Fingerprint Pattern Classification Using Convolution Neural Network

Wang-Su Jeon¹ and Sang-Yong Rhee²

¹Department of IT Convergence Engineering, Kyungnam University, Changwon, Korea

²Department of Computer Engineering, Kyungnam University, Changwon, Korea

Ijfis

Abstract

Biometrics technology determines the correct identity of a person by extracting human biological or behavioral characteristic data. As the possibility of hacking increases with the development of IT technology, interest in biometrics and authentication technology is greatly increasing. Currently, the most popular authentication technology is fingerprint recognition. For the sake of efficiency, fingerprint recognition is divided into two stages. In the first step, the inputted fingerprint image is subjected to a complicated preprocessing stage, and the fingerprint image is then classified. In the second step, the feature points of the classified fingerprints are extracted and compared with the fingerprint feature points stored in a database. Human beings can easily classify fingerprint patterns without complicated image processing. In this paper, we propose the use of a convolution neural network model combined with an ensemble model and a batch normalization technique after minimizing the number of the quality improvement processes required for a fingerprint image, which operates more similarly to human perception.

Keywords: Fingerprint recognition, Fingerprint classification, Batch normalization, Ensemble, CNN

1. Introduction

As the possibility of hacking increases owing to the development of IT technology, security technologies for biometrics continue to develop because existing authentication methods cannot effectively protect the personal information of an individual. Biometrics technology is used to extract human characteristics and behavioral information to determine whether they belong to a particular person. There are two types of such technology. The first is used to authenticate a user through verification with a database using 1:1 matching. The second is used to identify the user through a search of their data in the database using 1:N matching. This type uses the individual's biological characteristics such as iris, face, hand shape, ear shape, fingerprint, gait, and voice. Such biometric information differs from person to person and does not change over the years. There is also an advantage in that there is no risk of memorization or exposure such as with other existing authentication methods. Of the various types of biometric authentication services, fingerprint recognition has been most widely commercialized.

Contemporary fingerprint comparison technology began in 1684 when Neemia Couture of England first discovered that human fingerprints differ from each other. Later, Henry first

Received: Aug. 2, 2017
 Revised : Sep. 11, 2017
 Accepted: Sep. 20, 2017

Correspondence to: Sang-Yong Rhee
 (syrrhee@kyungnam.ac.kr)
 ©The Korean Institute of Intelligent Systems

©This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

presented a taxonomy of fingerprints in 1900. The classification of fingerprints is based on features such as ridges, endings, bifurcations, deltas, and cores, and according to the flow, whorl, right loop, double loop, left loop, and arch. Korea has about 41 million fingerprints images in its database. To find a match out of such a large number of fingerprint images, an accurate classification of the images is required.

Kang et al. [1, 2] proposed an effective preprocessing method for determining a threshold value using a neural network or a neuro-fuzzy method for the histogram extracted from a fingerprint image. Jeong and Lee [3] classified fingerprint images with 88% accuracy while modifying the Markov model value using a genetic algorithm after extracting the ridge information in 16 directions using the Markov model for preprocessed fingerprint images.

The biggest problem with existing researches is the thinning required to obtain ridge direction information. Thinning causes noise such as twigs that were not in the original image. To remove such noise, additional image processing should be applied. In this study, we try to classify fingerprints without a thinning process using a convolution neural network (CNN).

In 2012, Krizhevsky et al. [4], proposed by the Supervision team led by Hinton, won the ImageNet Large Scale Visual Recognition Competition (ILSVRC) by a large gap of 16% over the second-place team. After that, deep learning began to attract attention, and has been used in various fields such as recognition and classification. Wang et al. [5] proposed a fingerprint classification method using the deep learning method. Bae et al. [6] classified fingerprints through an improved version of the neuro-fuzzy technique. Peralta et al. [7] conducted a study combining an ensemble technique using an improved model of Alexnet. Other various studies have also been conducted.

The proposed system is shown in Figure 1. Preprocessing is applied to an input fingerprint image to improve the quality [8]. Fingerprint classification is then conducted through the extraction and learning of image features using a CNN.

The rest of this paper is composed as follows: In Section 2, we describe existing fingerprint classification methods. In Section 3, we describe the learning method, the models used, and the techniques of VGGNet [9], which is a type of CNN. Section 4 describes experiments conducted on fingerprint recognition and an analysis of the results. Section 5 provides some concluding remarks.

Figure 1. System structure.

2. Conventional Fingerprint Classification Method

The existing fingerprint classification method is as follows. To improve the quality of a fingerprint image, the noise is removed through normalization, and the fingerprint image area is isolated from the background using a segmentation technique. Then, binarization is applied based on the threshold value, and the image is enhanced using a Gabor or similar type of filter. Next, ridges 1 pixel thick are made using a thinning process. Finally, the types and positions of the minutiae are determined, and the fingerprints are classified by analyzing the directions of the entire ridges.

During the thinning process, unnecessary deformation of the ridges may occur, such as cut, twig, circular, or cross ridges, as shown in Figure 2. To remove these deformities, the end points are traced at the bifurcation point based on the angle between the end points and the average angle. Using certain criteria, cut ridges are then connected, twig ridges are removed, a part of the circular ridges is moved, and cross ridges are disconnected. Figure 3 shows the results of removing unnecessary ridges. Although these processes take a significant amount of time, human being can conduct this fingerprint classification without a thinning or post process. The purpose of this study is to implement the characteristics of human beings into the classification process.

3. Proposed Fingerprint Classification Method

3.1 Fingerprint Image Preprocessing

Preprocessing is also applied in the proposed method. Human beings can be said to perform image processing as determined

Figure 2. Unnecessary deformation of the ridges. (a) Cut, (b) twig, (c) circle, (d) cross.

Figure 3. Noise removal results.

Figure 4. Fingerprint image enhancement (a) before and (b) after preprocessing.

by the illusion of people. An image preprocessing allows features to be accurately extracted using a CNN. An input image is normalized and segmented by calculating the directionality of the ridges. Binarization and Gabor filtering are then applied. The thinning process is deemed unnecessary based on the drawbacks described in Section 2 and is excluded. An input image and the processing results are shown in Figure 4.

3.2 Ensemble Learning

To improve the recognition rate of an image in a CNN, the more characteristic information that is available the better. One enhancement method used is the ensemble technique [10]. As shown in Figure 5, the ensemble technique is a method for pre-

Figure 5. An ensemble model.

dicting a result using differently generated features and combining them in a plurality of models, thereby achieving a better performance than using a single model. However, this technique has certain disadvantages such as an over fitting, and a slower learning speed owing to the increase in the number of parameters. To overcome these drawbacks, we use a batch normalization method [11].

3.3 Batch Normalization

When learning a neural network, the data are usually used in mini-batch units. The mean and standard deviation of each feature are obtained and normalized between zero and 1, and then converted into new values using a scale factor and a shift factor. This method is called batch normalization. When actually used, as shown in Figure 6, a batch normalization layer is added before entering a specific hidden layer, and the input values are then modified and put into the activation function.

Batch normalization plays a role similar to the existing regularization, and has the advantage of solving the over fitting phenomenon. The learning rate can also be set to a large value, and the learning speed can be improved.

3.4 Learning Using a CNN

The CNN model used in this study is VGGNet. The University of Oxford's VGG team first discovered the concept of factorizing convolution, which is better at stacking 3×3 convolutions more deeply than 5×5 and 7×7 convolutions, and developed

Figure 6. Neural network using batch normalization.

Figure 7. Factorizing convolution.

VGGNet based on this concept. Factorizing convolution has the advantage of reducing the number of parameters and extracting good features over existing large size filters.

Factorization convolution is shown in Figure 7. We compared three different models, VGGNet (model 1), VGGNet with pre-processing (model 2), and VGGNet with batch normalization and ensemble-based models (model 3).

Table 1 shows the CNN structure used for the models 1 and 2, when applying basic VGGNet. VGGNet has a simple structure and receives an input of 224×224 in size. After applying zero padding to the input image, convolution is calculated using a 3×3 filter with a stride of 1 twice, and the output is calculated in the same way using a filter size of 2×2 with a stride 2.

Between this calculations, the convolution is applied three times using a filter size of 3×3 with a stride of 1, and the pooling method is repeated three times. The output to the last

Table 1. Basic VGGNet structure (model 1 and 2)

Type	Filter size / stride	Input size
Conv	$3 \times 3 / 1$	$224 \times 224 \times 64$
Conv	$3 \times 3 / 1$	$224 \times 224 \times 64$
Pooling	$2 \times 2 / 2$	$112 \times 112 \times 64$
Conv	$3 \times 3 / 1$	$112 \times 112 \times 128$
Conv	$3 \times 3 / 1$	$112 \times 112 \times 128$
Pooling	$2 \times 2 / 2$	$56 \times 56 \times 128$
Conv	$3 \times 3 / 1$	$56 \times 56 \times 256$
Conv	$3 \times 3 / 1$	$56 \times 56 \times 256$
Conv	$3 \times 3 / 1$	$56 \times 56 \times 256$
Pooling	$2 \times 2 / 2$	$28 \times 28 \times 256$
Conv	$3 \times 3 / 1$	$28 \times 28 \times 512$
Conv	$3 \times 3 / 1$	$28 \times 28 \times 512$
Conv	$3 \times 3 / 1$	$28 \times 28 \times 512$
Pooling	$2 \times 2 / 2$	$14 \times 14 \times 512$
Conv	$3 \times 3 / 1$	$14 \times 14 \times 512$
Conv	$3 \times 3 / 1$	$14 \times 14 \times 512$
Conv	$3 \times 3 / 1$	$14 \times 14 \times 512$
Pooling	$2 \times 2 / 2$	$7 \times 7 \times 512$
FC layer	-	$1 \times 1 \times 4096$
FC layer	-	$1 \times 1 \times 4096$
FC layer	-	$1 \times 1 \times 5$
Softmax	Classifier	$1 \times 1 \times 5$

ReLU layers are omitted from the table for brevity.

pooling is 7×7 in size. The output is used as the input of the fully connected layer, and the probability of each class when outputting five outputs during the last output is calculated using Softmax.

As shown in Table 2, the disadvantages of the ensemble technique can be minimized when adding batch normalization between the convolution operation and the activation function operation using the structure shown in Table 1.

4. Experiments and Analysis of the Results

4.1 Experiment Environment

The data used in this paper were supplied by the Fingerprint Verification Competition (FVC) 2000, 2002, 2004 [12–14] (Figure 8). The types of fingerprint images are classified into five categories, namely, whorl, right loop, double loop, left loop, and arch. Details of the images are shown in Table 3 and Figure 9. The database consists of 788 learning images and 100

Table 2. Improved VGGNet for ensemble (model 3)

Type	Filter size / stride	Input size
Conv	3×3 / 1	224×224×64
Batch norm	-	-
Conv	3×3 / 1	224×224×64
Batch norm	-	-
Pooling	2×2 / 2	112×112×64
Conv	3×3 / 1	112×112×128
Batch norm	-	-
Conv	3×3 / 1	112×112×128
Batch norm	-	-
Pooling	2×2 / 2	56×56×128
Conv	3×3 / 1	56×56×256
Batch norm	-	-
Conv	3×3 / 1	56×56×256
Batch norm	-	-
Conv	3×3 / 1	56×56×256
Batch norm	-	-
Pooling	2×2 / 2	28×28×256
Conv	3×3 / 1	28×28×512
Batch norm	-	-
Conv	3×3 / 1	28×28×512
Batch norm	-	-
Conv	3×3 / 1	28×28×512
Batch norm	-	-
Pooling	2×2 / 2	14×14×512
Conv	3×3 / 1	14×14×512
Batch norm	-	-
Conv	3×3 / 1	14×14×512
Batch norm	-	-
Pooling	2×2 / 2	7×7×512
FC layer	-	1×1×4096
FC layer	-	1×1×4096
FC layer	-	1×1×5
Softmax	Classifier	1×1×5

ReLU layers are omitted from the table for brevity.

test images.

The experiment environment used for the learning and testing is as follows: Linux Ubuntu 16.02.2 was used for the operating system. The hardware consists of an Intel i7-6770k CPU, 32

Figure 8. Fingerprint Verification Competition (FVC) 2000, 2002, 2004. **Figure 9.** Types of fingerprints: (a) whorl, (b) right loop, (c) double loop, (d) left loop, (e) and arch.**Table 3.** Type and number of fingerprints

Fingerprint type	Number of images
Whorl	176
Right loop	208
Double loop	164
Left loop	200
Arch	140

GB memory, two NVIDIA Maxwell TITAN X GPUs, and the Tensorflow 1.1.0 deep running framework.

Three types of CNN models, basic VGGNet (model 1), VGGNet using preprocessed images (model 2), and VGGNet combining ensemble techniques (model 3), were learned and their performance compared. A fingerprint image with a size of

Table 4. Hyper parameter setting

	Model 1	Model 2	Model 3
Epoch	200	200	200
Learning rate	0.001	0.001	0.01
Batch size	64	64	16
Optimizer	SGD	SGD	SGD

Table 5. Model performance evaluation

	Model 1	Model 2	Model 3
Image size	224×224	224×224	224×224
Accuracy	82.1%	94.2%	98.3%
Training time	8H 21M	8H 48M	10H 2M

350 × 348 was reduced to 224 × 224. The hyper parameters used for the learning are shown in Table 4.

The number of epochs was set to 200, and the learning rate was set to 0.001; however, in case of the model 3, it was set to 0.01 because this model uses batch normalization. The batch size was set to 64; however, because model 3 uses both ensemble and batch normalization, we set its batch size to 16 to reduce memory usage. The optimizer uses SGD.

4.2 Experiment Results and Analysis

The performance evaluation results using the three models proposed in this paper are shown in Table 5. For model 1, the learning time was the shortest but the accuracy was the lowest. For model 2, we can see that the accuracy is higher than the learning time. Model 3 has a slightly different performance than model 2, but has a long learning time. For model 3, an accuracy level of 97.1% was obtained before the ensemble was applied, and an accuracy level of 98.3% was achieved after the ensemble was applied. Considering the speed of learning and the performance improvement, model 3, which showed a 98.3% recognition rate, was deemed the best model.

Table 6 shows the recognition results for each type when applying model 3. Twenty fingerprint images were randomly selected five times for each type of fingerprint image and tested, resulting in a total of 100 images. The test results show that the classification rates for left loop, double loop, and right loop are slightly lower. An example of an image misclassified during the test is shown in Figure 10. Because the center point cannot be found in Figure 10(a), it was misclassified as a whorl despite originally being a left and right loop type. Because of the

Table 6. Confusion matrix of model 3

	W	R	D	L	A
W	99	0	1	0	0
R	4	96	0	0	0
D	0	0	98	2	0
L	5	1	0	94	0
A	1	0	0	0	99

Figure 10. Example of false acceptance and rejection: (a) left and right loop and (b) double loop.

contamination of the center, the image is difficult to classify by the human eye. The image shown in Figure 10(b) is a double loop, but was classified as a left loop because it has a pattern similar to this type of loop.

5. Conclusions

In this study, three models were created through a modification of the VGGNet structure, and fingerprint classification was applied using each model. In this study, we found that the preprocessing of a fingerprint is indispensable after comparing models 1 and 2. Model 3 has a small batch size but is fast and shows an excellent performance. Using model 3, the average classification rate was 97.2%.

The important aspect of fingerprint classification is a fast matching speed. In future research, we will study CNN structure optimization, and speed up the learning speed for fast matching. In addition, we will investigate methods to improve the classification performance of images with noise.

Conflict of Interest

No potential conflict of interest relevant to this article was reported.

References

- [1] J. Y. Kang, J. S. Lee, J. H. Lee, S. M. Kong, D. H. Kim, and S. B. Lee, "A study on the dynamic binary fingerprint recognition method using artificial intelligence," *Journal of Korean Institute of Intelligent Systems*, vol. 13, no. 1, pp. 57-62, 2003. <https://doi.org/10.5391/JKIS.2003.13.1.057>
- [2] W. J. Kim, C. G. Lee, Y. T. Kim, and S. B. Lee, "Implementation of embedded system and finger print identification using ART2," *Proceedings of KIFS Spring Conference*, vol. 16, no. 1, pp. 90-93, 2006.
- [3] H. W. Jung and J. H. Lee, "Various quality fingerprint classification using the optimal stochastic models," *Journal of the Korea Society for Simulation*, vol. 19, no. 1, pp. 143-151, 2010.
- [4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proceeding of Neural Information Processing Systems Conference and Workshop*, Lake Tahoe, NV, 2012, pp. 1097-1105.
- [5] R. Wang, C. Han, Y. Wu, and T. Guo, "Fingerprint classification based on depth neural network," 2014, Available: <https://arxiv.org/abs/1409.5188>
- [6] J. S. Bae, S. K. Oh, and H. K. Kim, "Design of fingerprints identification based on RBFNN using image processing techniques," *The transactions of The Korean Institute of Electrical Engineers*, vol. 65, no. 6, pp. 1060-1069, 2016. <https://doi.org/10.5370/KIEE.2016.65.6.1060>
- [7] D. Peralta, I. Triguero, S. Garcia, Y. Saeys, J. M. Benitez, and F. Herrera, "On the use of convolutional neural networks for robust classification of multiple fingerprint captures," 2017, Available: <https://arxiv.org/abs/1703.07270v3>
- [8] H. I. Kim, D. S. An, and C. W. Ryu, "Fingerprint recognition," in *Proceeding of the Biometric Consortium Conference*, 2001.
- [9] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large scale image recognition," 2015, Available: <https://arxiv.org/abs/1409.1556>
- [10] J. Wang, S. Xu, B. Duan, C. Liu, and J. Liang, "An ensemble classification algorithm based on information entropy for data streams," 2017, Available: <https://arxiv.org/abs/1708.03496>
- [11] S. Ioffe and C. Szegedy, "Batch normalization: accelerating deep network training by reducing internal covariate shift," in *Proceeding of the 32nd International Conference on Machine Learning*, Lille, France, 2015, pp. 448-456.
- [12] Fingerprint Verification Competition, "FVC 2000," Available: <http://bias.csr.unibo.it/fvc2000/>
- [13] Fingerprint Verification Competition, "FVC 2002," Available: <http://bias.csr.unibo.it/fvc2002/>
- [14] Fingerprint Verification Competition, "FVC 2004," Available: <http://bias.csr.unibo.it/fvc2004/>

Wang-Su Jeon received his B.S. degrees in Computer Engineering from Kyungnam University, Masan, South Korea, in 2016, and is currently pursuing the M.S. degree in IT Convergence Engineering at Kyungnam University, Changwon, Korea.

His present interests include computer vision, pattern recognition and machine learning.

E-mail : jws2218@naver.com

Sang-Yong Rhee received his B.S. and M.S. degrees in Industrial Engineering from Korea University, Seoul, Korea, in 1982 and 1984, respectively, and his Ph.D. degree in Industrial Engineering at Pohang University, Pohang, Korea. He is currently a professor at the Computer Engineering, Kyungnam University, Changwon, Korea. His research interests include computer vision, augmented reality, neuro-fuzzy and human-robot interface.

His present interests include computer vision, augmented reality, neuro-fuzzy and human-robot interface.

E-mail : syrhee@kyungnam.ac.kr