# A Fingerprint Recognition System Using Cellular Neural Networks

**Te-Jen Su[1*], Yan-Yi Du[1], Ying-Jen Cheng[1], and Yi-Hui Su[2]**

1.Center for Electronic Communication Technology, National Kaohsiung University of Applied Sciences, Kaohsiung, Taiwan 807, R.O.C.

2.Department of Information Management,Tzu-Hui Institute of Technology, Pintung, Taiwan 926, R.O.C.

*sutj@cc.kuas.edu.tw

**Abstract:**

In this paper we introduce a fingerprint recognition system based on Cellular Neural Networks (CNN) algorithms. This system is divided into three parts. First, the fingerprint image is enhanced then ridgeline is thinned. Finally matching is employed. Fingerprint thinning is proposed by employing a new approach using CNN. Simulation result shows the effectiveness of the proposed method.

**Keywords:** Fingerprint, Recognition, Cellular Neural Networks, Matching.

## 1. Introduction

At present, identifying is a hot topic in modern society and many methods are applied in personal identification. However, fingerprint verification is a popular and standard biometric identification. Fingerprint has immutability and individuality feature. It is used widely in security products such as entrance control, ATM, operating system, etc. Therefore, we put forward CNN develop fingerprint recognition.

The structure of Cellular Neural Networks (CNN) is presented by Chua [1]. CNN Universal Machine is a low cost and low power supercomputer and it is at least 1000 times faster than equivalent DSP solutions about complex image processing tasks [2] Hence, it's very useful on image processing. Recently, the published papers which use CNN to deal with fingerprint almost pay attention to fingerprint enhancement and pattern thinning [3] [4] [6]. However, the complete structure of fingerprint recognition is less provided. From above reason, we build the complete process about fingerprint recognition. The processes of fingerprint recognition, fingerprint enhancement, ridgeline thinning, matching, and recognition, are described clearly in section 2. It is noteworthy that ridge thinning and feature extraction are proposed by employing a new approach using CNN. Simulation results are presented in section 3. Finally, conclusions are given in section 4. Particularly, the thinning algorithm is better than Qun Gao's method. [10]

## 2. Description of theory and design

### 2.1 The flowchart of system

The system described in this paper is depicted in Fig.1. Fingerprint enhancement and ridgeline thinning are employed before matching. Then feature extraction, recording the bifurcation, and transformation are completed. Finally the results are compared with database of fingerprint for identifying.

This work was partially supported by 2004 project of Ministry of Education,R.O.C

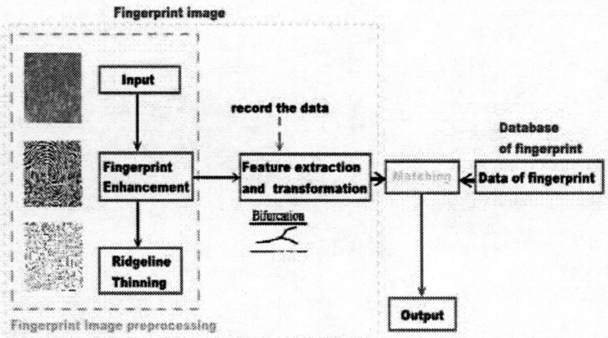


**Fig.1 The fingerprint recognition system**

### 2.2 Fingerprint enhancement

In the section about clearing and strengthening, due to the theorems are too many and very excellent. So, we refer to the paper "FINGERPRINT IMAGE ENHANCEMENT USING CNN GABOR-TYPE FILTERS" [5] published by Saatci, E. and Tavsanoglu, V., and apply the method to the picture of fingerprint.

GABOR-TYPE FILTERS is described that fingerprint which is like sinusoidal wave is presented in form of level and smooth profile, and every point which is on the profile is fit the relative varied angle. After importing the image, the angle of every point is compared with the angle of relative else point. If the value is different and the smooth result is presented by the other points, it will be enhanced, and then the clear picture is presented.

### 2.3 Ridgeline thinning

Because it is more difficult to search the bifurcation directly from fingerprint by Cellular Neural Networks, ridgeline must be thinned before extracting the feature point.

Ridgeline thinning, exactly as it's meaning, let the object on the image which could be presented by its skeleton, and the width is just one pixel. At this time, we refer to the parallel image thinning [7] and present the new method called CNN_Thinning by Cellular Neural Network to complete the work.

#### 2.3.1 Parallel image thinning

Border point is what we want to delete through the thinning processing. According to the definition: there are four kinds of border points which are east, west, south, and north four border points, respectively. To judge these pixels which are border points on the image are first important factor of thinning. So, the detailed definition about the border point is expressed below. As Fig.2 shows that let P0 is the center. If P4 is white, we call P0 as east border point and other points such as P8, P6, and P2 are to reason by analogy. Next, the processing of thinning is shown in Fig.3. At first, the image is divided into many blocks and the size of a block is 3 by 3. One direction border point has three kinds of templates. If the block is matched with any template, the pixel will be deleted. It must be noted that the ridgeline might disappear when four border points are considered at the same time. So, we just scan one direction every time. It will be one circle after scanning four directions, until the optimization thinning image is presented.

| | | |
|---|---|---|
| $P_1$ | $P_2$ | $P_3$ |
| $P_8$ | $P_0$ | $P_4$ |
| $P_7$ | $P_6$ | $P_5$ |

**Fig.2 Border point**

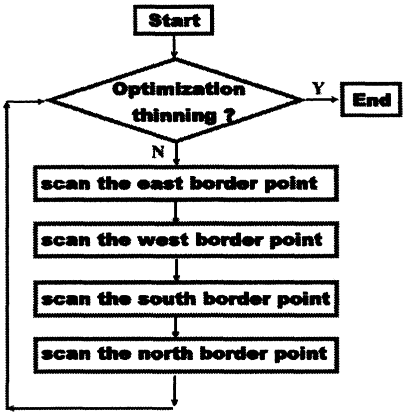


**Fig.3 Block diagram of thinning**

**2.3.2 Ridgeline thinning by CNN**

According to the above description, the thinning image will be presented if we constantly find the pixels, which are the border points, and delete them. It is more suitable to choose the CNN innovation theory to implement. By the way, one template of every border point is a complete method of CNN.

Before To perform the process of ridgeline thinning, every template must be named respectively and given some parameters. For example, the first templates of east border point. It is called CNNE1 and others are to the reason by analogy. All templates of CNN parameter (A, B, z) are presented as following depiction.

CNN_E1

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & -1 \\ 0 & 1 & 0 \end{bmatrix} \quad z = -4$$

CNN_E2

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & -1 \\ 0 & -1 & -1 \end{bmatrix} \quad z = -5$$

CNN_E3

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & -1 & -1 \\ 1 & 1 & -1 \\ 0 & 1 & 0 \end{bmatrix} \quad z = -5$$

CNN_W1

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad z = -4$$

CNN_W2

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 1 & 1 \\ -1 & -1 & 0 \end{bmatrix} \quad z = -5$$

CNN_W3

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} -1 & -1 & 0 \\ -1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad z = -5$$

CNN_S1

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & -1 & 0 \end{bmatrix} \quad z = -4$$

CNN_S2

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & -1 \\ 0 & -1 & -1 \end{bmatrix} \quad z = -5$$

CNN_S3

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & 1 & 0 \\ -1 & 1 & 1 \\ -1 & -1 & 0 \end{bmatrix} \quad z = -5$$

CNN_N1

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad z = -4$$

CNN_N2

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 & -1 & -1 \\ 1 & 1 & -1 \\ 0 & 1 & 0 \end{bmatrix} \quad z = -5$$

CNN_N3

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} -1 & -1 & 0 \\ -1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad z = -5$$

Through all the direction templates, the four directions border points are received. Finally, the thinning image is acquired via the image processing of CNN such as LOGOR and LOGDIF below.

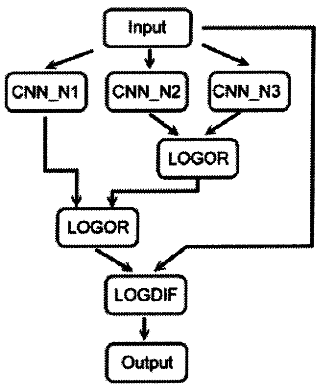


**Fig.4 Block diagram of border point searching**

### 2.4 Matching and Recognition

In this section, it is divided into two parts. The one is feature extraction and another is contrasting method. The detailed methods are introduced in next paragraph.

#### 2.4.1 Feature extraction

The feature point which is the cross point shown in Fig.5 is the important basis in fingerprint recognition. The idea about feature extraction is published by A. Wahab, S.H.Chin, and E.C.Tan [8]. It could be realize by CNN.


**Fig.5 Bifurcation**

##### 2.4.1.1 Feature Extraction using CNN

As Fig.2 depicted, P0 is the pixel that we want to analyze. The method of feature extraction could be described by the function.

$$\sum_{k=1}^{8} |P(k+1) - P(k)| = 12$$

For example: Besides the values of P3, P5, and P8 are 1, the others are -1. We take the amount to put in the function. If the answer is 12, this point is a cross point. In other words, if this point is cross point, its amount is necessary 12. In view of this character, we apply CNN theory to complete it. The parameters are given as follow.

CNN_Feature_Extract

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} b_1 & b_2 & b_3 \\ b_8 & U_{ij} & b_4 \\ b_7 & b_6 & b_5 \end{bmatrix} \quad z = -12$$

$$b_i = |U_k - U_i| \quad where\ k = (i+1)\ mod\ 8 \quad for\ i = 1 \sim 8$$

##### 2.4.1.2 State Dynamic Route

State Dynamic Route [1][9] is applied to analyze the path of varied state when the parameter is given. CNNFeatureExtract is presented below:

$$\dot{x}_{ij} = -x_{ij} + A \otimes Y_{ij} + B \otimes |U_{ij}| + z$$

$$\dot{x}_{ij} = \underbrace{-x_{ij} + a_{00} f(x_{ij})}_{g(x_{ij})} + \underbrace{\bar{A} \otimes Y_{ij} + B \otimes |U_{ij}| + z}_{w_{ij}(t)}$$

$$\dot{x}_{ij} = g(x_{ij}) + w_{ij}(x_{ij}, t)$$

Via this function, State Dynamic Route is shown as Fig.6. Only the pixels, which fit the feature points, are black and the others are become to white.

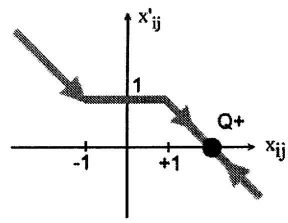


**Fig.6 (a) Dynamic Route of CNN_Feature_Extract**

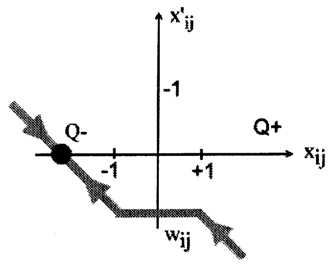


**Fig.6 (b) Dynamic Route of CNN_Feature_Extract**

#### 2.4.2 Matching

Fingerprint contrasting is the key process of all structure about identification. At the moment, there are many methods about contrasting, like field line matching, feature classification, and checking the distance of center point to delta, etc.

The new contrasting method is presented in the paper. The idea is implemented according to the distribution of feature points and the process of contrasting is as Fig.7 depicted.

At first, the point at the center is the origin and then distance and the angle of the feature point are recorded. Then, the records are classified by the structure of disc as Fig.8. The Mean and standard deviation of angle and distance in sector are calculated and recorded. Finally, the process of matching is performed.

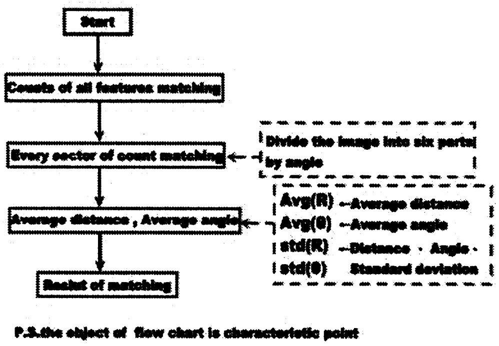


**Fig.7 Block diagram of matching**

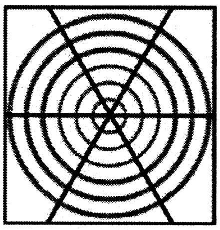

**Fig.8 Feature points are classified by structure of the disc**

### 3. Achievement of research

**Fig.9 Fingerprint enhancement**

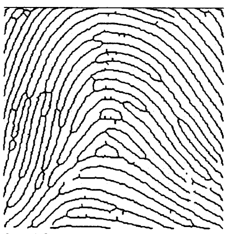

**Fig.10 ridgeline thinning**

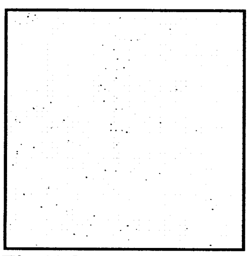

**Fig.11 feature extraction**

### 4. Conclusion

In above the demonstrations, fingerprint enhancement, ridgeline thinning, and feature extraction are employed successfully. Particularly, ridgeline thinning is an important image process and a fine thinning algorithm could improve the accuracy of matching. In other words, if the thinning pattern is not ideal, features will not be extracted accurately. In this paper, we make a comparison showed at Fig12 between our thinning algorithm and Qun Gao's. It is clear that the method by CNN is better than [10] either continuity of line or degree of thinning. Moreover, the algorithms of ridgeline thinning and feature extraction also could be applied in digital signature. Simulation result shows the effectiveness of the proposed method.

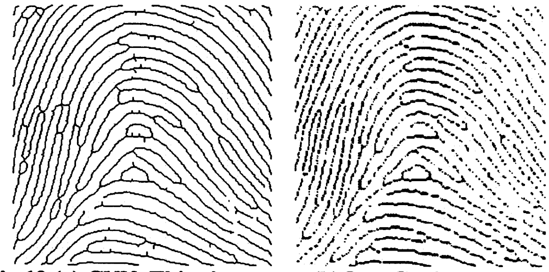

**Fig.12 (a) CNN_Thinning (b)Qun Gao's method**

### 5. References


[1] L. O. Chua and Lin Yang. "Cellular Neural Networks: Theory," IEEE Transactions on Circuits and Systems-I, vol. 35, pp. 1257-1272, Oct. 1988.

[2] http://burks.brighton.ac.uk/burks/foldoc/33/18.htm.

[3] T. Kozek, L.O. Chua, T. Roska, D. Wolf, R. Tetzlaff, F. Puffer, and K. Lotz, "Simulating nonlinear waves and partial differential equations via CNN - Part II. Typical Examples," IEEE Trans. on Circuits and Systems I: Fundamental Theory and Applications, vol. 42, pp. 816-820, 1995.

[4] K.R. Crounse and L. O. Chua, "Methods for Image Processing and Pattern Formmation in Cellular Neural Networks: A Turorial," IEEE Trans. on Circuits and Systems I: Fundamental Theory and Applications, vol. 42, pp. 583-601, 1995.

[5] E. Saatci, and V. Tavsanoglu, "Fingerprint Image Enhancement using CNN Gabor-Type Filters," Proc. Seventh IEEE International Workshop on Cellular Neural Networks and their Applications, pp.22-24, July 2002.

[6] B. E. Shi, "Gabor-Type Filtering in Space and Time with Cellular Neural Networks," IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, vol. 45, pp. 121-132, Feb. 1998.

[7] Chern Sheng Lin, DSP-Image and Sound Processing, Chien-Hau, 1997.

[8] A. Wahab, S.H. Chin, and E.C. Tan, "Novel approach to automated fingerprint recognition," IEE Proc.-Vision on Image Signal Process, vol. 145, pp. 160-166, June 1998.

[9] L. O. Chua, and Tamas Roska, Cellular neural networks and visual computing Foundation and applications. Mbridge University Press, 2002.

[10] Quit Gao, P. Forstei, K.R. Mobus, arid G.S. Moschytc, "Fingerprint recognition using CNNs: fingerprint preprocessing," IEEE International Symposium on Circuits and Systems, vol. 2, pp. 433-436, May 2001.

[11] Qun Gao, and G.S Moschytz, "Fingerprint feature matching using CNNs," Proceedings of International Symposium on Circuits and Systems, pp. III-73-6, vol. 3, May 2004.