2016 23rd International Conference on Pattern Recognition (ICPR)
Cancún Center, Cancún, México, December 4-8, 2016

A Novel Fingerprint Classification Method Based on Deep Learning

Ruxin Wang
School of Mathematical Science
University of
Chinese Academy of Sciences (UCAS)
Beijing 100049
Email: wangruxin12@mails.ucas.ac.cn

Congying Han
School of Mathematical Science, UCAS
Key Laboratory of Big Data Mining
and Knowledge Management, UCAS
Email: hancy@ucas.ac.cn

Tiande Guo
School of Mathematical Science, UCAS
Key Laboratory of Big Data Mining
and Knowledge Management, UCAS
Email: tdguo@ucas.ac.cn

***Abstract*—Fingerprint classification is an effective technique for reducing the candidate numbers of fingerprints in the stage of matching in automatic fingerprint identification system (AFIS). In recent years, deep learning is an emerging technology which has achieved great success in many fields, such as image processing, computer vision. In this paper, we have a preliminary attempt on the traditional fingerprint classification problem based on the new depth neural network method. For the four-class problem, only choosing orientation field as the classification feature, we achieve 91.4% accuracy using the stacked sparse autoencoders (SAE) with three hidden layers in the NIST-DB4 database. And then two classification probabilities are used for fuzzy classification which can effectively enhance the accuracy of classification. By only adjusting the probability threshold, we get the accuracy of classification is 96.1% (setting threshold is 0.85), 97.2% (setting threshold is 0.90) and 98.0% (setting threshold is 0.95) with a single layer SAE. Applying the fuzzy method, we obtain higher accuracy.**

I. INTRODUCTION

With the maturing of fingerprint identification technology and the expanding of fingerprint database, the accuracy and speed of fingerprint identification are required higher and higher. Fingerprint classification plays an important role in automatic fingerprint identification system (AFIS). According to Henry [1], the fingerprint is divided into five categories: the left loop, right loop, arch, tented arch and the whorl.

The classification tasks usually include two main stages: feature extraction and classification. In the past decades, many fingerprint classification algorithms have been proposed, incorporating different machine learning methods [2]. However, singular points' information is usually considered for many methods and it directly affects the final classification results. The error information of singular points often leads to the failure of classification. How to not utilize the singular points directly still can obtain a desirable classification result? Most recently, deep learning has achieved great success in other computer vision tasks. Through this kind of layered structure which has greater representational power, much more complex features can be obtained. Hinton [3] used the restricted Boltzmann machines (RBMS) to learn low-dimensional codes that work much better than PCA. Similarly, Pascal Vincent et al. [4] proposed the stacked denoising autoencoders model and used it for handwritten recognition. Nowadays, these deep learning methods [5] have been applied on a broader scale, such as visual tasks [6], natural language processing [7], and artificial intelligence [8]. Meanwhile, deep learning methods have also begun to play a role in some traditional fingerprint recognition problems, such as minutiae extraction [9], orientation field estimation [10].

In this paper, we adopt a deep network structure, the stacked sparse autoencoder (SAE) neural network (containing three hidden layer) [11], to learn a low-dimensional representation (i.e. features) of the input data, then the features obtained by SAE via a trained multi-classifier for fingerprint classification. In order to further improve the accuracy of the classification, a simple and effective fuzzy classification method is proposed. Section II gives a brief introduction about the SAE. And in section III, we introduce the softmax regression. Section IV presents experimental results for classification and fuzzy classification. Section V gives the conclusion for the paper.

II. UNSUPERVISED FEATURE SELF-TAUGHT LEARNING

Learning algorithm based on deep structure has been applied on a broader scale, especially in computer vision and image processing. At the same time, the related theory of deep learning has also been developed and updated. The sparse auto-encoder (sparse-AE) is one approach to automatically learn features from unlabeled data.

A standard sparse-AE is a 3-layer neural network comprising an input, hidden and output layer. It sets the target values to be equal to the inputs in network terminal. For the input training samples $\{x_i\}_1^m$, the cost function is as follows:

$$L(W, W', b, b') = \min_{W,W',b,b} \sum_i ||x_i - f(W'h_i + b')||_2^2 \\ +\lambda(||W||_F^2 + ||W'||_F^2) + \beta sparse(.) \qquad (1)$$

Where $h_i = f(Wx_i + b)$ represents the output of the network's hidden layer, $f(.)$ is a group of non-linear mapping functions with parameters (W, W') and (b, b'). The sparsity constraint $sparse(.)$ is employed in order to reduce the dimension and extract the more meaningful elements of the input data. $sparse(.)$ usually chooses the Kullback-Leibler

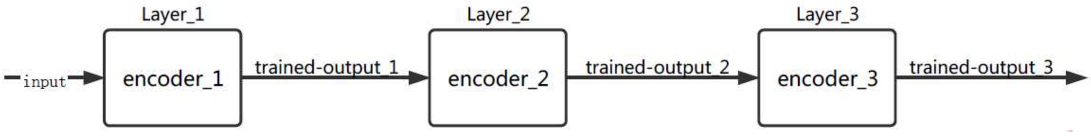


Fig. 1. a simplified structure chart of SAE with three layers.

(KL) divergence between the average activation of hidden unit and our desired level of sparsity.

$$KL(\rho||\hat{\rho}_j) = \rho \log \frac{\rho}{\hat{\rho}_j} + (1 - \rho \log \frac{1-\rho}{1-\hat{\rho}_j}). \quad (2)$$

where ρ is a sparsity parameter, typically a small value close to zero, $\hat{\rho}_j$ denotes the average activation of hidden unit j.

The SAE model is a neural network consisting of multiple layers of sparse auto-encoders in which the outputs of each layer is wired to the inputs of the successive layer (Figure.1). The basic unit of the SAE is the sparse-AE model.

The SAE can effectively learn a low-dimensional representation of the input data. The parameters of stacked structure are learned using greedy layer-wise training. Namely, use the output of each layer as input for the subsequent layer. At the same time, parameters can be optimized by a fine tuning strategy which treats all layers as a single model using the back propagation (BP) algorithm. So in one iteration, the parameters can be improved [11].

For fingerprint classification, a relatively small number of features extracted from fingerprint images. In particular, almost all the methods are based on one or more of the following features: ridge line flow, orientation image, singular points, and Gabor filter responses. Here, we only choose the orientation field as our classification feature and no additional processing is done for the original orientation field. There are some reasons for choosing input feature: the first one is that the orientation field belongs to global feature of a fingerprint, which can effectively express the ridge flow pattern and the ridge type. Secondly, due to the effect of detecting the number and the location of singular points, the classification result is quite sensitive, such as wrong in detecting the number of singular points would lead to misclassification. Thirdly, the Gabor filter responses are usually employed as a coding technique of center point by FingerCode-based method [12]. In conclusion, only the orientation information is chosen as the input feature for learning.

A. Computing the orientation field

Computing the orientation field is a crucial step for constructing an AFIS. In this paper, Rao's method [13] is exploited for computing the initial orientation field. The main steps are as following:

a) Divide image G into blocks of size $w \times w$, we choose $w = 20$ in our algorithm.

b) Compute the gradients $G_x(i,j)$ and $G_y(i,j)$ at each pixel (i,j).

c) Estimate the local orientation of each block centered at pixel using the following equations:

$$[G_{Bx}, G_{By}]^T_{(i,j)} = [\sum_{i=1}^{w}\sum_{j=1}^{w} G_{Sx}(i,j), \sum_{i=1}^{w}\sum_{j=1}^{w} G_{Sy}(i,j)]^T, \quad (3)$$

$$\theta_{ij} = \frac{1}{2}\pi + \frac{1}{2}tan^{-1}(\frac{G_{By}}{G_{Bx}}), \quad (4)$$

where

$$\begin{pmatrix} G_{Sx}(i,j) \\ G_{Sy}(i,j) \end{pmatrix} = \begin{pmatrix} G_x(i,j)^2 - G_y(i,j)^2 \\ 2G_x(i,j)^2 G_y(i,j)^2 \end{pmatrix}. \quad (5)$$

d) Adopt a local gaussian filter for smoothing.

In the stage of learning and testing, the orientation field θ are mapped into a continuous vector field $(\sin 2\theta, \cos 2\theta)^T$ in order to solve the ambiguity of θ and $\theta + \pi$.

B. Showing study results

According to the hypothesis of the SAE, learning process aims to learn an approximation identity function, so as to the output that is similar to the input. So the obtained features through SAE as a low-dimensional representation of the input data can effectively reconstruct the input orientation field. From Figure.2 to Figure.3 exhibit some reconstructed results.

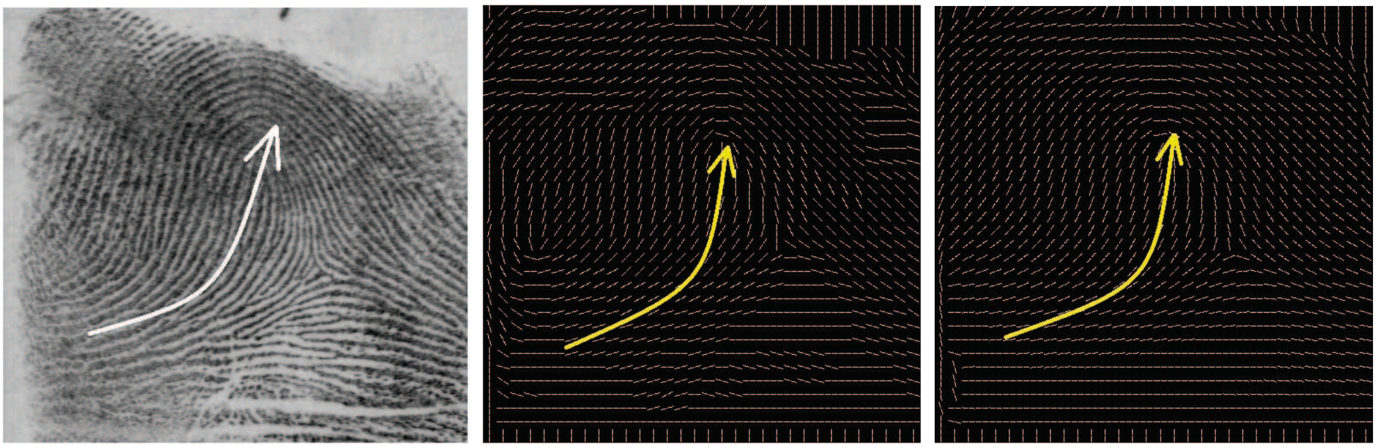
III. MULTI-CLASSIFIER

Fingerprint classification is a multi-class classification problem, the feature obtained by unsupervised learning and their labels (i.e. the type of fingerprints) are selected as the new input data to train a multi-classifier by supervised method. In this paper, the softmax regression model [14] is employed.

Softmax regression model can be regarded as a generalized linear model which is generated from a multinomial distribution. Consider a classification problem in which the class label y can take on any one of k values, so $y \in \{1, ..., k\}$. Given a test input x, we estimate the probability $p(y = j|x)$ that for each class j. Therefore, the model can be express as following:

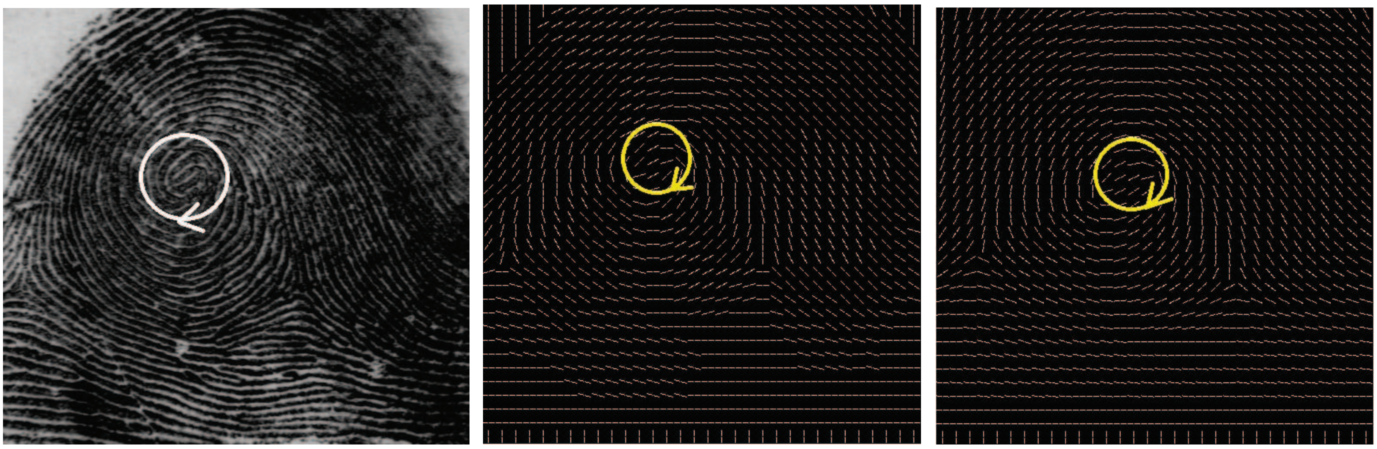
$$h_\theta(x) = \begin{bmatrix} p(y=1|x;\theta) \\ p(y=2|x;\theta) \\ \vdots \\ p(y=k|x;\theta) \end{bmatrix} = \frac{1}{\sum_{j=1}^{k} e^{\theta_k^T x}} \begin{bmatrix} e^{\theta_1^T x} \\ e^{\theta_2^T x} \\ \vdots \\ e^{\theta_k^T x} \end{bmatrix}. \quad (6)$$

Where $\theta_i \in R^{n+1}$ are the parameters of our model. Now, given a training set $\{(x^{(1)}, y^{(1)}), (x^{(2)}, y^{(2)}), ..., (x^{(m)}, y^{(m)})\}$



(a) (b) (c)

Fig. 2. The study results: (a) left loop, (b) is the input orientation field, (c) is the reconstructed orientation field by SAE.



(a) (b) (c)

Fig. 3. The study results: (a) whorl, (b) is the input orientation field, (c) is the reconstructed orientation field by SAE.

of m labeled examples, the model parameters θ are trained to minimize the following cost function:

$$L(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{k} I\{y^{(i)} = j\} \log \frac{e^{\theta_j^T x^{(i)}}}{\sum_{l=1}^{k} e^{\theta_l^T x^{(i)}}} + \frac{\lambda}{2}\sum_{i=1}^{k}\sum_{j=0}^{n}\theta_{ij}^2. \tag{7}$$

Where $I(.)$ is the indicator function, so that $I(true) = 1$, and $I(false) = 0$. The second term is a regularization term, $\lambda > 0$.

Through the previous introduction, now a new test fingerprint can be classified by extracting feature and selecting the trained multi-classifier.

IV. Experimental Results

The proposed fingerprint classification algorithm was tested on the NIST-DB4 database. The target class is four classes: arch (A), left loop (L), right loop (R) and whorl (W).

A. *Database*

NIST-DB4 database is used for testing classification accuracy by most of algorithms, which consists of 4000 fingerprint images (image size is 512×512) and the fingerprints have been manually labeled (A/L/R/TA/W). In our algorithm, half of them are chosen as the training set which is used for training the parameters of the SAE and softmax regression model, the other half as the testing set. The two parts have no cross.

B. *Experimental design*

Firstly, the fingerprint is divided into 20×20 blocks. Then the orientation of each block is computed, which obtains 25×25 direction values. Finally, convert them to the vector form. The dimension of input data is 1250. The *sigmoid* function is adopted as the network's activation function in that we normalize the input vector to $[0, 1]$.

In our paper, we don't have any pretreatment work before extracting the orientation field. We think the deep learning is

Fig. 4. Two examples of poor quality fingerprint and ambiguous fingerprint.

robust to fingerprints rotation, translation and some distortion and we hope that the human intervention as little as possible.

We first test the SAE with a single hidden layer containing 600 (400, 200 as control group) nodes. The result of classification is 90.35% (90.2%,89.7%) and the more results with different number of nodes are shown in Figure.5. The confusion matrix is described in Table I, Table II and Table III.

TABLE I
THE RESULT USING ONE HIDDEN LAYER WITH 200 NODES

True class 89.7%	Assigned class			
	A	L	R	W
A	768	19	12	1
L	63	321	0	16
R	55	0	335	10
W	7	10	13	370

TABLE II
THE RESULT USING ONE HIDDEN LAYER WITH 400 NODES

True class 90.2%	Assigned class			
	A	L	R	W
A	751	28	18	3
L	52	338	0	10
R	42	1	345	12
W	5	13	12	370

As can be seen from Figure.5, the changes of accuracy get smaller with increasing the number of nodes when other parameters of model unchanged. Considering the error of classification since the numerical calculation, we set the first hidden layer with 400 nodes in the next experiments.

TABLE III
THE RESULT USING ONE HIDDEN LAYER WITH 600 NODES

True class 90.35%	Assigned class			
	A	L	R	W
A	767	21	11	1
L	52	336	1	11
R	57	0	335	7
W	6	9	16	369

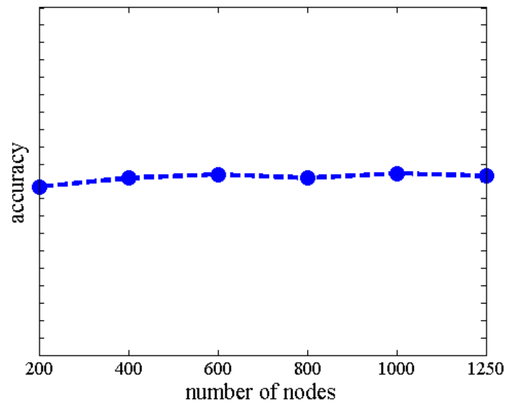


Fig. 5. Relation graph of number of nodes and accuracy in autoencoder with one hidden layer.

Then adding the number of hidden layers (two hidden layers with the nodes 400-100 in our system), the accuracy of classification is improved to 90.9%. The confusion matrix is shown in Table IV.

TABLE IV
THE RESULT USING TWO HIDDEN LAYERS WITH 400-100 NODES

True class 90.9%	Assigned class			
	A	L	R	W
A	767	17	16	0
L	49	336	2	13
R	54	1	338	7
W	7	8	8	377

At last, three hidden layers with the nodes 400-100-50 is employed and the accuracy of classification is improved to about 91.4%. Table V shows the confusion matrix. From these results, multi-layer structure (with fine tuning) has stronger learning power using the same data set.

TABLE V
THE RESULT USING THREE HIDDEN LAYERS WITH 400-100-50 NODES

True class 91.4%	Assigned class			
	A	L	R	W
A	770	18	11	1
L	49	342	0	9
R	47	0	345	8
W	5	10	16	369

C. Enhance classification accuracy

1) Reject option: In order to improve the accuracy of classification, the approach of establishing the "unknown" class is desirable, because there are many fingerprint images with poor quality in the database. These fingerprints often lead to the classification error. So putting these fingerprint images with poor quality into the unknown class is a feasible solution. Obviously, with the increasing numbers of fingerprints divided into unknown class, the accuracy of algorithm will significantly increase. For example, in Table VI, the result of [18] is 93.1% when about 1.8% fingerprints are rejected.

TABLE VI
THE PERFORMANCE OF CLASSIFICATION ALGORITHMS

Method	Test set	4 classes	rejection rate
Candela et al. [15]	second half	88.6%	0%
Karu et al. [16]	whole DB	91.1%	0%
Jain et al. [17]	whole DB	91.2%	0%
Yao et al. [18]	second half	93.1%	1.8%
Zhang et al. [19]	whole DB	95.3%	11.8%
Liu et al. [20]	whole DB	92.1%	-
ours	second half	91.4%	0%

2) Fuzzy classification: In practice, however, people always want to improve the accuracy of classification as far as possible without rising more costs. Fuzzy classification is an effective way to improve the classification accuracy.

Fingerprints with poor quality or ridge structure itself having characteristics of two different types such as Figure.4, which could easily cause a wrong classification for an AFIS, even for human experts. For instance, NIST-DB4 contains 350 ambiguous fingerprint pairs (about 17%) [21] which are marked by experts.

Fortunately, our classification results are outputted in probability form. Namely, for each fingerprint, its output includes four probability value corresponding to four categories. For simplicity, the four probability value $p1, p2, p3, p4$ are output in descending order. It is likely that $p1$ does not point to the correct category, especially for the samples which have a small $p1$ value. So in our fuzzy method, given a constant threshold, if the maximum probability value of fingerprint is smaller than the threshold, the fingerprint would be assigned to its second class at the same time. Take the single hidden layer network with 600 nodes (90.35%) for example, the result is given in Table VII.

Note: through the classification algorithm, every fingerprint has a class label, but in our fuzzy method, there are some

TABLE VII
THE RESULT OF FUZZY CLASSIFICATION UNDER DIFFERENT THRESHOLD VALUE

threshold δ	num1	num2	acc
0.60	145	2145	93.1%
0.70	277	2277	94.3%
0.75	347	2347	94.9%
0.80	464	2464	95.5%
0.85	616	2616	96.1%
0.90	806	2806	97.2%
0.95	1801	3081	98.0%
1.00	2000	4000	99.0%

fingerprints have two class labels. In Table VII, given the threshold δ, num1 indicates the number of fingerprints satisfying $p1 < \delta$; num2 is the total number of labeled fingerprints (the fingerprints with two labels are calculated two times); acc is the accuracy of the classification under the given threshold δ.

V. CONCLUSION

In this paper, we try to use a novel model, the depth neural network, for traditional fingerprint classification problem. By the unsupervised feature self-taught learning, a good low-dimensional representation of the input data is obtained. In order to further improve the accuracy of classification, the softmax regression is adopted for fuzzy classification.

The fuzzy method is very useful in practice. In real life, people always make all possible judgments about uncertain events. Based on this consideration, a secondary class is provided for each "suspicious" fingerprints. The experiments show that our algorithm can get about 99% accuracy when we consider the secondary class for each fingerprint.

Although a better result can be acquired by applying fuzzy method, some samples still can not be classified in to the right type. There are two main reasons. The first one is that the single feature has weak robustness for fingerprints with poor quality. The second one is that the ridge structure of some samples with a strong similarity to another type, the classifier can not recognize the right type or give it a high score.

In the future, multiple features and classifiers, like [22], can be considered for classification task, and in view of the learning ability of deep learning, the classification approach based on original images is also our next research focus, like [23].

ACKNOWLEDGMENT

This work was funded by the Chinese National Natural Science Foundation (11331012, 71271204, 11571014).

REFERENCES

[1] Henry E R. Classification and uses of finger prints[M]. HM Stationery Office, 1905.
[2] Galar M, Derrac J, Peralta D, et al. A survey of fingerprint classification Part I: Taxonomies on feature extraction methods and learning models[J]. Knowledge-based systems, 2015, 81: 76-97.
[3] Hinton G E, Salakhutdinov R R. Reducing the dimensionality of data with neural networks[J]. Science, 2006, 313(5786): 504-507.

[4] Vincent P, Larochelle H, Bengio Y, et al. Extracting and composing robust features with denoising autoencoders[C]//Proceedings of the 25th international conference on Machine learning. ACM, 2008: 1096-1103.
[5] LeCun Y, Bengio Y, Hinton G. Deep learning[J]. Nature, 2015, 521(7553): 436-444.
[6] Zhang Z, Luo P, Loy C C, et al. Learning deep representation for face alignment with auxiliary attributes[J]. IEEE transactions on pattern analysis and machine intelligence, 2016, 38(5): 918-930.
[7] Yao K, Zweig G, Hwang M Y, et al. Recurrent neural networks for language understanding[C]//INTERSPEECH. 2013: 2524-2528.
[8] Silver D, Huang A, Maddison C J, et al. Mastering the game of Go with deep neural networks and tree search[J]. Nature, 2016, 529(7587): 484-489.
[9] Sankaran A, Pandey P, Vatsa M, et al. On latent fingerprint minutiae extraction using stacked denoising sparse autoencoders[C]//Biometrics (IJCB), 2014 IEEE International Joint Conference on. IEEE, 2014: 1-7.
[10] Cao K, Jain A K. Latent orientation field estimation via convolutional neural network[C]//2015 International Conference on Biometrics (ICB). IEEE, 2015: 349-356.
[11] Ng A. Sparse autoencoder[J]. CS294A Lecture notes, 2011: 72.
[12] Hong J H, Min J K, Cho U K, et al. Fingerprint classification using one-vs-all support vector machines dynamically ordered with Bayes classifiers[J]. Pattern Recognition, 2008, 41(2): 662-671.
[13] Rao A R. A taxonomy for texture description and identification[M]. Springer Publishing Company, Incorporated, 2012.
[14] Gold S, Rangarajan A. Softmax to softassign: Neural network algorithms for combinatorial optimization[J]. Journal of Artificial Neural Networks, 1996, 2(4): 381-399.
[15] Candela G T, Grother P J, Watson C I, et al. PCASYS-A pattern-level classification automation system for fingerprints[J]. NIST technical report NISTIR, 1995, 5647.
[16] Karu K, Jain A K. Fingerprint classification[J]. Pattern recognition, 1996, 29(3): 389-404.
[17] Jain A K, Minut S. Hierarchical kernel fitting for fingerprint classification and alignment[C]//Pattern Recognition, 2002. Proceedings. 16th International Conference on. IEEE, 2002, 2: 469-473.
[18] Yao Y, Frasconi P, Pontil M. Fingerprint classification with combinations of support vector machines[C]//Audio-and Video-Based Biometric Person Authentication. Springer Berlin Heidelberg, 2001: 253-258.
[19] Zhang Q, Yan H. Fingerprint classification based on extraction and analysis of singularities and pseudo ridges[J]. Pattern Recognition, 2004, 37(11): 2233-2243.
[20] Liu M. Fingerprint Classification Based on Singularities[C]//Pattern Recognition, 2009. CCPR 2009. Chinese Conference on. IEEE, 2009: 1-5.
[21] Watson C I, Wilson C L. NIST special database 4[J]. Fingerprint Database, National Institute of Standards and Technology, 1992, 17: 77.
[22] Cao K, Pang L, Liang J, et al. Fingerprint classification by a hierarchical classifier[J]. Pattern Recognition, 2013, 46(12): 3186-3197.
[23] Sun Y, Wang X, Tang X. Deep learning face representation from predicting 10,000 classes[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014: 1891-1898.