Practice of Gesture Recognition Based on Resnet50

Zhiming Li *
Center for Information Technology, Hexi University, Zhangye 734000, Gansu, China

*Corresponding author: zhangyelzm@163.com

Abstract. This paper mainly describes and analyzes the common network models of deep learning. By comparing the existing network structure in the industry, this paper focuses on the static gesture recognition algorithm based on the residual network resnet50. In this paper, we choose the depth residual network resnet50 to realize the gesture control of web browser, and apply the gesture recognition method proposed in this paper.

Keywords: Gesture Recognition, Deep Learning, Static Gesture, Convolutional Neural Network

1. Introduction
In recent years, AI has gradually come into the public's vision. Artificial intelligence redefines the interaction between human and machine, making the communication between human and machine more convenient. From the long history of science and technology development, we can find that the key inducement of every change is almost interactive experience Innovation: from mouse and keyboard in PC era to touch in mobile Internet era. As a common body language and a natural way of interaction in daily life, gesture plays an irreplaceable role in human-computer interaction. Compared with traditional recognition methods such as Hu moment and hog, which have low accuracy and low robustness, deep learning model can simulate human brain, non-linear modeling, dynamic understanding and recognition gesture, with high accuracy and robustness [1].

2. Convolution Neural Network and Its Properties
Different from the traditional artificial neural network, convolutional neural network has more network layers. The neurons of convolution layer can extract higher-level features from the previous input layer to get the feature map, reduce the dimension and data redundancy through the convergence layer, and finally classify and identify through the full connection layer and classifier[2]. The research of gesture recognition in this paper is mainly based on convolutional neural network, and its properties are as follows: Network connection. Feature extraction and Biological similarity.

3. Gesture Recognition Based On Resnet50

3.1. Technical Route
Because the network is easy to degenerate when the network depth increases, and the high complexity of artificial feature extraction is difficult to avoid in the traditional deep learning gesture method, and
the convolution neural network has the advantages of weight sharing, which can greatly reduce the computational complexity and high accuracy, this paper uses the depth residual network model based on resnet50 for gesture recognition [3].

First, the weight parameters of the model are pre trained on the source data, then the residual network layer of the new model is initialized with the obtained model weight parameters, then the final model is obtained by fine tuning training on the target data set, and finally the performance of the model is verified on the test set [4]. Resnet50 uses the convolution layer and residual network in front of it to extract the features of the picture (similar to the hog feature, but the effect is better than the hog feature), and then uses the cross entropy loss function to update the network parameters, so that RESNET network can predict the categories of the image, that is, input a 225 * 225 3-channel color image, and output the picture categories[5].

3.2. Residual
In order to solve the problem of gradient degradation, this paper introduces the method of recurrent learning, which is completed by constructing a recurrent block. The cross layer connection in this module is shortcut, which can transfer the input across layers and add the output after convolution, which can effectively improve the training accuracy. It is the introduction of this module that makes the deeper network possible, and the shortcut enables the gradient to be directly propagated back to the front layer[6].

The shortcut connection does not generate additional parameters or increase complexity, and its essence is to map equally. The function to be learned at this time is $H(x) = x$, But this function is a potential function, it is difficult to fit it. The output of the residual network is assumed to be $H(x)$. The output after convolution is $F(x)$, then $H(x) = F(x) + x$, $F(x) = \left( \omega, \delta \left( \omega, \delta (\omega, x) \right) \right)$, where $\omega$ is convolution operation, $\delta$ is convolution operation. IF $F(x) = 0$, the identity mapping function mentioned above is constructed $H(x) = x$. So the problem is transformed into learning a residual function that is easy to fit $F(x) = H(x) - x$. The experiment shows that only when the residual block is no less than two layers can it improve the effect.

3.3. Identity Block
Identity block is a standard module in RESNET, which is characterized by the same dimension of input activation value (a [1]) and output activation value (a [1 + 2]). There are conv2d and relu operations in each layer. In order to speed up the training, the batch norm layer is added. Although this makes the execution process more complicated, it only needs a few lines of code in keras.

The identity module can be implemented in four steps, as shown in Figure 1.

![Identity Block Flowchart](image)

Fig 1. Identity block flowchart

3.4. Revolutionary Block
When the dimensions of input and output are different, we need to use the revolutionary block to solve this problem. The biggest difference between it and the identity block is that the conv2d operation is
added in the shortcut loop, and the input x can be changed into the dimension needed in the subsequent addition operation. The specific flow chart is as follows:

![Flow chart of the revolutionary block](image)

**Fig2.** Flow chart of the revolutionary block

3.5. Batch Norm

In order to study the distribution of training data and achieve good generalization effect in the test set, it is necessary to normalize the data in CV. However, if each batch input data has different distribution, it will obviously bring difficulties to the network training. On the other hand, the data distribution is also changing after layers of network calculation, which is called internal variable shift [7]. Covariate shift mainly describes the difference in distribution between training data and test data, which has an impact on the generalization and training speed of the network. We often use the method of normalization or whitening. Batch norm is a normalization algorithm which is often used to accelerate neural network training and speed up convergence speed and stability. It can be said that it is an essential part of the current depth network[8].

To reduce internal covariate shift, assuming that each layer of the neural network is not normalized, the output data of each layer is normalized to 0 mean and 1 variance, which satisfies the distribution of the positive and the negative[9]. However, there is a problem at this time. The data distribution of each layer is the standard distribution of the positive and the negative, which results in that it can't learn the features of the input data completely. Because the feature distribution of the hard-working learning is normalized, it can directly learn the features of the input data. It is obviously unreasonable to normalize each layer. Therefore, modify it slightly and add the trainable parameters for normalization, that is, batch norm implementation.

3.6. Softmax Classifier

The softmax classifier used in this paper is an algorithm that divides the target variables into multiple classes. It receives the feature matrix of the input of the full connection layer and outputs the probability values of each class corresponding to the input target[10].

4. Implementation of Gesture Recognition System Based On Deep Learning

4.1. Experimental Environment

The hardware environment of this system is Intel Core (TM) i5-8300h CPU @ 2.30ghz four core, memory 8GB, computer built-in camera, pixel 720p. The software environment is Anaconda Spyder 3.3.3, python 3.7, using Python language.

4.2. Test Set Effect

After training, the model is stored in the model folder. After training, the classifier can be loaded directly for prediction.

Call the camera, set the ROI in the display area, only detect the objects in the green box and recognize the gesture category, then operate the browser to complete different operations according to different categories. The recognition rate of the test sample set is shown in Table 1.
Table 1 Test set identification results

| Gesture number | Accuracy | Time consuming /s |
|----------------|----------|-------------------|
| 0              | 0.96875  | 31.49926          |
| 1              | 0.86956  | 48.11863          |
| 2              | 0.84090  | 59.87664          |
| 3              | 0.94285  | 74.32048          |
| 4              | 0.91428  | 40.75666          |
| 5              | 0.95918  | 48.24339          |
| 6              | 0.97014  | 59.51310          |
| 7              | 0.875    | 53.43883          |
| 8              | 1.0      | 75.12285          |
| 9              | 0.98412  | 44.41402          |
| 10             | 1.0      | 100.75803         |
| Average value  | 0.93861  | 57.82381          |

From table 1, it can be seen that the average recognition accuracy of the test sample is 93.861%. Its insensitivity to light and rotation solves the impact of light change and gesture rotation on the recognition rate in the environment. Using for learning and training, the recognition rate is high, simple and fast, with better real-time performance, and the recognition effect is ideal.

4.3. Implementation of Gesture Control System

Implementation idea: after identifying gesture categories, match controllers according to different categories, and complete different operations by running browser through selenium tool test. Table 2 shows the relationship between predefined static gestures and browser page operations.

Table 2 Predefined static gestures and page operations

| Static gesture | Page operations          |
|----------------|-------------------------|
| 0              | Open Firefox browser    |
| 1              | Open Baidu news         |
| 2              | Window maximizing        |
| 10             | Window restore           |
| 8              | Window slide to bottom   |
| 5              | Window slide to top      |
| 6              | Open next page           |
| 7              | Open previous page       |
| 9              | Close Firefox browser    |

5. Summary

This paper mainly describes and analyzes the commonly used network model of deep learning. By comparing the existing network structure in the industry, this paper focuses on the static gesture recognition algorithm based on the residual network resnet50. In this paper, we choose the depth residual network resnet50 to implement the gesture control of web browser, and apply the gesture recognition method proposed in this paper. The idea of implementing gesture control browser is gesture matching controller, which runs and controls the browser through selenium call, a tool for web application testing. In this paper, eight control gestures are defined to realize the functions: open Firefox browser, open Baidu news, maximize the window, restore the window, slide the window to the bottom, open the next page, open the previous page, and close Firefox browser. Through the test, after the gesture control of the browser, the given eight control gestures can effectively control the browser, which proves the feasibility of the method.
Reference
[1] Bindu Verma, Ayesha Choudhary: Grassmann manifold based dynamic hand gesture recognition using depth data. Multimedia Tools and Applications 79 (3), 2213-2237(2020).
[2] C. Bhuvaneshwari, A. Manjunathan: Advanced gesture recognition system using long-term recurrent convolution network. Materials Today: Proceedings 21(Pt 1), 731-733(2020).
[3] Wang Caiyue, Zhang Zhiyi, Xi Zhao: A Human Body Based on Sift-Neural Network Algorithm Attitude Recognition Method. Journal of Medical Imaging and Health Informatics 10(1), 129-133(2020).
[4] Bin Hu, Jiacun Wang: Deep Learning Based Hand Gesture Recognition and UAV Flight Controls. International Journal of Automation and Computing 17 (1), 17-29(2020).
[5] Chengfeng Jian, Junjie Li: Real-time multi-trajectory matching for dynamic hand gesture recognition. IET Image Processing 14(2), 236-244(2020).
[6] Ning-shi Yao,Qiu-yang Tao,Wei-yu Liu,Zhen Liu,Ye Tian,Pei-yu Wang,Timothy Li,Fumin Zhang: Autonomous flying blimp interaction with human in an indoor space. Frontiers of Information Technology & Electronic Engineering 20(01), 45-59(2019).
[7] Prachi Sharma, Radhey Shyam Anand: Depth data and fusion of feature descriptors for static gesture recognition. IET Image Processing 14(5), 909-920(2020).
[8] Taiping Mo, Peng Sun: Research on key issues of gesture recognition for artificial intelligence. Soft Computing 24 (8), 5795-5803(2020).
[9] Linda Nanan Vallée, Sao Mai Nguyen, Christophe Lohr, et al: Human Skeleton Detection, Modeling and Gesture Imitation Learning for a Social Purpose. Engineering 12(02), 90-98(2020).
[10] Gao Yongqiang, Lu Xiong, Sun Junbin, Tao Xianglin, Huang Xiaomei, Yan Yuxing, Liu Jia: Vision-Based Hand Gesture Recognition for Human-Computer Interaction——A Survey. Wuhan University Journal of Natural Sciences 25(02), 169-184(2020).(in Chinese)