Optimal Design of Deep Residual Network Based on Image Classification of Fashion-MNIST Dataset

Yusi Tang, Hanguo Cui and Shuyong Liu*
College of Power Engineering, Naval University of Engineering, Wuhan, China

*Corresponding author email: lsydh@sina.com

Abstract. With the increasing number of layers of deep neural network, the performance of the model tends to be saturated gradually. As a very deep network architecture, the deep residual network shows beneficial characteristics in precision and convergence. This paper deeply studied the essence of deep residual network and intended to design an optimal model for classification in Fashion-MNIST dataset. Due to the significant effects of the hyper-parameters in neural network models, the method of designing model structure and optimizing training process were investigated. Next, the performances of the training models under different conditions were compared, and finally an optimal training model whose accuracy was up to 96.21% was obtained. The experimental result shows that the performance of the model can be improved by partly widening the network and selecting more advanced training process.

1. Introduction
In recent years, the model of deep neural network (DNN) has achieved great success in the field of image classification, which attracts great attention [1]. With deep architectures, DNN can be established to mine the useful information from raw data and learn more advanced semantic features to approximate complex non-linear functions. The most advanced DNN architecture has become more and more deep since AlexNet came out: AlexNet has only 5 convolution layers, while VGGNet and GoogleNet (Inception_v1) have 19 layers and 22 layers respectively, which were presented in the followed years. It's tempting to conclude that the expression ability and feature extraction ability of DNN increase when the network gets deeper [2-3]. However, with the increase of network’s depth, the gradient disappears, which makes it difficult to train the deep layers effectively and leads to a trend of saturated model performance, and the accuracy of model may decline rapidly even worse. In 2015, Kaiming He put forward the deep residual network (ResNet) to ease the training of DNN [4]. The design of identity mapping was introduced into the network, which ingeniously alleviated the problem of gradient explosion or gradient disappearance and network degradation caused by the increase of depth, therefore the number of the network layers can be pushed from dozens to thousands. The presence of ResNet greatly improved the accuracy of the model, made it possible to train very deep network, and has been applied to fields of face recognition [5], object detection [6], semantic segmentation [7], etc., and is a significant breakthrough in the field of image classification.
Based on image classification application of the open source dataset Fashion-MNIST [8], this paper designed several different models and data enhancement methods. Comparing the performance of each model under different optimization methods through ablation experiments, the optimal classification model under this dataset was obtained.
2. Structure of the Network Model

2.1. Residual Network (ResNet)

In the deep layers of neural network, due to its non-linear transformation characteristics, the features learned by the shallow neural network may be lost gradually with the forward propagation. The core idea of ResNet is to introduce identity mapping into deep layers to simplify it into shallow layers, so as to prevent performance degradation caused by continuous stacking of network layers. In fact, the existing neural network is hard to fit the potential identity mapping function \( H(x) = x \). But if the network is designed as \( H(x) = F(x) + x \), which makes the identity mapping directly taken as a part of the network, it can be transformed into learning a residual function \( F(x) = H(x) - x \). As a result, an identity mapping \( H(x) = x \) will be constituted when \( F(x) = 0 \).

The basic unit of the ResNet is the residual unit, as shown in the figure 1.

![Figure 1. Structure of residual unit.](image)

If the input of the first residual element is \( x_i \), then the output of this layer, in other words, the input of the next layer is

\[
x_{i+1} = f(x_i + F(x_i, W_i))
\]

(1)

Where \( F(x_i, W_i) \) is the residual function, \( W_i \) is the weight parameter corresponding to the residual function, and \( f(*) \) is the non-linear activation function ReLU.

Considering the size and number of labels of the dataset selected, it is not appropriate to build a too large model. The structure and parameters of the specific ResNet used in this paper are shown in the table 1.

| Layer name       | Parameters                                                                 | Output size |
|------------------|---------------------------------------------------------------------------|-------------|
| Input            |                                                                           | 32×32       |
| Convolution      | \((7×7, 64), \text{stride 2}, \text{padding 3}\)                          | 16×16       |
| Max pooling      | \(3×3, \text{stride 2}, \text{padding 1}\)                               | 8×8         |
| Convolution      | \([3×3,64]^4, 3×3, \text{stride 1}, \text{padding 1}\)                   | 8×8         |
| Convolution      | \([3×3,128]^4, 3×3, \text{stride 2}, \text{padding 1}\)                  | 4×4         |
| Convolution      | \([3×3,256]^4, 3×3, \text{stride 2}, \text{padding 1}\)                  | 2×2         |
| Convolution      | \([3×3,512]^4, 3×3, \text{stride 2}, \text{padding 1}\)                  | 1×1         |
| Full Connection  | \text{dimension 1000, softmax}                                           | 1           |

*One residual unit with two convolutional layers.*
2.2. **Wide Residual Network (WRN)**

Because of the jump connection of ResNet, the gradient can't flow through the weight layer of every residual unit during reverse training. In this case, most residual units can only provide little information, only a few residual units can learn useful expressions and extract useful features [9]. By reducing the depth and increasing the width on the basis of ResNet, the more effective model WRN is obtained. The "width" here refers to the number of channels of the feature map, and in the convolutional layer refers to increasing the number of convolution kernels.

The structure and parameters of the WRN are shown in the table 2.

| Layer name       | Parameters                                      | Output size |
|------------------|-------------------------------------------------|-------------|
| Input            | (3×3, 16) stride 1, padding 1                   | 32×32       |
| Convolution      | $\left[ \begin{array}{c} 3 \times 3, 16 \times k^a \\ 3 \times 3, 16 \times k \end{array} \right] \times N^b$, stride 1, padding 1 | 32×32       |
| Convolution      | $\left[ \begin{array}{c} 3 \times 3, 32 \times k^a \\ 3 \times 3, 32 \times k \end{array} \right] \times N$, stride 2, padding 1 | 16×16       |
| Convolution      | $\left[ \begin{array}{c} 3 \times 3, 64 \times k^a \\ 3 \times 3, 64 \times k \end{array} \right] \times N$, stride 2, padding 1 | 8×8         |
| Average Pooling  |                                                 | 8×8         |
| Full Connection  | dimension 1000, softmax                         | 1×1         |

| $^a$ $k$ is width factor, when it is 4, the number of convolution kernels is equal to ResNet. |
| $^b$ $N$ is the number of the residual units. |

The optional parameters selected in this paper are as follows: $k = 5$, $N = 5$.

2.3. **Pyramidal Residual Network (PyramidNet)**

The deep residual network can actually be regarded as a set of relatively shallow networks, and consequently deleting a single residual unit from ResNet is equivalent to remove some shallow networks in the integrated network, without obvious having impact on the overall performance. The width of PyramidNet increases with the increase of depth, which is similar to the pyramid structure that gradually widens from top to bottom.

The specific calculation formula for the number of channels of PyramidNet is as follows

$$D_k = \begin{cases} 16 & \text{if } k = 1 \\ D_{k-1} + \frac{\alpha}{N} & \text{if } 2 \leq k \leq N + 1 \end{cases}$$

(2)

where $k$ represents the $k$-th layer, $N$ represents the total number of layers, $D_k$ represents the number of channels of the $k$-th layer, and $\alpha$ represents the number of output channels of the last layer.

The structure and parameters of the WRN are shown in the table 3.

The optional parameters selected in this paper are as follows: $\alpha = 24$, $N_2 = 4$, $N_3 = 6$, $N_4 = 5$.
Table 3. Structure and parameters of PyramidNet.

| Layer name      | Parameters                                                                 | Output size |
|-----------------|----------------------------------------------------------------------------|-------------|
| Input           | (3×3, 16), stride 1, padding 1                                             | 32×32       |
| Convolution     | $\left[3\times 3, 16 + \alpha(k - 1)/N\right]$                           | 32×32       |
|                 | $\times N_2^a$, stride 1, padding 1                                       |             |
| Convolution     | $\left[3\times 3, 16 + \alpha(k - 1)/N\right]$                           | 16×16       |
|                 | $\times N_3$, stride 2, padding 1                                          |             |
| Convolution     | $\left[3\times 3, 16 + \alpha(k - 1)/N\right]$                           | 8×8         |
|                 | $\times N_4$, stride 2, padding 1                                          |             |
| Average Pooling | $\left(8\times 8, 16+\alpha\right)$                                      | 1×1         |
| Full Connection | dimension 1000, softmax 1                                                  |             |

$N_y$ denotes the number of residual units in a group, $N_2 + N_3 + N_4 = N$

3. Optimization Method

3.1. Learning Strategy

WarmUp had been proven an effective procedure for many tasks. Different from the common strategy of decreasing learning rate, WarmUp uses the gradually increasing learning rate to initialize the network, and gradually initializes it as a better search space. When the batch size is large, this method can avoid over-fitting during the initial training. When using the warm up strategy with Cosine Annealing learning rate, the effect is usually good.

Ranger optimizer consists of RAdam and LookAhead mechanisms. RAdam is a new variant of the classic optimizer Adam. The adaptive learning rate can be automatically and dynamically adjusted, and the changes in the training process or the impact of momentum are taken into account. The performance of Ranger is better than the traditional manual warmup. The mechanisms of LookAhead reduces the need for a large number of super parameter adjustments, and achieves faster convergence for different depth learning tasks with minimal computational overhead.

3.2. Data Augmentation

Resize, RandomCrop, RandomHorizontalFlip and RandomErasing are the methods commonly used to enhance training data. During this process, more different training images will be generated, which will reduce the risk of over-fitting and make the model more robust. At the same time, normalization of the data makes the model converge faster.

Auto-Augment designs a data enhanced search space, one of which is composed of many sub strategies. In each batch size, a sub strategy will be randomly chosen for each image. At the same time, the best strategy will be found based on the search algorithm, so that the neural network can produce the highest verification accuracy for the target dataset.

CutMix cuts and pastes patches to the images of two random samples in the training set. When training the images, the labels are also mixed into the patch area in proportion. This is equivalent to the regularization of neural network, so that it has better robustness in the training process.

Test time enhancement (TTA) can improve the accuracy by several percentage points. During the training, multiple versions of the original image are created and input into the model, including different region clipping and changing the zoom level. Then the average output of multiple versions will be calculated as the final output score of the image. By inputting multiple versions of the image in the model and taking the average value, TTA can solve the problems that the display area of original image may lack some important features.
4. Research on Experiment

4.1. Data Description
This paper selected the open source dataset Fashion-MNIST, which contains 10 kinds of clothing pictures. The resolution of each image was 28×28, including 60000 training data and 10000 test data. The structure of dataset is shown in the table 4.

Table 4. Structure of dataset.

| Label | Description   | Label | Description |
|-------|---------------|-------|-------------|
| 0     | T-shirt/top   | 5     | Sandal      |
| 1     | Trouser       | 6     | Shirt       |
| 2     | Pullover      | 7     | Sneaker     |
| 3     | Dress         | 8     | Bag         |
| 4     | Coat          | 9     | Ankle boot  |

4.2. Training Method
for the benefit of speeding up the convergence of the network and improve the performance to a certain extent, the transfer learning method was used to initialize the parameters of models by using the models which had been trained on other datasets. In order to compare the effects of different models and optimization methods on model performance, a set of ablation experiments were designed, as shown in the table 5.

Table 5. Detail of the ablation experiments.

| Group No. | Model      | Learning strategy | Data augmentation                  |
|-----------|------------|-------------------|------------------------------------|
| 1         | ResNet     |                   |                                    |
| 2         | ResNet     | WarmUp            |                                    |
| 3         | ResNet     | WarmUp Auto-Augment & CutMix |
| 4         | WRN        | WarmUp            | Auto-Augment & CutMix              |
| 5         | PyramidNet | WarmUp            | Auto-Augment & CutMix              |
| 6         | PyramidNet | Ranger            | Auto-Augment & CutMix              |
| 7         | PyramidNet | Ranger Auto-Augment & CutMix &TTA |

In each group, batch size is 128, base learning rate is 0.0003, trained by Adam for 300 epochs. The operations of data preprocessing are follows: Resize(36×36) => RandomCrop(32×32) => RandomHorizontalFlip(50%) => RandomErasing(probability: 50%, max-proportion: 40%). For WarmUp strategy, each group using it should be pre-training with an initial learning rate of 0.1 for 10 epochs.

4.3. Result and Discussion
Accuracy and F1-score were taken as the evaluation index of model performance, which are defined as following formula.

\[
\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}
\]

\[
\begin{align*}
\text{precision} &= \frac{TP}{TP + FP} \\
\text{recall} &= \frac{TP}{TP + FN} \\
\text{F1-score} &= \frac{2 \cdot \text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \tag{4}
\end{align*}
\]
Where $TP$ presents the number of true positive samples, $TN$ presents the number of true negative samples, $FP$ presents the number of false positive samples, $FN$ presents the number of false negative samples. The result of the experiments is shown in the figure 2.

![Figure 2. Result for image classification on Fashion-MNIST.](image)

Through experimental comparison and analysis, it is clear to see the following point of view. First, the ResNet with no learning strategy and data augmentation (Group No.1) can achieve 94.21% accuracy and 94.18% F1-score, and shows a strong ability in DNN training. Next, compared with ResNet, the accuracy of WRN and PyramidNet is improved by 0.38% and 0.96% respectively, which indicates that increasing the network width and channel number can effectively improve the model performance. Furthermore, data enhancement benefits a lot. After using Auto-Augment and CutMix, the accuracy of ResNet is improved by 0.21%, which means that the data enhancement method can effectively avoid over-fitting problem and make the model perform better in the test set. At the same time, the training strategy is also very important. In the initial stage of training, the WarmUp strategy can avoid over-fitting and improve the stability of the model. The Ranger optimizer with LookAhead mechanism can converge faster in the classification task and improve the accuracy. Last but not least, Using TTA technology can better integrate the advantages of the above optimization methods, and further improve the performance of the model to 96.21% accuracy (Group No.7).

5. Conclusion
In this paper, ResNet, WRN and PyramidNet were designed for image classification tasks on Fashion-MNIST dataset using depth residual network, some efficient data enhancement methods and model learning strategies are combined. Through a series of experiments, it was found that increasing the network width and number of channels could effectively improve the model performance, and data enhancement and learning strategies also had a greater impact on the model performance. Finally, a good model was obtained, which is based on PyramidNet, using Ranger, Auto-Augment, CutMix and TTA technology, and its accuracy can reach 96.21%. Whether this conclusion is applicable to other datasets still needs further study.

References
[1] Deng L 2014 A tutorial survey of architectures, algorithms, and applications for deep learning *APSIPA Transactions on Signal and Information Processing* 3
[2] Wu Z, Shen C and van den Hengel A 2019 Wider or Deeper: Revisiting the ResNet Model for Visual Recognition *Pattern Recognition* 90 119–33
[3] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V and Rabinovich A 2015 Going deeper with convolutions 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
[4] He K, Zhang X, Ren S and Sun J 2016 Deep Residual Learning for Image Recognition 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
[5] Liu W, Wu L, Xu Y and Wang D 2019 Deep Residual Equivariant Mapping for Multi-angle
Face Recognition  

Biometric Recognition 145–54

[6] Wu Z, Shen C and van den Hengel A 2019 Wider or Deeper: Revisiting the ResNet Model for Visual Recognition *Pattern Recognition* **90** 119–33

[7] Xia K-J, Yin H-S and Zhang Y-D 2018 Deep Semantic Segmentation of Kidney and Space-Occupying Lesion Area Based on SCNN and ResNet Models Combined with SIFT-Flow Algorithm *J. Med. Syst.* **43** 2

[8] Xiao H, Rasul K and Vollgraf R 2017 Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms

[9] Que Y and Lee H J 2018 Densely Connected Convolutional Networks for Multi-Exposure Fusion 2018 *International Conference on Computational Science and Computational Intelligence (CSCI)*