) = \begin{cases} 
  x, & x > 0 \\
  0, & x < 0 
\end{cases} \quad (1) \]

Compared with the linear correction unit, the linear correction unit with parameters cannot only have the same characteristics in the \( x > 0 \) part, but also retain some useful information in the \( x < 0 \) part to further improve the reconstruction quality. Therefore, linear correction units with parameters are used instead of linear correction units.

For the linear correction unit with parameters, its expression is written as follows for the convenience of writing and comparison:

\[ f(x) = \begin{cases} 
  x, & x > 0 \\
  cx, & x < 0 
\end{cases} \quad (2) \]

\( c \) is the slope of the negative part. Make the two functions consistent in \( x > 0 \) by setting, and the function value is \( x \) itself.

2.3. Pooling layer
There is no pooling layer in the original SRCNN, and the pooling layer is added after the convolution of the first layer and the second layer. Since the essence of pooling layer can also be regarded as a 2×2 convolution layer, the purpose of reducing the output feature vector of the convolution layer, reducing the dimension and improving the training speed can be achieved. Meanwhile, after the pooling layer is added, the network depth reaches to five layers, and the deeper network layer is conducive to the learning of image expression. Using maximum pooling, you can output the maximum value of the sub-block of the image matrix.

3. Super-resolution reconstruction method based on improved SRCNN

3.1. Image extension
Due to the interpolation method will blur the image and cover the detailed information, it can achieve the effect of image preprocessing. Therefore, the double-cubic interpolation method is used to expand the image of the training set. Although the size of the expanded image is consistent with the real high resolution, the detailed features are covered, which meets the requirements of the experiment on low-resolution images. The schematic diagram of image expansion is shown in figure 1.

3.2. Feature extraction
In order to get the feature graph composed of the edge and detail features of the input image, it is necessary to extract the image blocks from the low-resolution image. In the original SRCNN network, the operation of the first layer is expressed in the following form:
\[ F_i(X) = \max(0, W_i \ast X + B_i) \]  
\[ F_i(X) = \max(0, W_i \ast X + B_i) + \min(0, W_i \ast X + B_i) \]  
(3)  
(4)

In the improved network, the feature extraction layer is modified to:

\[ F_2(X) = \max(0, W_2 \ast F_1(X) + B_2) \]  
\[ F_2(X) = \max(0, W_2 \ast F_1(X) + B_2) + \min(0, W_2 \ast F_1(X) + B_2) \]  
(5)  
(6)

Where \( X \) is the original image, \( W_1 \) is the convolution kernel, \( B_1 \) is the deviation, \( \ast \) is the convolution operation, \( c \) is the number of channels in the input image, \( f_1 \) is the size of a single filter, \( n_1 \) is the number of convolution kernel, and \( c_1 \times f_1 \times n_1 \) is the size of \( W_1 \).

3.3. Feature mapping

Feature mapping is to map feature vectors from low-resolution space to corresponding high-resolution space. After image feature extraction in the first layer network, \( n_1 \) dimension feature vectors are obtained. The second layer network maps the \( n_1 \) dimension eigenvectors obtained from the first layer to the \( n_2 \) dimension vectors. The output of the first layer is the input of the second layer, so the operation of the second layer can be expressed as:

\[ F_2(X) = \max(0, W_2 \ast F_1(X) + B_2) \]  
\[ F_2(X) = \max(0, W_2 \ast F_1(X) + B_2) + \min(0, W_2 \ast F_1(X) + B_2) \]  
(7)

3.4. Image reconstruction

Image reconstruction is carried out by using the image blocks of the first two layers as features to produce the final high-resolution image.

\[ F_3(X) = \max(0, W_3 \ast F_2(X) + B_3) \]  
\[ F_3(X) = \max(0, W_3 \ast F_2(X) + B_3) + \min(0, W_3 \ast F_2(X) + B_3) \]  
(8)

Where, \( W_3 \) is a linear filter with a size of \( n_2 \times f_2 \times f_2 \times c \) and \( B_3 \) is the deviation.

3.5. Network parameter estimation and optimization

Network parameters \( \theta = \{ W_1, W_2, W_3, B_1, B_2, B_3 \} \) use mean square error (MSE) as loss function to be optimized. In fact, the training process of network is the process of continuous optimization of network parameters. The form of MSE is as follows:

\[ L(\theta) = \frac{1}{n} \sum_{j=1}^{n} || F(Y_j, \theta) - X_j ||^2 \]  

The optimal solution of parameter \( \theta \) is obtained by minimizing the difference between \( F(Y_j, \theta) \) and \( X_j \). Where, \( n \) is the number of training sets, \( \{ Y_j \} \) is the set of high-resolution image blocks, and \( \{ X_j \} \) is the set of corresponding low-resolution image blocks.

The purpose of using a gradient descent method to optimize parameters and seek the optimal solution is to minimize the loss function. Even if the parameters advance along the direction of gradient descent by a step, they will find the point where the gradient gradually stabilizes and no longer changes. Here, the step size is the learning rate. Currently, the most commonly used gradient descent methods are batch gradient descent method, random gradient descent method and small batch gradient descent method. When updating parameters, batch gradient descent method uses all samples.
for calculation. When the sample data is large, its convergence efficiency is very low. The stochastic gradient descent method is to find a sample for gradient updating, so it may not update in the right direction every time. Although a better local optimal solution or even the global optimal solution may be found in the end, the problem of loss function oscillation may occur in this method. As a compromise of the above two methods, the low-batch gradient descent method has certain stability but still cannot eliminate the problem of oscillation.

To solve the above problems, this paper aims to use the gradient descent method with impulse to improve the stability of the loss function when updating parameters with the stochastic gradient descent method, and at the same time, another advantage of reducing the oscillation of the loss function is to further improve the efficiency of convergence.

The parameter updating formula of stochastic gradient descent is as follows:

\[
\begin{aligned}
\Delta_t &= \Delta_{t-1} + \alpha \nabla L(W_t^l) \\
W_{t}^l &= W_{t-1}^l + \Delta_t
\end{aligned}
\]  
(9)

Where, \( t \) is the number of iterations, \( l \) is the number of convolution layer, \( \Delta_t \) is the offset of gradient update, \( \nabla L(W_t^l) = \frac{\partial L}{\partial (W_t^l)} \) is the gradient, and \( \alpha \) is the learning rate. When using the gradient descent method with impulse to update the weight, considering the influence of the result of the last weight update on the current update direction, the method with impulse to update the weight is as follows:

\[
\begin{aligned}
\Delta_t &= \mu \Delta_{t-1} + \alpha \nabla L(W_t^l + \mu \Delta_{t-1}) \\
W_{t}^l &= W_{t-1}^l + \Delta_t
\end{aligned}
\]  
(10)

Since the offset \( \Delta_t \) of gradient update is an accumulated term in the iteration process, a weight coefficient \( \mu \) of the accumulated term is added, and the weighted offset \( \mu \Delta_{t-1} \) of the last update is taken as the impulse of the current gradient update, so as to correct the direction of gradient update. Generally, the weight coefficient \( \mu \) is close to 1, and the reference value given in TensorFlow is 0.9.

The gradient update diagram of the gradient descent method with impulse is shown below, showing the two-step gradient iteration process. The dotted line with points is the process of updating the gradient by using the random gradient descent method. The dotted line indicates that the gradient is updated according to the new gradient direction first, and then the weighted offset is calculated as impulse to adjust the gradient direction, and finally the new gradient updating direction shown by the solid line is obtained. The schematic diagram of gradient updating is shown in figure 2.

![Figure 2. schematic diagram of gradient update](image_url)

### 4. Experimental results and analysis

Experiments using computer configuration for the Intel Core i5-6500@3.2GH, GPU NVIDIA GTX970, 16 g memory, used in the simulation software of Matlab R2017a, pycharm2017.3.3 experimental training sets of NTIRE2017 DIV2K (http://data.vision.Ee.Ethz.Ch/CVL/DIV2K/DIV2K_train_HR.Zip), high-definition image data sets, the amplification factor is 4. Set \( n_1=32, n_2=32 \), set the initial learning rate in network training to a higher value, that is \( 10^{-4} \) for each layer. The traditional bicubic interpolation, ScSR algorithm and SRCNN algorithm were selected.
for comparative test. Use SET5 and SET14 as test image sets. The experimental results are shown in table 3 and table 4. From the experimental results in the SET5 test set, it can be seen that the peak signal to noise ratio (PSNR) of the Woman image did not change before and after the improvement, the ratio of the PSNR of other images and the structure similarity was significantly improved compared with the original algorithm under the improved algorithm. Two representative and detailed images were selected for comparison test. The test image is shown in figure 3, and the resolution of each image is shown in table 2.

![Figure 3. test image (a) butterfly; (b) comic](image)

Table 2. test image resolution

| picture | resolution |
|---------|------------|
| butterfly | 256x256    |
| comic     | 250x361    |

Table 3. PSNR on SET5 test set, average SSIM

| Picture | Bicubic | ScSR | SRCNN | Improved method |
|---------|---------|------|-------|-----------------|
| Baby    | 33.91dB | 34.29dB | 34.42dB | **34.64dB**     |
| Bird    | 32.58dB | 34.11dB | 34.19dB | **34.27dB**     |
| Butterfly | 24.04dB | 25.46dB | 27.66dB | **28.10dB**     |
| Head    | 30.87dB | 32.58dB | 31.79dB | **32.15dB**     |
| Woman   | 28.56dB | 29.85dB | 30.44dB | 30.44dB         |
| SSIM    | 0.8687  | 0.8817 | 0.8890 | 0.8939          |

Table 4. average PSNR, SSIM on SET14 test set

| Picture | Bicubic | ScSR | SRCNN | Improved method |
|---------|---------|------|-------|-----------------|
| PSNR    | 27.47dB | 28.17dB | 28.85dB | 28.91dB         |
| SSIM    | 0.7722  | 0.7968 | 0.8136 | 0.8176          |

For the convenience of comparison and testing, the initial image amplified by interpolation method is inserted into the network model, and the final high-resolution image is obtained after the processing of the network model. The same magnification factor of 4 is used in all experiments.

4.1. data sets used in the experiment

The experiment uses the training sets NTIRE2017 DIV2K (http://data.vision.ee.ethz.ch/cvl/DIV2K/DIV2K_train_HR.zip), high-definition image data set, containing 800 around 2040x1356 resolution of high-definition images, and the corresponding resolution 800 copies in 510x339 or low-resolution image set includes people, flowers, birds, buildings, details of the many aspects, such as insects, have higher reference for the reconstruction of the experimental effect. The test set uses the generic test set from Image NET's SET5 and SET14 that includes people, flowers, birds, buildings, and insects.
4.2. objective standard evaluation

Usually, the image after compression will differ from the original image. PSNR is widely used in image quality evaluation. However, because the sensitivity of human eyes to image errors is not absolute, PSNR cannot be completely consistent with the visual perception of human eyes. In order to make the objective evaluation standard more consistent with the subjective visual experience of human eyes, the structural similarity (SSIM) concept was put forward by the image and video engineering laboratory of the University of Texas at Austin in 2004. This is an index to measure image similarity, which can evaluate image visual quality more effectively. Since the structure similarity was proposed, it has been widely used to evaluate image quality in image processing together with PSNR.

The contrast image quality is higher, the PSNR value is higher, and the image distortion is less. However, the closer the value of SSIM is to 1, the more similar the structure of the generated image is to the original image, the better the reconstruction result will be.

![Figure 4. comparison of butterfly GT reconstruction in SET5(a)Bicubic (b)SRCNN (c) improved method](image1)

![Figure 5. comic reconstruction comparison in SET14 (a)Bicubic (b)SRCNN (c) improved method](image2)

![Figure 6. three-channel pixel distribution experiment results of butterfly images before and after reconstruction (a) before reconstruction (b) after reconstruction](image3)

As it is seen from Table III, the experimental results of using SET5 test set are compared among themselves, where the improved algorithm is compared with the bicubic interpolation and SRCNN algorithm, and we can see that the super-resolution results of using the improved algorithm are better. On the SET14 test set, the same experiment is carried out and the same results are obtained, as shown in table 4.

By enlarging the butterfly's wing pattern in figure 4 and the girl's headdress in figure 5, it is seen that the image reconstructed by using this method obtains better results, richer details and clearer overall. Figure 6 shows the experimental results of pixel distribution before and after reconstruction of butterfly image. The horizontal coordinate represents 256 grayscales and the vertical coordinate represents the number of pixels.

5. Summary and prospect

After analyzing the shortcomings of using linear correction unit in the original neural network, this paper proposes an improved linear correction unit with parameters for feature extraction. Combining with PSNR and SSIM, two important image quality evaluation methods, through a series of comparative experiments and verification experiments on DIV2K data set and ImageNET data set, it is shown that the improved algorithm is superior to the SRCNN algorithm before the improvement in both subjective vision and objective evaluation. At present, there are still many directions to be explored.
considered in the research field of image super-resolution reconstruction using deep learning method. How to optimize the network structure to make the algorithm more efficient and the network training time shorter needs to be further studied.

References
[1] Tsai R Y, Huang T S. Multiple frame image restoration and registration. Advances in Computer Vision and Image Processing[C].Greenwich: JAI, 1984.317-339.
[2] Xiao J, Pang G, Zhang Y. Adaptive shock filter for image superresolution and enhancement[J]. Journal of Visual Communication and Image Representation, 2016, 40:168-177.
[3] Hang K, Tao D, Gao X. Learning multiple linear mappings for efficient single image superresolution[C]. IEEE Transactions on Image Processing, 2015, 24(3):846-861.
[4] Polatkan G, Zhou M Y, Carin L, et al. A Bayesian nonparametric approach to image superresolution [C]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 37(2):346-358.
[5] Yu D, Deng L, Seide F. The deep tensor neural network with applications to large vocabulary speech recognition[C]. IEEE Transactions on Audio, Speech, and Language Processing, 2013, 21(2):388-396.
[6] Xu B, Wang N Y, Chen T Q. Empirical evaluation of rectified activations in convolutional network [J]. In: Proceedings of the 32th International Conference on Machine Learning: Deep Learning Workshop. Lille, France: ICML, 2015.