FINGERPRINT RECOGNITION USING NEURAL NETWORK

W. F. Leung, S. H. Leung, W. H. Lau and Andrew Luk

Department of Electronic Engineering
City Polytechnic of Hong Kong
83 Tat Chee Avenue
Hong Kong

ABSTRACT

This paper describes a neural network based approach for automated fingerprint recognition. Minutiae are extracted from the fingerprint image via a multilayer perceptron (MLP) classifier with one hidden layer. The backpropagation learning technique is used for its training. Selected features are represented in a special way such that they are simultaneously invariant under shift, rotation and scaling. Simulation results are obtained with good detection ratio and low failure rate. The proposed method is found to be reliable for system with a small set of fingerprint data.

1. INTRODUCTION

Fingerprints are imprints formed by friction ridges of the skin in fingers and thumbs. They have long been used for identification because of their immutability and individuality. Immutability refers to the permanent and unchanging character of the pattern on each finger, from before birth until decomposition after death. Individuality refers to the uniqueness of ridge details across individuals, the probability that two fingerprints are alike is about 1 in 1.9×10^{15}.

The use of computers in fingerprint recognition is highly desirable in many applications, such as building or area security and police work to identify criminals. Recently, automated fingerprint classification techniques have been investigated [1][2]. The most prevalent current model for automated fingerprint identification systems is the Minutiae-Coordinate model (also called the FBI representation of prints). Most commercial systems are based on the basic FBI model. Classification is usually performed by locating the positions of certain distinctive features, known as minutiae, which are the beginnings and endings of ridges or forks (bifurcations) in the ridge lines of the print. Current fingerprint recognition systems are sensitive to errors in the positioning of prints at acquisition, and hence require to position the prints correctly for coordinates calculation or reference line placement. Our proposed method is based on a data model for fingerprints that is structural rather than coordinate. This structural data model is robust with respect to translation, rotation, and distortions.

0-7803-0118-8/91/0009-0226$01.00

Most of the pattern recognition systems are composed of four building blocks (see Figure 1). The first step is image acquisition, i.e. converting a scene into an array of numbers that can be manipulated by the computer. The second part is preprocessing, which involves removing noise, enhancing the picture, and, if necessary, segmenting the image into meaningful regions to be analyzed separately. The third phase is feature extraction, whereby the image is represented by a set of numerical "features" to remove redundancy from the data and reduce its dimension. The fourth building block is classification. This is the last stage of an image recognition system, where a class label is assigned to the unknown image/object by examining its extracted features and comparing them with class representations that the classifier has learned during its training stage.

The main focus of this paper is on the feature extraction and classification stages. Neural networks enable solutions to be found to problems where algorithmic methods are too computationally intensive or do not exist. They also offer significant speed advantages over conventional techniques. The problems of feature extraction and classification therefore seem to be a suitable application for neural nets. In this paper we have concentrated on the multilayer perceptron network using backpropagation learning technique [3]. This combination is the simplest and most widely used neural network. In the next section, we will discuss the configuration of the preprocessing system. Sections 3 and 4 describe in more detail the feature extraction and recognition procedures involved and the difficulties encountered. Some results and discussions will be presented in section 5 and the paper will be summarized with a brief conclusion section.

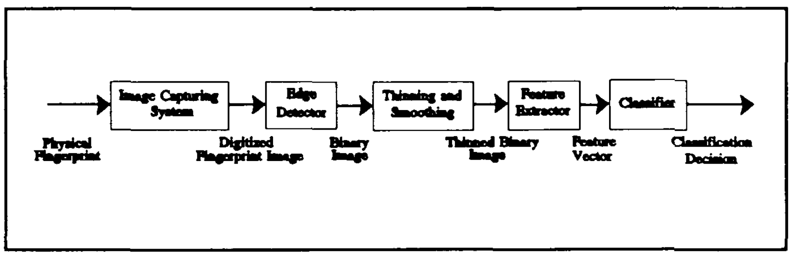

Fig. 1 Fingerprint Recognition System

2. PREPROCESSING SYSTEM

A 'clean' fingerprint image should be obtained to facilitate recognition, therefore methods of simple, standard image processing techniques [4] for edge enhancement and binarization or fingerprint filter design [5][6] are required. The first phase of our work is to capture the fingerprint image and convert it to a digital representation with a resolution of 512x512 by 256 gray levels. It is then subsampled and a valid region is selected based on visual inspection. Histogram equalization technique is sometimes used to increase the contrast if the illumination condition is poor. Although the original image is in gray-scale, we are only interested in binary information, i.e. the foreground fingerprint ridges and the background valleys. Therefore binarization is usually performed by using a Laplacian edge detection operator followed by a thresholding technique. The binary image is further enhanced by a thinning algorithm (modified from [7]) which reduces the image ridges to a skeletal structure with thickness of only one pixel and without changing the connectivity of the print. This algorithm is of the "banana peeling" type, where contour points are peeled off according to certain rules.

After obtaining the binary form of the fingerprint image, there may be some irregularities in the print. This is because of the imperfections of the fingerprint such as ridge gaps, which are usually caused by: (a) skinfolds and contiguous ridges, or (b) the spreading of ink due to finger pressure, or (c) in the worse cases by excessive inking or by smearing during rolling of the finger. Unfortunately, these irregularities cannot be avoided. To remedy this problem, smoothing of the binary image is necessary. The smoothing operations include: (a) filling holes, (b) deleting redundant points, (c) removing noisy points, and (d) filling potential missing points. For more details, see Ref. [8].

3. FEATURE EXTRACTION AND SELECTION

Extraction of appropriate features is one of the most important tasks for a recognition system. Because it is impractical to match a given input image or image representation with all the image templates stored in the system, it is necessary to find a compact set of features which can represent much of the useful information (in the sense of discriminability) present in the original data. Selection of "good" features is a crucial step in the process since the next stage sees only these features and acts upon them. "Good" features are those satisfying two requirements: (i) small intraclass invariance -- slightly different shapes with similar general characteristics should have numerically close values, and (ii) large interclass separation -- features from different classes should be quite different numerically. The next phase of our work involves the extraction of features from the thinned binary image. A multilayer perceptron network of three layers is trained to detect the minutiae in the thinned print image of size 128x128. The first layer of the network has nine units associated with the components of the input vector. The hidden layer has five units and the output layer has one unit corresponding to the number of the classes. The network is trained to output a '1' when the input window is centred on the feature to be located and it outputs a '0' if minutiae are not present. Figure 3 shows the initial training patterns which are composed of 16 samples of bifurcations in eight different orientations and 36 samples of non-bifurcations. The network is trained by using the backpropagation learning technique as described by Rumelhart et al [3] with the weight change is updated according to the equation,

$$\Delta w_{ij}(n) \quad = \quad -\eta \frac{\partial E}{\partial w_{ij}} + \alpha \Delta w_{ij}(n-1)$$

where E is the energy function which is defined as the sum of the square difference between the desired output response and the actual output response of each training example,

w_{ij} is the connection weight between unit i and unit j of the network,

$\Delta w_{ij}(n)$ is the weight change in the n^{th} cycle,

η is the learning rate,

and α is the momentum term.

In all experiments, a learning rate of 0.3 and a momentum of 0.9 are used. The training procedure is performed on a 80386-based PC run at 25MHz with a floating point coprocessor. It takes about 30 seconds for the network to achieve satisfactory convergence.

The trained network is then used to analyze the complete image by raster scanning the fingerprint via window of size 3x3. The network is proved to be very effective at identifying the positions of the minutiae and gives only few false responses. These false responses mainly come from the blurred areas in the original image where the signal level is low. In order to prevent the falsely reported features and select "significant" minutiae, two more rules are added to the system to guarantee perfect ridge forks are detected while excluding all other features. They are: (i) at those potential minutiae feature points, we re-examine them by increasing the window size to 5x5, and (ii) if two or more minutiae are too close together, we ignore all of them. Figure 2 shows the distribution of minutiae on two identical fingerprints (a) before and (b) after applying the rules. Note that on the left hand side of Fig. 2(a), there are a lot of feature points that are identified by the network but they are too closely linked together to be of significance in classification. When the above two rules are applied to the extracted features, these points are correctly removed. The network is then further tested on a number of fingerprint images and the important minutiae are also correctly selected.

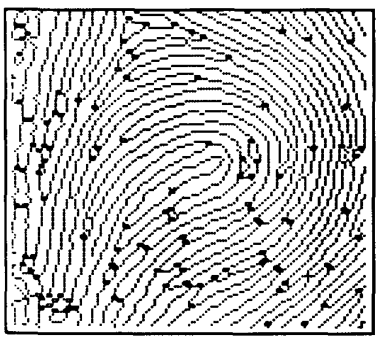

Fig. 2(a)

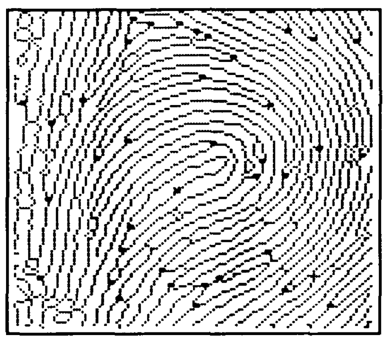

Fig. 2(b)

4. INVARIANT RECOGNITION

An important issue in applying neural networks to image recognition is the representation of feature data as input to the network. Additionally, a flexible recognition system must be able to recognize an object regardless of its orientation, size, and location in the field of view. This requirement is similar to the translation-, rotation-, and scale-invariancy properties for the extracted features. The location of a reference point (or a centre point) of the fingerprint is important for invariant recognition and has to be determined. To accomplish this, an approach of contour tracing [4] is used to find one or more turning points (i.e. points with maximum rate of change of tracing movement). These points are then used to find the reference point of the fingerprint.

The Euclidean distances d(i) from each feature point i to the centre point are calculated. The referencing of the distance data to the centre point confers the property of positional invariance. The data is now sorted in ascending order from d(0) to d(N) and this operation gives the data the property of rotational invariance. In order to make the data becomes invariant to scale change, it is normalized to unity by the shortest distance d(0), i.e. $d_{norm}(i) = d(0)/d(i)$, $i = 0..N$. This normalization will weight those feature points nearer to the centre more heavily because these points are usually more significant in classification. On the other hand, feature points at the borders are usually of poor quality or more noisy (due to excessive inking or smearing when obtaining the physical fingerprint) and, therefore, should be weighted less. Consequently the centroidal data patterns output in the form of amplitude spectrum should be shift, scale and rotation independent. Also the invariant feature vectors are in the range [0,1], they can be directly used as the training/stored vectors in the MLP classifier.

5. RESULTS AND DISCUSSIONS

Of the 30 fingerprints that we have collected so far, 10 were used to train the network. Using the method we proposed in sections 3 and 4, all the fingerprints can be recognized correctly. The invariant amplitude spectra of the first four fingerprints (shown in Figs. 5 and 6) are shown in Figure 4. It can be seen that the discrimination of prints is easy. Figures 5(a)-(f) show six digitized fingerprint images and Figures 6(a)-(f) show their corresponding thinned binary images with "significant" minutiae are marked by 'o' and the centre point is marked by an 'X'. Notice that Figures 5(a) and (b) are the same fingerprints with different positions while Figures 5(c) and (d) are of different types of fingerprints.

The recognition rate of fingerprints depends much on the quality of fingerprints and effectiveness of the preprocessing system, such as the threshold level used in edge detection which is a subjective issue. Also if there are too many broken lines or noisy points in the image, the preprocessing system of contour tracing may fail. Thus an intelligent connection algorithm to recover broken lines and suppress spurious irregularities is necessary.

6. CONCLUSIONS

We have presented a fingerprint recognition system that uses local features to identify fingerprints. Preprocessing techniques are first applied to produce a clean, thinned binary fingerprint ridge structure which is ready for feature extraction. Backpropagation-trained neural network are successfully used to extract the minutiae in the print. Good features are then selected and represented in a simple data model which allows correct recognition of prints even in the presence of positioning errors. The use of neural network supports fast classification rate and saves the user from spending enormous amount of time to derive rule-based databases for matching.

7. ACKNOWLEDGMENT

This research work was performed with support from the City Polytechnic of Hong Kong Strategic Indicated Grant #700-073.

8. REFERENCES

[1] A. K. Hrechak and J. A. McHugh, "Automated Fingerprint Recognition Using Structural Matching," Pattern Recognition, vol. 23, no. 8, pp. 893-904, 1990.

[2] K. Rao and K. Black, "Type Classification Of Fingerprints: A Syntactic Approach," IEEE Trans. Pattern Anal. Mach. Intell., PAMI-2, pp. 223-231, 1980.

[3] D. E. Rumelhart, G. E. Hinton and R. J. Williams, "Learning Internal Representations by Error Propagation," chapter 8 of vol. 1 in Parallel Distributed Processing, MIT Press, Cambridge, Mass., USA, 1986.

[4] A. Luk, S. H. Leung, C. K. Lee and W. H. Lau, "A Two-level Classifier For Fingerprint Recognition," in Proc. IEEE 1991 International Symposium on CAS, Singapore, June 11-14, 1991, pp. 2625-2628.

[5] L. O'Gorman and J. V. Nickerson, "An Approach To Fingerprint Filter Design," Pattern Recognition, vol. 22, no. 1, pp. 29-38, 1989.

[6] M. T. Leung, W. E. Engeler and P. Frank, "Finger Image Processing Using Neural Network," in Proc. of IEEE Region 10 Conference on Comp. and Commun. Sys., Hong Kong, 1990, pp. 582-586.

[7] G. Feigin and N. Ben-Yosef, "Line Thinning Algorithm," in Proc. of SPIE : Applications of Digital Image Processing, vol. 397, Switzerland, 19-22 April 1983, pp. 108-112.

[8] A. Luk, S. H. Leung, W. K. Suen and C. Y. Cheng, "A Low Cost Fingerprint Recognition Subsystem For Security Systems," in Proc. of International AMSE Conf. on Signals and Systems, Brighton, UK, AMSE Press, vol. 1, 1989, pp. 221-232.

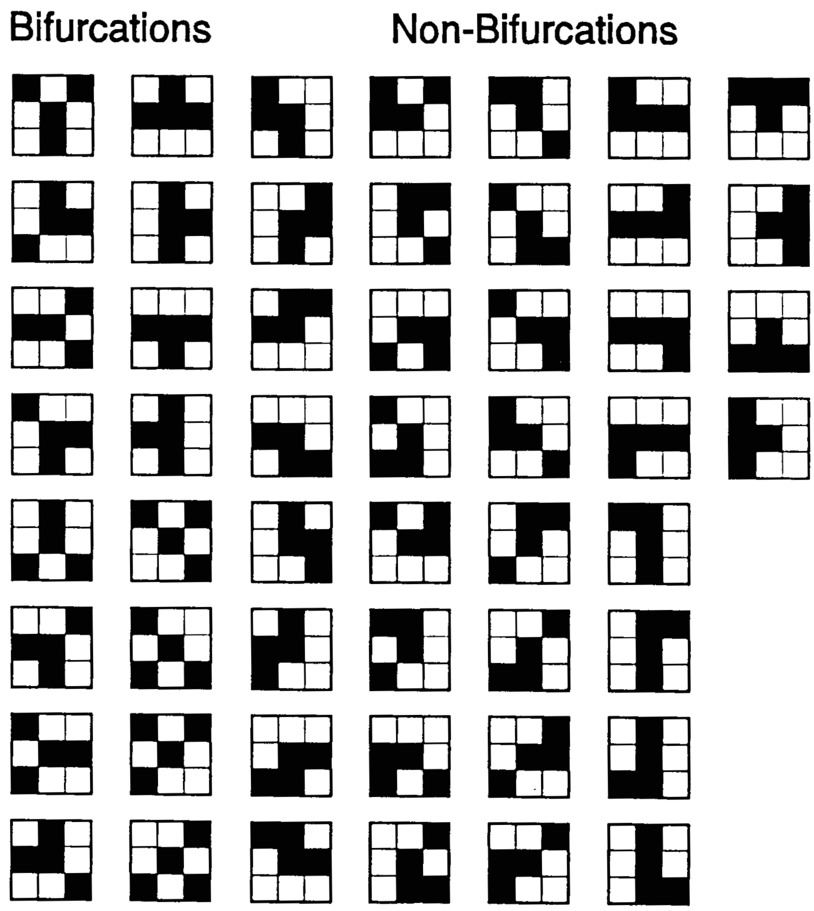

Fig. 3 16 bifurcated and 32 non-bifurcated training samples

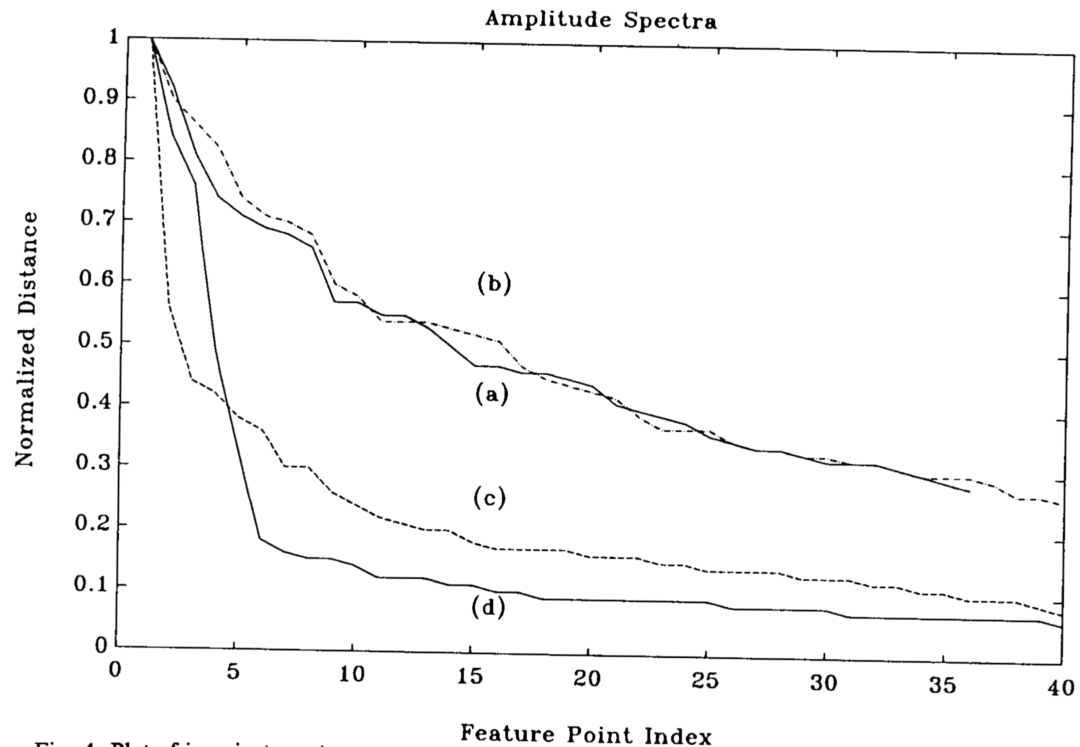

Fig. 4 Plot of invariant spectra

Fig. 5(a)

Fig. 5(b)

Fig. 5(c)

Fig. 5 Original gray-level fingerprints

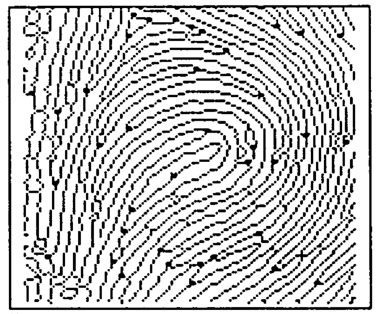

Fig. 6(a)

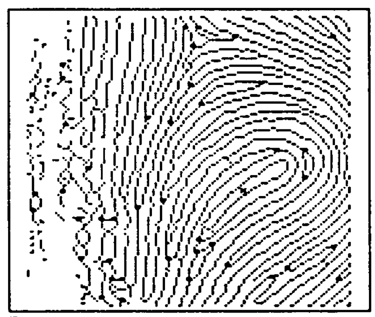

Fig. 6(b)

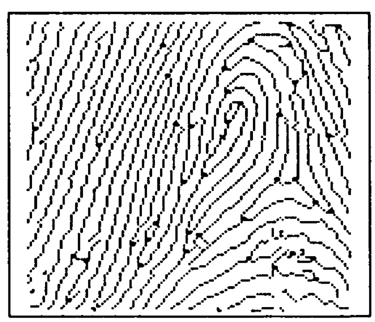

Fig. 6(c)

Fig. 6 Feature points extracted from thinned prints

Fig. 5(d)

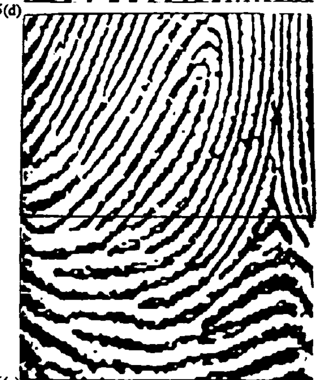

Fig. 5(e)

Fig. 5(f)

Fig. 5 Original gray-level fingerprints

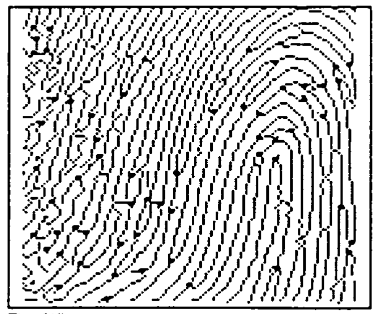

Fig. 6(d)

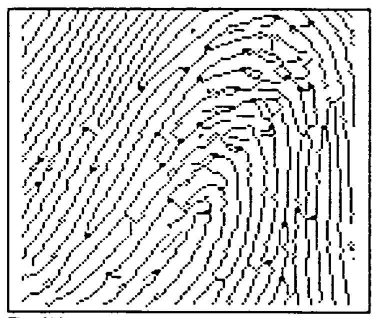

Fig. 6(e)

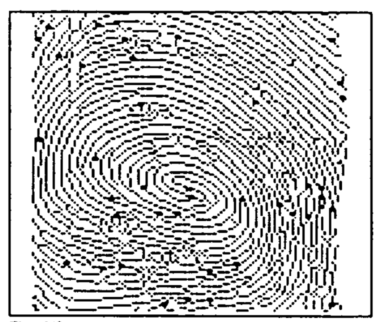

Fig. 6(f)

Fig. 6 Feature points extracted from thinned prints