Finger Vein Recognition System Based on Convolutional Neural Network and Android

Yangfeng Wang¹*, Tao Chen¹

¹ College of Electronic Information and Automation, Tianjin University of Science and Technology, Tianjin, 300222, China

*Corresponding author’s e-mail: wyfisabeauty@163.com

Abstract. With the rapid development of science and technology, biotechnology has developed rapidly. Among the many biometric technologies, finger vein technology has the characteristics of vitality, portability, and non-replicability, so it is considered to be the most promising biometric technology. However, the accuracy of finger vein recognition is affected by the collection device, the surrounding temperature and the algorithm. The flaws cannot be applied to real life on a large scale. This paper designs a finger vein recognition system based on convolutional neural network and Android, which mainly includes the following three parts. First, the system hardware includes the design of the acquisition device, the selection of the core development board and the display screen. Second, the design of the entire system software architecture is based on the MVVM architecture, which ensures low coupling of the program and is easy for later expansion and maintenance. The software includes collection function, recognition function and administrator function. Finally, a lightweight neural network is proposed for finger vein feature extraction, and proposed a storage method based on MMKV to meet the real-time performance of the system.

1. Introduction

In modern life, identity authentication information has been used in various industries, such as finance, transportation, education, and medical care. However, with the rapid development of electronic information technology and the increase in the scope of people’s activities, identity authentication has become more and more difficult, which is specifically manifested in the deciphering of biometrics and fraudulent use of identity information. The traditional authentication methods mainly include electronic lock, password lock, verification code and RFID card, which are easy to forget, easy to lose, easy to wear and easy to copy. And it has become increasingly difficult to meet people’s needs for ease of use and security in identity authentication. Biometrics technology came into being, including fingerprints, faces, voiceprints, and finger veins. Among the many biometric technologies, finger vein recognition technology has become a major research hotspot due to its vitality, non-reproducibility, and uniqueness.

At present, most finger vein recognition systems can extract insufficient information, and are easily affected by finger rotation, translation, and the light intensity of the acquisition device, resulting in low recognition accuracy. Currently few finger vein recognition systems based on the Android platform are implemented. Based on the existing problems in most systems, this article uses convolutional neural networks for training on the public data set on the basis of convolutional neural networks, and uses convolutional neural networks as feature extractors to design and implemented a finger vein recognition system based on Android. On the one hand, the learning ability of convolutional neural...
network is used to extract richer feature information, which can effectively combat the influence of poor finger vein collection. On the other hand, this system is implemented based on the Android platform. In the case of fewer Android finger vein recognition systems, it is of certain significance to promote the popularization of subsequent finger vein recognition on the Android side.

2. Lightweight convolutional neural network design

2.1. Convolutional Neural Network Theory
With the development of deep learning, convolutional neural networks have been widely used in the computer field. In particular, it has been widely used in various image classification tasks, which has significantly improved the recognition accuracy of classification. It is different from traditional algorithms in that it does not require manual selection of feature extraction methods, but extracts features through convolution kernels. Through the weight sharing mechanism of feature maps, it can achieve the purpose of reducing the number of parameters. The organizational structure of a complete convolutional neural network generally includes an input layer, a convolutional layer(Conv), an excitation layer, a pooling layer, a fully connected layer(FC), and a classification layer[1].

2.2. Proposed network structure
Traditional neural network channels are independent of each other, thus ignoring the connection between channels. This paper designs a new module by fusing ResNet module and SENet module, named SRNet. The entire module is shown in Figure 1. It inherits the advantages of both the SENet module and the ResNet module. On the one hand, it no longer directly passes the input to the next layer, but adds SENet to the residual module to establish the connection between the channels. On the other hand, in the ResNet module, the original two 1 × 1 convolutional layers were removed to reduce the amount of parameters, and a BN layer was added at the same time to normalize the output data, and then integrate with SEnet to achieve the purpose of feature fusion.

![Figure 1 SRNet module](image)

Because the recognition process of finger veins is based on the Android system, and the Android system has limited computing power compared to the computer. However, the calculation process of the convolutional layer needs to run on the Android system, and too many convolutional layers will cause the calculation to be slow. Therefore, the entire network model designed needs to be lightweight, so as to increase the calculation speed of the entire system to meet the real-time performance of the system[2]. In order to meet the real-time performance of the system, and to prevent too few convolutional layers from causing too slow network training and fitting, this paper designs a finger vein network model based on SRNet and AlexNet, named SRAlexNet, which is similar to the AlexNet network[3]. It is composed of multiple different convolutional layers, multiple SRNets, multiple pooling layers, and multiple fully connected layers. The entire network consists of five layers of convolutional layers, three layers of SRNet, three layers of pooling layers, and three layers Fully connected layer composition, its structure is shown in Table 1, In this network, the input is 128 × 128 × 3, and the output is 512 dimensions.
Table 1 SRAlexNet structure

| Layer   | Number of convolution kernels | The size of the convolution kernel | Step size |
|---------|-------------------------------|-----------------------------------|-----------|
| Conv1   | 48                            | $11 \times 11$                     | 4         |
| MaxPool1| 3                             | $3 \times 3$                       | 2         |
| SRNet   | 48                            | $5 \times 5$                       | 1         |
| Conv2   | 128                           | $3 \times 3$                       | 1         |
| SRNet   | 128                           | $5 \times 5$                       | 1         |
| Conv3   | 192                           | $3 \times 3$                       | 1         |
| SRNet   | 192                           | $5 \times 5$                       | 1         |
| Conv4   | 192                           | $3 \times 3$                       | 1         |
| MaxPool3| 192                           | $3 \times 3$                       | 2         |
| SRNet   | 192                           | $5 \times 5$                       | 1         |
| Conv5   | 192                           | $3 \times 3$                       | 2         |
| SRNet   | 192                           | $5 \times 5$                       | 1         |
| FC layer1 | 512                         | $1 \times 1$                      | 1         |
| Dropout1|                               |                                   |           |
| FC layer2 | Class                      | $1 \times 1$                      | 1         |

3. Design and Implementation of Android Software and Hardware

3.1. Android platform
Android is a software platform and operating system based on the Linux kernel. Its architecture is a five-layer structure, namely the Linux kernel layer, the hardware abstraction layer, the system runtime layer, the application framework layer and the application layer.

3.2. System hardware selection

3.2.1 Collection device. The finger vein acquisition device used in this article uses the transmission principle to obtain vein images through the irradiation of near-infrared light. The entire collection device includes a light source module, an imaging module, a filter module, a power supply module, and a display module.

3.2.2 Android development board. The Android development board in this system uses the S5P6818 octa-core processor based on the Cortex-A53 architecture, with a main frequency of 1.4GHZ, which has faster computing performance. In terms of storage, the memory is 2GB of DDR3, and the external storage is 16GB of EMMC. It can meet the storage and reading of high-dimensional characteristics. This Android development board supports the UVC protocol and can be connected to an external USB camera to meet the needs of finger vein information collection. There are three USB ports on the development board, which can meet the power supply requirements of LED lights and infrared cameras at the same time.

3.2.3 Touchable capacitive display. In terms of human-computer interaction, a 7-inch RGB capacitive screen is selected, without the need for an additional keyboard and mouse, to directly complete operations and information input. The physical objects of the acquisition device.

3.3. Software design
The system software part adopts MVVM architecture design. The advantage of adopting MVVM architecture is that the view and business logic can be separated, better decoupling, and convenient for the addition and maintenance of later functions. The Native layer is mainly for image processing-
related operations, the Model layer is mainly for data operations, and the feature vector of finger vein images is extracted through the trained network model. The ViewModel layer is for data processing, calculating the cosine distance between finger vein images, and judging whether the recognition is passed. The View layer is UI interactive display. The overall architecture is shown in Figure 2. Finally, a storage method based on MMKV is used, which is a key-value storage method.

![Figure 2 MMVM module](image)

3.3.1 Network model migration. This article is to realize finger vein recognition on the Android platform. Android has limited computing power. The network training requires GPU. The training process of the model is completed on the PC side. The corresponding network model and weights are saved and cropped, and saved as "pth" format. The saved "pth" file is dependent on the python environment. Because it is more difficult to call python-related files on the Android side, it needs to be converted into a .pt file that does not depend on the python environment. Finally, the Android side extracts finger vein features by loading .pt files. The model acquisition flowchart is shown in Figure 3.

![Figure 3 The model acquisition flowchart](image)

3.3.2 Design and realization of system functions. In the software part, the acquisition, identification and administrator modules are designed. The acquisition module is the user's vein information and saves it locally. The recognition module is to compare the similarity between the registered sample and the sample to be tested, and the recognition can be judged through the set threshold. The administrator module is mainly used to manage the addition and deletion of users. The overall function design of the system is shown in Figure 4 (a), and the physical object of the entire system is shown in Figure 4 (b).
4. Experimental results

4.1. Network model verification

4.1.1 Experimental data. FV-USM data comes from the Malaysian data set[7]. It contains the finger veins of 106 individuals. Each person collected 4 fingers and 6 pictures for each person. There are a total of 636 categories and a total number of 2544 pictures. SD_DB data comes from the Malaysian data set. It contains finger veins of 106 individual. Each person collected 6 fingers and 6 pictures for each person. There are a total of 636 categories and a total number of 3816 pictures.

4.1.2 Training process and results. Select 400 types of pictures in the two data sets, and use SRAlexNet and AlexNet for training respectively. First, the training set and the test set are divided into 4:2. Each picture is normalized to $128 \times 128 \times 3$ after obtaining the ROI area through image processing. Then use offline expansion methods, including rotation, translation, increase noise, to expand the original data set to 94,400 sheets, and finally test on the test set. The accuracy results of the network proposed and other networks are shown in the Table 2.

| Data set  | Network module | Accuracy (%) |
|-----------|----------------|--------------|
| FV-USM    | LBP[6]         | 96.02        |
|           | CNN[7]         | 97.53        |
|           | AlexNet        | 96.08        |
|           | SRAlexNet      | 97.93        |
| SD-DB     | SPF[8]         | 87.00        |
|           | CNN[7]         | 96.62        |
|           | AlexNet        | 95.78        |
|           | SRAlexNet      | 97.71        |

4.2. System verification

4.2.1 Experimental data. The system verification is based on a self-collected data set. A total of 10 people were collected, each of whom collected 6 fingers and 6 pictures for each finger. There are a total of 40 categories and a total of 360 picture.

4.2.2 System real-time verification. The accuracy of finger vein recognition in the entire system is determined by the complexity of the network and the storage method. The recognition rate in this system is 94.53%. This paper proposes a storage method based on MMKV. In order to verify that the access speed of this framework is compared with other frameworks, the time of different access methods is shown in Table 3. In addition, this article also compares the time complexity, forward
propagation time and recognition time of the AlexNet model and the SRAlexNet model. The specific parameters are shown in Table 4.

| Storage framework | MMKV | SharPreference | Sql |
|-------------------|------|---------------|-----|
| Storage time (ms/picture) | 0.029 | 0.033 | 0.63 |
| Read time (ms) | 4.51 | 43.47 | 21 |

Table 3 Different access frame time

| Network module | The size of module(MB) | Flops(M) | Forward propagation time (ms) | Recognition time (ms) |
|----------------|------------------------|----------|-------------------------------|----------------------|
| AlexNet        | 241.86                 | 715.54   | 155                           | 12152                |
| SRAlexNet      | 12.61                  | 96.77    | 24                            | 1609                 |

It can be seen from Table 3 and Table 4 that the SRAlexNet and MMKV-based storage methods proposed in this paper can better meet the real-time performance of the system.

5. Conclusion

This paper builds a complete finger vein recognition system and designs a human-computer interaction interface, including acquisition module, recognition module and administrator module. The model training and image processing process is completed on the PC side, and the network model and weight parameters are cut at the same time. Finally, qualified images are collected through the acquisition device in the Android system, and the network model and weight parameters are loaded for registration and identification.

Acknowledgments

Thanks to everyone who participated in the project.

References

[1] Luo R, Zhang K. Research on Finger Vein Recognition Based on Improved Convolutional Neural Network[J]. International Journal of Social Science and Education Research, 2020, 3(4): 107-114.
[2] Ganesh M, Pednekar P, Prabhuswamy P, et al. CNN-Based Android Malware Detection[C]// 2017 International Conference on Software Security and Assurance (ICSSA). IEEE Computer Soc.
[3] You Y, Zhang Z, Hsieh C J, et al. 100-epoch ImageNet Training with AlexNet in 24 Minutes[J]. Journal of Jinggangshan University, 2016.
[4] Saracino A, D Sgandurra, Dini G, et al. MADAM: Effective and Efficient Behavior-based Android Malware Detection and Prevention[J]. IEEE Transactions on Dependable & Secure Computing, 2018:1-1.
[5] Mishra S M. Android SDK[M]. John Wiley & Sons, Inc, 2015.
[6] Friedman L, Stern H, Price L R, et al. Why Temporal Persistence of Biometric Features is so Valuable for Classification Performance[J]. Sensors, 2020, 20(16): 4555.
[7] G. Hong, M. B. Lee, and K. R. Park, “Convolutional neural network based finger-vein recognition using NIR image sensors,” Sensor, vol. 17, no. 6, p. 1297, 2017.
[8] Fei L, Yin Y, Yang G, et al. Finger vein recognition with superpixel-based features[C]// IEEE International Joint Conference on Biometrics. IEEE, 2014.