# Convolutional Neural Network Structure Optimization based on Network Pruning

Xudong Zhong

School of Computer, Central China Normal University, Wuhan, China

15279032388@163.com

**Abstract.** Recently, the number of layers of neural network model is deeper and deeper, the number of parameters is more and more, and the calculation scale is also larger and larger. This improves the use conditions of some excellent models, which is not conducive to the wide application of deep learning methods in more fields. In view of this trend of increasing the size of neural network models, in this paper, we optimize the structure of a convolutional neural network model for image super-resolution, which reduces the size of the model. The model structure optimization method we use is network pruning, which simplifies the number of layers and parameters of the model, improves the effect of the model and reduces the computational consumption of the model. The key insight of network pruning is to remove the relatively redundant and unimportant parts of the network to make the original network sparser and more streamlined. And the simplified model can keep the original performance. The original model used a cascade structure for multiple sampling, resulting in the increase of the scale of the neural network. By removing the redundant sampling structure and retaining only one sampling process, the number of layers of the model is reduced to 1/3 of the original. Under the same data set (BSD300) training, the PSNR (evaluation index of model effect) of the model is improved from 24.471 db to 24.490 db, and the training time is reduced by 13.8% of the original.

**Keywords:** CNN; Network Pruning; Parameter Scale; Model Performance; Training Time.

## 1. Introduction

With the continuous development of convolutional neural network, its structure becomes more and more complex. Generally speaking, large models work better than small models, so the development direction of convolutional neural networks has shifted to deeper structures and more parameters. However, the large-scale neural network structure increases the consumption of memory and computation, which limits its promotion and use.

This paper describes the process of structural optimization of a convolutional neural network for super-resolution accurate reconstruction of an image. Based on the use of convolutional layers, this neural network uses a three-level cascade method to sample the input image multiple times, so as to obtain more feature information and improve the model effect. Through the analysis of this model, we learned that though the three-level cascaded network structure can obtain more information, it increases the number of network parameters and increases the computational consumption. So, we wondered if we could reduce the number of parameters by simplifying the model structure while maintaining model performance? The challenge faced by this idea is that the three-level cascade structure has been used many times to obtain more image feature information, thereby improving the performance of the model. So how can we keep the model performance unchanged or improve with less information while simplifying the structure? Our solution for this is to simplify the three-level cascade structure to one level. The original structure is proven to be redundant in subsequent experiments. The characteristic information they extracted is also invalid. Through network pruning, the scale of the model is reduced to 1/3 of the original, and the computational consumption of training the model is also reduced. But the performance of the model will be improved. Among them, PSNR (model performance evaluation index), which was originally 24.471 db, is now 24.490 db. At the same time, the training time was reduced by 13.8%. In general, we simplify the three-level cascade structure of the convolutional neural network model to one level through network pruning, which reduces the number of parameters and computational consumption and improves the performance.

### 1.1 Related Work

Nowadays, with the rapid development of deep learning field, people often construct complex deep neural networks and train them on datasets to obtain models that perform better than traditional machine learning algorithms. However, as the parameter scale of the neural network model gradually increases, the simplification of the model is imminent. Optimizing the structure of convolutional neural network [1, 2] is helpful for the promotion and application of deep learning in a wider range, and it has become a research hotspot in the field of deep learning. One of the recent popular structural optimization methods is network pruning [1, 4, 5]. A more comprehensive approach to structural optimization can be found in [1].

Network pruning has the characteristics of simple implementation and excellent effect, and has become an important technology for optimizing the structure of convolutional neural networks. In different stages of convolutional neural training, the methods of network pruning are also different. The network pruning method during network training is sparse constraint, and after network training is structured pruning [6, 7]. For the former, the specific implementation process is to add a sparsity constraint to the optimization function of the neural network, so that the network structure tends to be sparser during the training process. The advantage of this end-to-end processing method is that there is no need to pre-train the model in advance, which simplifies the optimization process of the neural network. For the latter, pruning an already trained model. Neural networks can also be made leaner by removing redundant, unimportant structures from the model. All in all, whether using sparse constraints during model training or removing model structure after training, the ultimate goal is to make the network weight matrix sparse. This is also an important way to improve network training speed and prevent network overfitting [8, 9, 10]. However, the disadvantage of this method of network pruning is that it needs to manually analyze the network structure and judge the importance of each structure, so it requires a certain accumulation of experience and a lot of energy.

### 1.2 Motivations and Contributions

The results of convolutional neural networks in the fields of image [11] and visual recognition, natural language processing [12] are huge. However, when solving larger-scale problems, convolutional neural networks have the characteristics of numerous parameters, huge structures, and high computational consumption. Therefore, the storage cost of the network is significantly increased, which will limit the promotion and application of network models. In addition to this, there may also be a problem of over-parameterization in convolutional neural networks, which means that there are redundant parameters. Therefore, the simplification of the network model is necessary. To this end, various model compression methods are proposed to solve this problem. Among them, network pruning are currently popular methods. It reduces storage capacity and computing overhead by removing redundant structures and parameters in the network.

This paper successfully optimizes a convolutional neural network through the network pruning, and the improved model mainly has three advantages:

- The parameters of the model have declined significantly. The number of layers of the initial convolutional neural network structure has 15 layers. However, there is a lot of redundant structures in the model. The model uses a three-level cascade structure to make one normal processing and two down sample processing. In the experiment, we prove that two down sample processing is redundant and invalid. Therefore, we remove two down sample processing, so that the scale of the overall scale is small to the original 1/3, the number of layers is reduced to 5 layers. This will make the memory occupied by the model, and the performance requirements of the running equipment are also reduced, and the model's application scenario will be increased.
- The calculation overhead during model training is greatly reduced. Since we simplify the network structure through the network pruning, only the process of image processing is retained. This makes the number of model parameters decrease, and the calculation overhead of model training is reduced. The training time is reduced by 13.8% more than the original model.

- The model effect is improved. The initial model uses a three-level cascade structure, although this will increase the model size, but excess structures will also extract more feature information. Therefore, we must consider whether the model will reduce performance after the network is simplified. After experiment, we have found that the model performance after the network pruning has a small increase, which proves that the extra-feature information extracted is redundant. It also proves that these structures in the model are redundant.

## 2. Method

The convolutional neural network mentioned [3] in this paper uses a three-level cascade structure. The input to the first image processing pass is the full image, and the input to the second and third processing passes is the downsampled image. The initial model repeats the convolution operation on the image three times through the cascade structure, and extracts the information multiple times. However, the three repeated convolution process increases the number of model parameters, thereby increasing the storage cost and computational cost.

To solve this problem, some optimization methods for convolutional neural networks are mentioned in [1,2]. The method chosen in this paper to optimize the network is network pruning [5, 6, 7].

### 2.1 Network Pruning

In general, network pruning is to delete some unimportant parts under the condition that the network performance is not degraded. According to the different objects to be deleted, pruning can be divided into weight pruning and neuron pruning. The former deletes the unimportant weights in the network, and the latter deletes the unimportant neurons in the network.

In this paper, the neuron pruning method is used to prune the unimportant neurons in the model. After analyzing the structure of the model, we crop the structure of the model used to process the downsampled image, this part of the structure takes up 2/3 of the model size in total. After network pruning, the performance of the model remains unchanged and the number of parameters is greatly reduced.

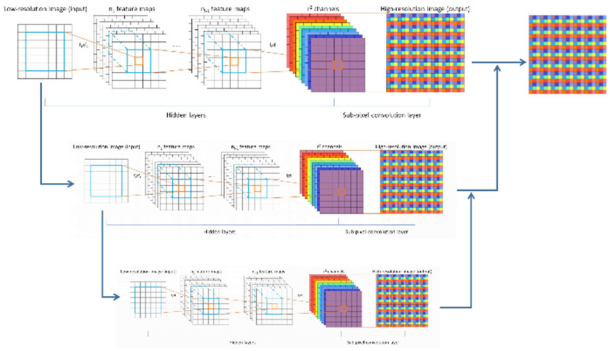

**Figure 1.** Initial convolutional neural network structure.

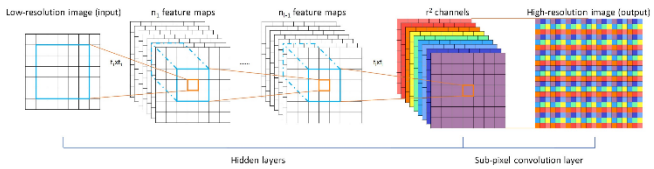


**Figure 2.** Convolutional neural network structure after network pruning.

# 3. Experiments

In this experiment, we investigate how network pruning can be done so that the model can be reduced in size while maintaining performance. The evaluation criterion for model size is the number of parameters. The evaluation criteria for model performance are PSNR and training time. In the experiments, we performed different network pruning operations on the models and compared them according to the PSNR and training time of the changed models. It is found that removing the double down sampling processing structure in the model is a better method for network pruning.

## 3.1 Experiment Preparation

We use the publicly available benchmark dataset BSD300 during our experiments. The dataset is provided by the computer vision group of the University of Berkeley, including 200 training images and 100 test images.

The model used in the experiments is a convolutional neural network model whose function is image super resolution, consisting of 15 layers in total.

Before the experiment starts, for the initial model, we use the BSD300 dataset for training, and record the PSNR and training time of the model under 30epochs, 50epochs, and 100epochs training, respectively. These results will be used for comparison of model performance after network pruning.

## 3.2 Implementation Details

In the initial stage of the experiment, through the analysis of the initial model structure, we found that the model using a three-level cascade structure is the biggest reason for the increase of the model size. Therefore, in the experiments, we focus on how to optimize the cascade structure of the model.

First of all, since the model is a three-level cascade structure, we remove the last level in the cascade structure. After that, the improved model was trained on the same data set, and it was found that the PSNR of the new model under 30epochs, 50epochs, and 100epochs training improved, and the training time decreased. This proves that this network pruning method is effective.

Furthermore, we continue to think along this line of thought: since the model has been successfully optimized by removing the primary cascade structure, can the model be optimized by removing all the cascade structures? Therefore, we continued to experiment with this idea. We crop all the structures used for cascading in the model, leaving only one image processing pass. And train the second improved model. The result is that under 30epochs, 50epochs, and 100epochs training, the model has improved from the first improvement.

## 3.3 Results

In this experiment, we perform network pruning on the initial model. The new model is obtained by removing the three-level cascade structure of the convolutional neural network. These models were trained on the same dataset, and their performance and training time were compared. It can be concluded that removing the cascade structure of the model can improve the performance of the model, reduce the computational consumption of the model, and reduce the parameter scale.

**Table 1.** Average PSNR (dB) for different models. The best results for each category are shown in bold

| Model | *30epochs* | *50epochs* | *100epochs* |
|---|---|---|---|
| Initial model | 24.471 | 24.613 | 24.727 |
| First improvement | **24.544** | **24.636** | **24.768** |
| Second improvement | 24.491 | 24.622 | 24.732 |
| Average | 24.502 | 24.624 | 24.746 |

**Table 2.** Average training time *(min)* for different models. The best results for each category are shown in bold

| Model | *30epochs* | *50epochs* | *100epochs* |
|---|---|---|---|
| Initial model | 11:31 | 19:24 | 39:06 |
| First improvement | 10:14 | 17:11 | 35:00 |
| Second improvement | **7:53** | **13:35** | **26:16** |
| Average | 9:53 | 16:43 | 33:27 |

## 4. Conclusion

In this paper, we demonstrate that the initial convolutional neural network model using a three-level cascade structure is suboptimal. This structure does not improve the effect of the model, but increases the scale of the model and requires more storage space and computing resources. To solve this problem, we propose a method of network pruning. At the same time, we analyze the model structure and think that the cascade structure can be used as a breakthrough point. To this end, we try to remove part of the cascade structure of the model in the experiment, and compare the effect of the improved model with the original model. After the comparison, it was found that the performance of the first improved model did not decrease, and the computational consumption and memory consumption were reduced. This clearly justifies pruning against the model cascade structure. Therefore, we continue to experiment with this idea. In the next experiment, we remove the entire cascade structure. It was found that the performance of the model with the second improvement did not drop, while the computational consumption and storage consumption decreased again compared with the first improvement. Through network pruning, we simplify the structure of the initial model and no longer use the three-level cascade structure.

Compared with the initial model, the performance of the final improved model remains unchanged, the model size is greatly reduced (the model size is 1/3 of the original), and the computational consumption is significantly reduced (the training time is reduced by 13.8%).

## 5. Future Work

Network pruning can remove redundant structures in the network, thereby optimizing and adjusting the entire model architecture. This method compresses the network size at the cost of less precision loss, making the model run faster and smaller in scale, and the accuracy is similar to the original. It is the most widely used network structure optimization design method. Most of the current methods are to remove redundant connections or neurons in the network. This low-level pruning has the risk of non-structural. In addition to network pruning, other convolutional neural network structure optimization methods include tensor decomposition, knowledge transfer, and fine module design. In the future, we will consider using these methods to improve the model.

## References

[1] Lin Jing-Dong, Wu Xin-Yi, Chai Yi, Yin Hong-Peng. Structure optimization of convolutional neural networks: a survey. Acta Automatica Sinica, 2020, 46(1): 24-37.

[2] Bai C, Huang L, Chen JN, Pan X, Chen SY. Optimization of deep convolutional neural network for large scale image classification. Ruan Jian Xue Bao/Journal fo Software, 2018,29(4):1029-1038.

[3] W. Shi et al., "Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 1874-1883, doi: 10.1109/CVPR.2016.207.

[4] Lei J, Gao X, Song J, Wang XL, Song ML. Survey of deep neural network model compression. Ruan Jian Xue Bao/Journal of Software, 2018,29(2):251-266.

[5] HUANG Cong, CHANG Tao, TAN Hu, et al. Neural network pruning based on weight similarity. Journal of Frontiers of Computer Science and Technology, 2018, 12(8): 1278-1285.

[6] Jiang C H. The Study of Pruning Methods of Deep Neural NetWork. University of Science and Technology of China, 2019.

[7] Shen X X. Research of Compress Deep Neural Networks by Network Pruning[D]. Jilin University, 2021

[8] Liu Danfeng, Liu Jianxia. A Neural Network Model for the Overfitting Problem in Deep Learning [J]. Natural Science Journal of Xiangtan University, 2018, 40(2): 96-99.

[9] TAO Li, YANG Shuo, YANG Wei. Research on model building and over-fitting of deep learning[J]. Computer Era, 2018(2): 14–17, 21.

[10] XIE Luyang, XIA Zhaojun, ZHU Shaohua, ZHANG Daiqing, ZHAO Fengkui. Analysis and Research of Overfitting of Image Recognition Based on Convolutional Neural Networks. Software Engineering, 2019, 22 (10): 27-29.

[11] ZHENG Yuanpan, LI Guangyang, and LI Ye. Survey of application of deep learning in image recognition[J]. Computer Engineering and Applications, 2019, 55(12): 20–36.

[12] Xi X F, Zhou G D. A Survey on Deep Learning for Natural Language Processing. Automatic Synica Acta, 2016, 42(10): 1445-1465.