The Comparison and Analysis of Classic Convolutional Neural Network in the Field of Computer Vision

Hanyue He *

International School, Beijing University of Posts and Telecommunications, Beijing 100876, China

* Corresponding author e-mail: 2017213127@bupt.edu.cn

Abstract. Convolutional neural network algorithm has been proposed as early as 20 years ago, but due to limited numerical computing ability and insufficient learning samples, it stays in the research stage, and is not popular in the application-end. In recent years, with the emergence of large-scale tagged data sets, the support of computer hardware and the improvement of related algorithms, convolutional neural networks have begun to be widely used in the field of pattern recognition and image processing. By analysing the development of CNN (Convolutional Neural Network) and two major models: VGGNet and ResNet, this paper demonstrates the importance of depth, width and residual learning in convolutional neural networks, and summarizes the trends of CNN structure and the basic paradigm of neural networks.

1. Introduction

Computer vision is a simulation of biological vision using computers and related equipment. As one of the most deeply studied neural networks in the field of computer vision, convolutional neural networks have performed extremely well in computer image recognition, video analysis, and natural language processing. Many excellent network models were invented during the development of CNN, such as VGGNet[4] and ResNet[6]. This paper will compare and analyze the two classic convolutional neural networks (VGGNet and ResNet) and summarize the development trends of CNN network structure. Among them, VGGNet increases the depth of CNN by continuously stacking small-sized convolutional layers; By introducing residual learning, ResNet alleviates the degradation problem (caused by the increasing depth of the network) and the vanishing gradients problem. These structural designs make the network layer deeper and the convolution layer wider, which makes the performance of the convolutional neural network reach a higher level.

This paper will make a review of the network before VGG in Sect. 2, compare and analyze the two major convolutional neural networks in Sect. 3, conduct experimental demonstration in Sect. 4. Finally in Sect. 5, we will summarize the changing trends in CNN structure and propose some views and understandings on future applications.

2. Literature Review

With the advent of the large-scale tagged dataset ImageNet[15] and the support of computer hardware, a deeper network structure, AlexNet, emerged. The accuracy rate of AlexNet classification far exceeds the traditional method, making the development of CNN exploding. As shown in Figure 2, from AlexNet (2012) to ResNet (2015), the model has deepened year by year and the test error rate has become lower and lower. It can be seen that in order to achieve greater performance gains, the network needs to go deeper[7] and wider[8,9].
In addition to modifying the structure, people improve the performance of deep neural network through techniques such as data augmentation[2], dropout[10,13], non-saturating activation functions, and local response normalization[2]. Data augmentation is to extend the sample training set by image shifting, horizontal flipping and grayscale adjusting, which is equivalent to adding noise to the data, so as to alleviate over-fitting[2] and improve generalization performance. In the neural network with a large amount of parameters, the dropout layer is used to randomly ignore a part of the neurons, thereby alleviating the over-fitting phenomenon and enhancing the generalization performance of the model. Using non-saturating activation functions (such as ReLU[2] and PReLU) instead of the traditional saturating activation functions accelerates the speed of network training, reduces the computational complexity, makes the network more robust, and alleviates the vanishing gradients—which lays the foundation for increasing network depth.

3. Comparison
The bulk of this section concerns the comparison of the algorithm and structure of two classic convolutional neural networks. The core of the algorithm comparison are VGGNet and ResNet.

3.1. VGGNet
In general, the characteristics in which the VGG model differs from the traditional CNN model[1,10] are as following: (a). The convolution kernel becomes smaller [14](a stack of a large number of 3*3 conv). Replacing a single large convolution kernel with a stack of multiple small convolution kernels, can improve the accuracy and reduce the amount of calculation and parameters of the architecture; (b). The pooling kernel becomes smaller (2*2) and is even (to better match the size of the feature map) to further reduce the loss of spatial information; (c). Deeper and wider. It can be seen from (a) that a stack of small-sized convolution kernels is equivalent to large-scale convolution kernels (2 layers of conv3 are equivalent to 1 layer of conv5, and 3 layers of conv3 are equivalent to 1 layer of conv7 etc.). By continuously stacking conv3 at various stages, the depth of the network is deepened. And by introducing more layers of nonlinear activation functions, the ability to represent features is greatly improved.

The entire network is composed of convolution layers, pooling layers, fully connected layers, and a LRN layer. As shown in Table 1, most convolution layers are
Table 1. ConvNet configurations of VGG[4].

| ConvNet Configuration | A layer | B layer | C layer | D layer | E layer |
|-----------------------|---------|---------|---------|---------|---------|
| A weight layers       | 11       | 11       | 13       | 16       | 19       |
| B weight layers       | 11       | 15       | 16       | 16       | 19       |
| C weight layers       | 15       | 16       | 19       |          |          |
| Input (224 × 224 RGB image) | conv3-64 | conv3-64 | conv3-64 | conv3-64 | conv3-64 |
| maxpool               | conv3-128 | conv3-128 | conv3-128 | conv3-128 | conv3-128 |
| maxpool               | conv3-256 | conv3-256 | conv3-256 | conv3-256 | conv3-256 |
| maxpool               | conv3-512 | conv3-512 | conv3-512 | conv3-512 | conv3-512 |
| maxpool               | conv3-512 | conv3-512 | conv3-512 | conv3-512 | conv3-512 |
| maxpool               | FC-4096  | FC-4096  | FC-1000  | softmax  |          |

3*3, and in one of the network configurations, 1*1 convolution layers are used as a linear transformation of the input channels[7]. After each stage, downsampling is performed using a maxpooling layer of size 2*2. For training, the classifier consists of three fully connected layers and a softmax layer[3,12,11]. The first two fully connected layers have a width of 4096, and the third has a width of 1000 corresponding to 1000 classifications of ImageNet. Finally, the softmax regression maps the confidence of each category to complete the classification. For testing, the three original fully connected layers of VGGNet are replaced by full convolutional layers, which are one 7*7 convolution layer and two 2*1 convolution layers.

Because the convolutional layers of VGGNet are of the same receptive field[14,3], the entire network structure is very clean. And it’s also proved that the classification task can increase the depth of the CNN by using successively stacking small-sized convolution layers, thereby improving the accuracy. However, at the same time, VGG has a large amount of parameters, which generates excessive memory consumption and is not conducive to the generalization ability of the model.

3.2. ResNet

The core idea of the residual block is to directly add the output of a shallow layer to a deep layer, so that when the characteristics of the shallow network are optimal, the identity mapping of a deeper layer is achieved by a new-built shortcut connection instead of by “plain” network (stacking layers). The task of the original plain network layer is changed from the identity mapping to the residual mapping, which is equivalent to a fine-tuning of the shallow output. The identity map will not introduce additional parameters and computational complexity. However, feature maps will undergo downsampling at different stages of the network. In order to achieve dimension matching, a linear projection by the shortcut connections is required[6] (introduction of new parameters) . As shown in Figure 4, this is the basic residual block. However, as the network deepens, the amount of parameters has explosive increased. By referring to the design criteria in Inception, a 1*1 convolutional layer is used to achieve dimensionality reduction before spatial aggregation, which not only ensures the spatial information of the feature but also greatly reduces the computational cost. The improved block is called the bottleneck block[6] (Figure 5).
Figure 3. A residual building block[6]  Figure 4. A “bottleneck” building block[6]

The residual block realizes the identity mapping of the deep network in the forward propagation, and in the direction propagation, it will significantly reduce the value of the parameters in the block, so that the parameters in the network are more sensitive to the loss value of backpropagation[1,17]. Although it does not solve the problem of vanishing gradients fundamentally, but by reducing the parameters, the effect of backpropagation loss is relatively increased, and a certain regularization effect is also generated. Finally, the number of network layers is further increased, and it is easier to optimize and the performance is better. However, on the extremely deep model of more than 1000 layers, there is still a problem of slow training and over-fitting.

4. Experiment -- CIFAR-10 & CIFAR-100 classification

We trained and tested VGG-16, GoogLeNet, ResNet-50 and ResNet-101 on the CIFAR-10 and CIFAR-100 datasets. The CIFAR-10 dataset includes 60,000 32*32 color images of 10 classes (each class has 6,000 images), and is split into two sets: 50,000 images for training and 10,000 images for testing[5]. The CIFAR-100 dataset is similar to CIFAR-10, and the only difference is it has 100 classes, each containing 600 images.

We select VGG16, GoogLeNet, ResNet-50 and ResNet-101 separately for training. We use Batch Normalization and ReLU after all convolutional layers and use a weight decay of 0.0005 and a momentum of 0.9. All networks are trained on two GeForce RTX 2080Tis with a batch size of 128. The learning rate starts from 0.1. When the number of Epoch reaches 15 and its loss is no longer falling, we use the strategy of adjusting the learning rate dynamically to multiply the original learning rate by a scale factor of 0.2. We follow the simple data augmentation in [3] for training: 4 pixels are padded per side and a 32*32 crop is randomly sampled from the padded image or its horizontal flip. In testing, we only use the original 32*32 image as the input to the network.
Figure 5. Experimental results on CIFAR-100. Top: train loss. The orange curve presents VGG16. The blue curve presents ResNet-101 and the red presents ResNet-50. Middle: train accuracy. The gray curve presents VGG16. The red curve presents ResNet-101 and the blue presents ResNet-50. Bottom: test accuracy. The dark blue curve presents VGG16. The pink curve presents ResNet-101 and the light blue presents ResNet.

Table 2. Classification performance.

| Method    | Parameters | CIFAR-10  | CIFAR-100 |
|-----------|------------|-----------|-----------|
|           |            | Top1-error(%) | Top5-error(%) | Top1-error(%) | Top5-error(%) |
| VGG-16    | 34.01M     | 6.52      | 0.26      | 27.44      | 9.89        |
| ResNet-50 | 23.7M      | 4.92      | 0.11      | 23.87      | 6.15        |
| ResNet-101| 42.7M      | **3.91**  | **0**     | **21.34**  | **5.73**    |

The experimental results suggest that our reproducibility of the CNN models is better than the original, which has a lot to do with the change of weight decay and the adjustment of the learning rate. On CIFAR-10 and CIFAR-100, ResNet-101 has the highest accuracy, which indicates that the increase in network depth and residual learning bring significant accuracy improvements. Apart from this, from Figure 6, we can find that the convergence speed of ResNet-50 is evidently faster than VGG-16, which proves that the residual learning is effective to alleviating the vanishing gradients and accelerating the training process.

5. Conclusion

This paper analyzes the development of CNN and its structural changes, and compares two models: VGGNet, and ResNet. From VGG to ResNet, the trend of the network is going deeper and deeper. VGG increases the network depth by continuously stacking convolution layers, and improves the network performance greatly. ResNet's residual learning ensures the network in good working order after being deepened.

In summary, VGGNet and ResNet greatly alleviate the vanishing gradient, high model complexity and degradation problems, and the recognition ability of CNN has already exceeded the human level. The two paradigms of deepening and residual learning have become the trend of CNN development, and a series of excellent neural networks have been derived accordingly. However, problems such as vanishing gradient have not been solved completely. Further optimization is still needed in future research.

References

[1] Lecun, Y. , Boser, B. , Denker, J. S. , Henderson, D. , Howard, R. E. , & Hubbard, W. , et al. (1989). Backpropagation applied to handwritten zip code recognition. Neural Computation, 1(4), 541-551.
[2] Krizhevsky, A. , Sutskever, I. , & Hinton, G. . (2012). ImageNet Classification with Deep Convolutional Neural Networks. NIPS(Vol.25). Curran Associates Inc.
[3] Lee, C. Y., Xie, S., Gallagher, P., Zhang, Z., & Tu, Z. (2015, February). Deeply-supervised nets. In Artificial intelligence and statistics (pp. 562-570).
[4] Simonyan, K. , & Zisserman, A. . (2014). Very deep convolutional networks for large-scale image
recognition. Computer Science.

[5] Krizhevsky, A., & Hinton, G. (2009). Learning multiple layers of features from tiny images (Vol. 1, No. 4, p. 7). Technical report, University of Toronto.

[6] He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition.

[7] Lin, M., Chen, Q., & Yan, S. (2013). Network in network. Computer Science.

[8] Zeiler, M. D., & Fergus, R. (2013). Visualizing and understanding convolutional networks.

[9] Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R., & LeCun, Y. (2013). Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229.

[10] Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580.

[11] Zeiler, M. D., & Fergus, R. (2013). Stochastic pooling for regularization of deep convolutional neural networks. Eprint Arxiv.

[12] Goodfellow, I. J., Warde-Farley, D., Mirza, M., Courville, A., & Bengio, Y. (2013). Maxout networks. Computer Science, 1319-1327.

[13] He, K., Zhang, X., Ren, S., & Sun, J. (2015). Delving deep into rectifiers: surpassing human-level performance on imagenet classification.

[14] Ciresan, D. C., Meier, U., Masei, J., Gambardella, L. M., & Jürgen Schmidhuber. (2011). Flexible, High Performance Convolutional Neural Networks for Image Classification. IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011. DBLP.

[15] Deng, J., Dong, W., Socher, R., Li, L. J., & Li, F. F. (2009). ImageNet: a Large-Scale Hierarchical Image Database. 2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA. IEEE.

[16] #Deep Learning Review#: LeNet, AlexNet, GoogLeNet, VGG, ResNet.(n.d.). Retrieved August 21, 2019, from https://zhuanlan.zhihu.com/p/24657060

[17] Lecun, Y., Boser, B. E., Denker, J. S., Henderson, D., & Jackel, L. D. (1989). Handwritten Digit Recognition with a Back-Propagation Network. Advances in Neural Information Processing Systems 2, [NIPS Conference, Denver, Colorado, USA, November 27-30, 1989]. Morgan Kaufmann.