# Hyperspectral Images Reconstruction based on the 3D Fast Super-Resolution Convolutional Neural Networks

Tang Lijing<sup>1</sup>, Wang Wenju<sup>1</sup>, Gu Sicheng<sup>1</sup>, Wei Taotao<sup>1</sup>, Liao Xiting<sup>1</sup>

<sup>1</sup> University of Shanghai for Science and Technology, China

**Abstract.** Hyperspectral images are widely used with the development of remote sensing technology, but its low spatial resolution greatly limits the application of hyperspectral images. Therefore, we proposed a hyperspectral image super-resolution reconstruction algorithm based on 3d FSRCNN framework model, which preprocessed the hyperspectral image data set successively, extracted features, reduced dimensions, nonlinear mapping, expanded dimensions, and deconvolution to finally realize image super-resolution reconstruction. In this algorithm, three-dimensional convolution is used to convolve both spatial dimension and spectral dimension to capture spatial spectrum characteristics. The whole convolutional network consists of 6 layers, one input layer, four convolutional layers, and one deconvolution layer. For the first five layers of convolution, PReLU was used as the activation function, which effectively prevented the phenomenon of nerve necrosis and improved the model fitting without increasing the computational cost and overfitting risk. Experimental results show that the proposed algorithm can reconstruct high spatial resolution images with less computation and reduce spectral distortion effectively.

Keywords: Hyperspectral Images, Convolutional Neural Network(CNN), Super Resolution

# 1. Introduction

Due to the limitations of imaging sensor technology, signal-to-noise ratio and time, there is a certain trade-off between spatial resolution and spectral resolution. Therefore, hyperspectral images often have a low spatial resolution, which greatly limits their effectiveness in practical applications. Thus, how to improve the spatial resolution of hyperspectral images is of great significance.

In 2017, Galliani and others to try in the spectral dimension of hyperspectral image super-resolution, prove the feasibility and advantage of the convolution spectrum dimension of [1], to solve the problems of the spectral distortion also provides a new way of thinking. In the application of hyperspectral based on CNN, Makantasis et al. used random principal component analysis (RPCA) [2] to integrate spatial-spectral information into CNN, but the information would be lost, so the spatial-spectral characteristics provided to CNN by RPCA could not be directly extended to the super-resolution. To explore the hyperspectral image super-resolution spatial context and space spectrum identification, Mei et al., this paper proposes a threedimensional full convolution neural network (3D - FCNN) framework [3], using three-dimensional convolution operation to study between the adjacent pixels of space environment and the spectral correlation of adjacent band image, thus reduce the spectral distortion, but this method is still lacking in computational efficiency. In 2016, Dong Chao et al. proposed the FSRCNN (fast-super-resolution Convolutional Neural Networks) model [4] based on SRCNN. FSRCNN obtains the same sensory field with a smaller number of parameters, improves the computing efficiency and overcomes some shortcomings of SRCNN. However, in terms of reconstruction effect, other models inferior to the Convolutional layer number are still used for twodimensional images. The two-dimensional convolutional layer mainly considers spatial information. When these networks are directly applied to hyperspectral images in the form of band-to-band, they will ignore the strong spectral correlation of hyperspectral images, thus easily leading to spectral distortion [1]. Therefore,

<sup>+</sup> Corresponding author. Tel.: + (86)13636675266.

E-mail address: wangwenju666@163.com.

when applying a convolutional neural network to hyperspectral images, its image characteristics should be taken into account, that is, the spatial background of adjacent pixels and the spectral correlation between adjacent bands should be taken into account, to maintain the spectral fidelity of the output image. So, based on the two-dimensional FSRCNN, we proposed a 3D FSRCNN framework and applied it to the super-resolution reconstruction of hyperspectral images.

# 2. 3D FSRCNN Hyperspectral Image Reconstruction with Super-resolution

This paper proposes a 3D FSRCNN framework, which applied to the super-resolution processing of hyperspectral images. The flow chart is shown in figure 1. By studying the spatial background and spectral correlation of hyperspectral images, the two-dimensional model was mapped to the three-dimensional level. Considering the hardware facilities, we did not adopt a large data set. Our work is mainly divided 1) data preprocessing: standardize the data set, reduce the calculated value, improve the operation efficiency, and then divide the data set into training set and test set in proportion; 2) training: import the training set and mark off a small part as the verification set to verify the performance of the model during the training; 3) test: test the effect on the trained model with test set. For a hyperspectral data set, we can divide the training set, verification set and test set to ensure the same distribution of three sets and avoid meaningless model detection.



Fig. 1: Algorithm flow chart

## 2.1. 3D FSRCNN Algorithm Framework

Compared with the first proposed SRCNN, 2d FSRCNN has its merits and obvious advantages in processing ordinary color images, but its application in 3d images has not been well applied. In this experiment, we took two-dimensional FSRCNN as the prototype [4] and hyperspectral images as the object and proposed a 3D FSRCNN framework to deal with the super-resolution problem of hyperspectral images. The model framework is shown in figure 2.



Fig. 2: 3D-FSRCNN framework

To consider both spatial information and spectral information, the convolution kernel adopted is all three-dimensional. As shown in figure 3, the convolution framework of 3D-FSRCNN can be divided into six parts.

1) Preprocessing: first, the standardized image is sampled with gaussian, and then it is amplified to the original size through double-cubic interpolation, to achieve the purpose of input as "low-resolution" image;

2) Feature extraction: fragmenting feature extraction is performed on the pre-processed data set. The features extracted by the filter size  $s_1 \times s_1 \times s_1$  are the input of the next layer;

3) Dimensionality reduction: to reduce the computation and complexity of the model,  $s2 \times s2 \times s2$  convolution kernel is adopted to reduce the dimensionality of the data;

4) Nonlinear mapping: two  $s3 \times s3 \times s3$  filters are used in series. Compared with a large filter, the series of two smaller filters can reduce the computational cost of parameters and ensure that the receptive field remains unchanged;

5) Dimension expansion: the reconstruction effect of lower dimensions is not good, so this step expands the data to the dimension before dimension reduction to ensure the reconstruction effect. The filter size is  $s2 \times s2 \times s2$ ;

6) Deconvolution: Zero Padding refers to the number of laps around the original data that add 0. To prevent the boundary effect from happening in the training process, the proposed 3D-FSRCNN USES valid convolution, that is, without filling in 0, the size of the image will decrease as the convolutional layer deepens. Therefore, a filter of size  $s4\times s4\times s4$  is adopted to make the output image consistent with the size of the original image, and the output of this layer is the output of the whole convolutional network.

Compared with the two-dimensional FSRCNN, the proposed 3D-FSRCNN model requires preprocessing of the data set, the bicubic interpolation only considers the spatial information, and the three-dimensional convolution can reduce the spectral distortion to improve the accuracy of experimental results. In terms of parameter setting of a convolutional layer, some improvements are also made which are different from 2D FSRCNN. Stride refers to the number of intermediate data that the window skips from the current position to the next position. In 2d FSRCNN, the value of the Stride between different layers is not the same. In the 3D-FSRCNN proposed in this paper, the Stride is all 1. Because of the change in step size, when s3=3, four filters of  $3\times3\times3$  size were used in the two-dimensional FSRCNN, while the improved 3D-FSRCNN only used two filters of  $3\times3\times3$  size. The specific parameters of the rest convolution kernel were set as mentioned in section 2.2.3. The activation function finally adopted in our framework algorithm is PReLU function; The loss function used in this paper is MSE, and the mathematical expression is shown[5] in equation (1).

$$MSE = \frac{1}{m} \sum_{n=0}^{m} \left[ \frac{1}{hwc} \sum_{i=0}^{h} \sum_{j=0}^{w} \sum_{k=0}^{c} (Y_{ijk} - X_{ijk}) \right]$$
(1)

Where m is the number of training samples, h and w are the length and width of network F(X) output, c is the frequency band of training samples, Y is the center pixel of high-resolution images, and X is the network output.

Two-dimensional FSRCNN is used to deal with the problem of the super-resolution of ordinary color images. It is not advisable to directly apply to hyperspectral images. Therefore, the proposed 3D-FSRCNN model is improved according to the characteristics of objects, as shown in figure 3.

Compared with two-dimensional FSRCNN, 3D-FSRCNN needs to preprocess the data set, and the latter is more concise on the number of network layers. Zero Padding refers to the original data in 0 turns around, there are two kinds of forms -- same and valid, to prevent the boundary effect occurred in the process of training, the proposed 3 d - FSRCNN USES is valid convolution, i.e. not fill 0, valid form leads to the final output image is smaller than the input image, but because we the last layer of deconvolution, adopted our output and input in size is the same. Stride refers to the number of intermediate data skipped from the current position to the next position of the window. In two-dimensional FSRCNN, the Stride value between different layers is not the same. In the 3D-FSRCNN proposed in this paper, the Stride value is all 1, and the step length and the setting of the convolution kernel parameter are mutually affected. Because of the different step sizes, when s3=3, four filters of  $3\times3$  size were used in 2D FSRCNN, while the improved 3D-FSRCNN only used two filters of  $3 \times 3 \times 3$ , and the specific parameters of the rest convolution kernel were set in section 2.1.2.



Fig 3. FSRCNN and 3D-FSRCNN model comparison.

#### 2.2. Data Pre-processing

Data preprocessing we carried out data standardization and data segmentation, among which the main work is data standardization.

#### (1) Data normalization

Normalization of data is the scaling of data into a small, specific range. Min-max normalization, also known as deviation normalization, is used in this experiment as a linear transformation of the original data, resulting in the result falling to the interval [0,1]. The conversion function is shown[5] in formula (2) :

The sequence  $x_1, x_2, ..., X_n$  to the transformation

$$y_i = \frac{x_i - \min\left\{x_j\right\}}{\max_{1 \le j \le n} \left\{x_j\right\} - \min_{1 \le j \le n} \left\{x_j\right\}}$$
(2)

Where Max is the maximum value of sample data, min is the minimum value of sample data, and the new sequences y1, y2, ...,  $yn \in [0,1]$  and dimensionless. One drawback to this approach is that when new data is added, it can lead to changes in Max and min that need to be redefined.

#### (2) Data segmentation

The standardized data set is divided into a training set and a test set in a certain proportion, and the proportion should be flexibly determined according to the size of the data set.

#### **2.3.** Model Parameter Setting

The whole convolutional network in the proposed model consists of 6 layers, one input layer, four convolutional layers, and one deconvolution layer. The last five layers are connected to the input layer in turn. For the first five layers of convolution, PReLU was used as the activation function, which effectively prevented the phenomenon of nerve necrosis and improved the model fitting without increasing the computational cost and overfitting risk. For the deconvolution layer, the output of the layer is the output of the whole network, which is not affected by the activation function. In general, the number of parameters to be optimized in CNN is proportional to the number of neurons. Regarding image super-resolution, the scale of input has a great influence on the scale of the network. The larger the scale of input, the greater the parameter calculation, and the greater the requirement for hardware performance. Therefore, in our experiment, the whole training set image is divided into sub-images of 33\*33\*102, which are fed to the convolutional layer, and the parameters are set as s1=5, s2=1, s3=3 and s4=9.

The convolutional layer changes are shown in table 1. The input limit is  $33 \times 33 \times c \times 1$  pixel sub-image cube, where  $33 \times 33$  is the spatial size, c is the special size determined according to the characteristics of the sensor, and the color channel of the hyperspectral image is set as 1. All the filters in the continuous

convolutional layer were designed to learn the spectral information from the adjacent bands of c. It is important to note that large sub-images can also be used as input to design the model, which has a structure similar to that of this article. However, as mentioned above, input size has an impact on the network, so for easier and faster training, the size of our network is limited to a relatively small range.

Layer	Input Size	Kernel Size	Output Size
Conv1	(33, 33, c, 1)	(5, 5, 5, 56)	(29, 29, (c-4), 56)
PRelu1	(29, 29, (c-4), 56)	-	(29, 29, (c-4), 56)
Conv2	(29, 29, (c-4), 56)	(1, 1, 1, 12)	(29, 29, (c-4), 12)
PRelu2	(29, 29, (c-4), 12)	-	(29, 29, (c-4), 12)
Conv3	(29, 29, (c-4), 12)	(3, 3, 3, 12)	(27, 27, (c-6), 12)
PRelu3	(27, 27, (c-6), 12)	-	(27, 27, (c-6), 12)
Conv4	(27, 27, (c-6), 12)	(3, 3, 3, 12)	(25, 25, (c-8), 12)
PRelu4	(25, 25, (c-8), 12)	-	(25, 25, (c-8), 12)
Conv5	(25, 25, (c-8), 12)	(1, 1, 1, 56)	(25, 25, (c-8), 56)
PRelu5	(25, 25, (c-8), 56)	-	(25, 25, (c-8), 56)
DeConv	(25, 25, (c-8), 56)	(9,9,9,1)	(33, 33, c, 1)

Table 1: Convolutional layer details of 3D-FSRCNN

#### **3.** Experiments and results

We chose Pavia Centre, a hyperspectral image data set, as the experimental data, with 1096\*1096 pixels in the center and 102 spectral bands. To make the experiment more convincing, the Pavia Center scene data used in the experiment discarded the samples that did not contain any information; only 1096\*715 valid pixels were used. This data set is taken as the ground truth value of high spatial resolution hyperspectral images, and the performance of the proposed hyperspectral image super-resolution model based on 3D-FSRCNN is trained and evaluated. The size of the image used for training is set to 33\*33\*102\*1, which is obtained by overlapping the original data. The whole experiment was designed, trained and tested based on Matlab and Keras, with a basic learning rate of 0.0005.

To evaluate the quality of spatial reconstruction, mean peak signal to noise ratio (MPSNR) and mean structure similarity (MSSIM) were used. MPSNR is defined as shown[5] in formula (3).

$$MPSNR = \frac{1}{c} \sum_{i=0}^{c} 10 \times \log_{10} \left[ \frac{MAX_i^2}{MSE_i} \right]$$
(3)

The MSSIM between reconstructed image F(X) and ground truth value Y is defined as shown[5] in equation (4).

$$MSSIM = \frac{1}{c} \sum_{i=0}^{c} \frac{(2\mu_{F(X)_{i}}\mu_{Y_{i}})(2\sigma_{F(X)_{i}Y_{i}}+c_{2})}{(\mu_{F(X)_{i}}^{2}+\mu_{Y_{i}}^{2}+c_{1})(\sigma_{F(X)_{i}}^{2}+\sigma_{Y_{i}}^{2}+c_{2})}$$
(4)

Where F(X)i and Yi represent the i-band images of F(X) and Y,  $\mu_{F(x)i}^{2}$  and  $\mu_{Yi}^{2}$  are the mean of  $F(x)_{i}$  and  $Y_{i}$ , respectively,  $\sigma_{F(x)i}^{2}$  are the variances of  $F(x)_{i}$  and  $Y_{i}$ ,  $\sigma_{F(x)iYi}$  is the covariance of  $F(x)_{i}$  and  $Y_{i}$ ,  $c_{1}$  and  $c_{2}$  are set to 0.0001 and 0.0009, respectively. Generally, higher MPSNR and MSSIM values mean better visual quality.

#### 3.1 Experimental results and analysis

The performance of the proposed 3D-FSRCNN was evaluated on the Pavia Centre, a hyperspectral image data set. For each data set, we marked out a  $132 \times 132$  area as the test set and took 0.05% of the rest as the verification set. Finally, the rest was the real training set. The verification set is included in the training set but does not participate in the process of gradient descent. By observing the results of the verification set, we can appropriately change the super parameters such as learning rate and iteration number to make the results optimal on the verification set, which can effectively improve the efficiency of the experiment. To simulate the hyperspectral image with low spatial resolution, we first sampled the image with a factor of 2.

By comparing our model with bicubic interpolation and 3D-FCNN models, the model performance was qualitatively compared from two data indicators of PSNR and SSIM. The comparison results are shown in

Table 2, and the corresponding visual effects are shown in figure 4. The red box is the contrast area, and the reconstruction is shown in the upper left corner.



Fig. 4: The visual effects of hyperspectral images with different super-resolution methods

Data set	evaluation index	Bicubic	3D-FCNN	3D-FSRCNN
Pavia	MPSNR	28.84	32.19	33.42
University	MSSIM	0.939	0.972	0.958

Table 2: Comparison of indexes of different methods of hyperspectral image super-resolution

The results show that, in general, the PSNR value of the proposed model 3D-FSRCNN is the highest. Compared with the double cubic interpolation, the image quality reconstructed by 3D-FSRCNN is more optimized in terms of both two indicators and visual effects. The average PSNR value is 4.58 higher and the average SSIM value is 0.019 higher. Compared with 3D-FCNN, the proposed model can output an image with the same size as the input image, while 3D-FCNN will lose the boundary with the convolution, and the input and output are inconsistent.

Table 3 lists the calculation time comparison between the experimental model and the original superresolution algorithm. All the algorithm experiments are conducted on a computer with a GPU of GeForce RTX 2080. As can be seen from the data in the table, in the case of using the same size data set, the improved 3D-FSRCNN proposed by us has more parameters, but the training time is nearly doubled, while the difference in test time is not significant.

	Table 3: The two models were compared on the Pavia Centre dataset			
	3D-FCNN	3D-FSRCNN		
Total params	39,405	10,496,325		
Training	6.3h	3.5h		
Test	0.414s	0.474s		

**T** 1 1 2 **T** 

# 4. Conclusion

In this paper, we have proposed a novel 3D-FSRCNN model which is inspired by the existing FSRCNN model to solve the super-resolution reconstruction of hyperspectral images. This architecture can be used to learn the End to End full spectrum mapping between low-resolution images and high-resolution pictures. Compared to traditional super-resolution algorithms, which mostly based on a two-dimensional convolutional neural network, a three-dimensional convolutional kernel is capable of exploring the spatial

features of adjacent regions and the spectral correlations of neighboring bands in the case of reconstruction of hyperspectral images and it can effectively reduce the distortion.

Deep Learning is currently a hot research area of computer vision and plentiful models have been proposed recently to deal with various problems. In the future, how to optimize the convolutional neural network may be another breakthrough.

## 5. Acknowledgements

The financial support for this work was provided in part by the Natural Science Foundation of Shanghai under Grant 19ZR1435900.

# 6. References

- [1] Galliani, S.; Lanaras, C.; Marmanis, D.; Baltsavias, E.; Schindler, K. Learned Spectral Super-Resolution. arXiv, 2017.
- [2] Liu, S.; Jiao, L.; Yang, S. Hierarchical sparse learning with spectral-spatial information for hyperspectral imagery denoising. Sensors 2016, 16, 1718. [CrossRef] [PubMed].
- [3] Li, J.; Yuan, Q.; Shen, H.; Zhang, L. Noise Removal from Hyperspectral Image with Joint Spectral-Spatial Distributed Sparse Representation. IEEE Trans. Geosci. Remote Sens. 2016, 54, 5425–5439. [CrossRef].
- [4] Fauvel, M.; Tarabalka, Y.; Benediktsson, J.A.; Chanussot, J.; Tilton, J.C. Advances in Spectral-Spatial Classification of Hyperspectral Images. Proc. IEEE 2013, 101, 652–675. [CrossRef].
- [5] Wang, Q.; Meng, Z.; Li, X. Locality Adaptive Discriminant Analysis for Spectral-Spatial Classification of Hyperspectral Images. IEEE Geosci. Remote Sens. Lett. 2017, 11, 2077–2081. [CrossRef].
- [6] Li, J.; Yuan, Q.; Shen, H.; Meng, X.; Zhang, L. Hyperspectral Image Super-Resolution by Spectral Mixture Analysis and Spatial-Spectral Group Sparsity. IEEE Geosci. Remote Sens. Lett. 2016, 13, 1250–1254. [CrossRef].
- [7] Makantasis, K.; Karantzalos, K.; Doulamis, A.; Doulamis, N. Deep supervised learning for hyperspectral data classification through convolutional neural networks. In Proceedings of the 2015 IEEE International Geoscience and Remote Sensing Symposium (IGARSS), Milan, Italy, 26–31 July 2015; pp. 4959–4962.
- [8] Shaohui Mei , Xin Yuan, Jingyu Ji , Yifan Zhang , Shuai Wan and Qian Du. Hyperspectral image spatial superresolution via 3D full convolutional neural network. MDPI AG.
- [9] Chao Dong, Chen Change Loy, Xiaoou Tang. Accelerating the Super-Resolution Convolutional Neural Network[J]. Computer Vision - 14th European Conference, ECCV 2016, Proceedings