A Sweeping Fingerprint Verification System using the Template Matching Method

SUN-YEN TAN [1], WEN-TZENG HUANG [2], CHIN-HSING CHEN [3], YUAN-JEN CHANG [3]

[1] Department of Electronic Engineering
National Taipei University of Technology
No. 1, Sec. 3, Chung-hsiao E. Rd., Taipei,10608, Taiwan, R.O.C.
sytan@ntut.edu.tw

[2] Department of Computer Science and Information Engineering
Mingsin University of Science and Technology
No.1, Xinxing Rd., Xinfeng Hsinchu 30401, Taiwan, R.O.C.
wthuang@must.edu.tw

[3] Department of Management Information Systems
Central Taiwan University of Science and Technology
No.666, Buzih Road, Beitun District, Taichung City 40601, Taiwan, R.O.C.
{ronchang, chchen } @ctust.edu.tw

Abstract: - Electronic products have adopted the technology of the fingerprint recognition system to offer protection mechanism. The characteristics of Line-Sensor are smaller size, lower power consumption, and lower cost. Therefore, it is suitable for designing fingerprint sensor in embedded systems. Generally, a Line-Sensor builds up a completed picture through assembling the image segments received after one sweeping the fingerprint. However, sometimes, the sweeping rate is unstable and the pressing strength is varied. To avoid such problems influencing the final results, the Fast Normalized Cross Correlation (FastNCC) algorithm is employed here to process and correct the image data. Moreover, the Group Delay Spectrum (GDS) and Dynamic Programming (DP) are implemented to complete fingerprint comparisons. Two different fingerprint matching algorithms are to exploit and demonstrate the verification rate of the system. FastNCC results in 93.8% of the verification rate, while SSD obtains 92.5% of the verification rate.

Key-Words: - Line-Sensor, Normalized Cross Correlation, Template matching, GDS, DP matching, Threshold.

1 Introduction

There are no two fully identical fingerprints and one's fingerprints can never be counterfeited [4][5]. Therefore, the fingerprint recognition technology is used to substitute the function of password, thus to enhance the security required by the users.

Two types of fingerprint sensors are Line-Sensors and Block-Sensors [2]-[5]. The Line-Sensor has been adopted in this study, since it requires lower power, smaller size, and fewer costs than the Block-Sensor in the fingerprint system.

The characteristics of Line-Sensor are smaller size, lower power consumption, and lower cost. Therefore, it is suitable for designing fingerprint sensor in embedded systems. Our earlier paper discussed the sweeping fingerprint verification system [12].

Dovgan and Gams proposed an intelligent entry control using fingerprint verification system [13]. Meghdadi and Jalilzadeh proposed the techniques for artificial fingerprint scanners [14]. Su et al. worked on fingerprint matching methods [15]. Dabbah et al. presented a high efficiency combined accuracy fingerprint recognition algorithm [16].

The paper is organized as follows: Session 2 demonstrates the system framework. Session 3 addresses the relevant materials and methods.

Session 4 shows the experimental results. The conclusions are shown in Session 5.

2 The System Framework

There are three stages, called input, image reconstruction, and fingerprint matching stages, in this fingerprint system framework as illustrated in Fig. 1.

Input Stage

Through a USB interface with the fingerprint Line-Sensors, a fingerprint image can be retrieved and then delivered to the computer program and database for processing.

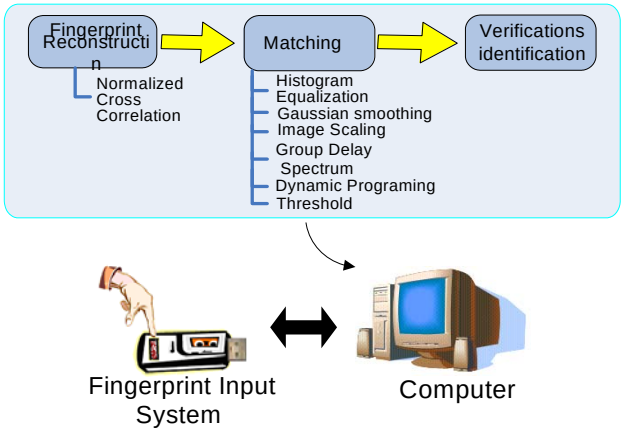

Fig. 1. Fingerprint system framework.

Image Reconstruction Stage

Because the Line-Sensor sensor area is designed by the linear structure, smaller than one's full fingers, it is necessary to re-organize the input fingerprint segments until the whole fingerprint image is completely constructed by the system [2] [3]. For reconstructing a completed fingerprint, the Fast Normalized Cross Correlation algorithm (FastNCC) [6] is applied to figure out the continuing relation between two neighbor fingerprint segments, and to further eliminate the embedded redundancy. The Normalized Cross Correlation algorithm (NCC) 0 used spatial analyses to evaluate the similarities among segments. That is, NCC exploits the distance between the captured segments to estimate the image similarities. It captures the pictures of test template and does a matching with the registered template line by line, until the most similarity appearance [7].

Fingerprint Matching Stage

Moreover, there are five sub-steps in this stage. First, the cylinder average method is used to enlarge the grey area. Then, the Gauss filter, a low-pass filter, is used to eliminate the impurities. Third, the image compression is completed through the Bresenham algorithm. The Group Delay Spectrum (GDS) [10] [11] is used to capture the characteristic values in which the spectrum analyses control the breadth and the concentric circles of the fingerprint in the fourth step. This GDS strengthens the individual characteristics of one's fingerprint and is hardly affected by the shade and break of images. The fifth step is to use the Dynamic Programming (DP) algorithm to match two fingerprints. DP is the best method to get the optimal path between the starting and ending points [1] [10] [11].

3 Materials and Methods

After getting the original fingerprint images, these can be processed by the following algorithms with its individual function step by step. Finally, after these processes, this system can verify the fingerprint owner. These processes, including the reconstruction algorithm, image enhancement, image compression, characteristic values capture, DP matching, and threshold setting, are the major components in our fingerprint system and are discussed as follows.

3.1 Reconstruction Algorithms

There are two functions here. The "fingerprint capturing" is used to access one's fingerprint and the "reconstruction fingerprint segments" is to merge the non-cumulating block of segments into one whole image.

Fingerprint Capturing

Calculation for the standard deviation (SD) helps to detect whether one's fingerprint has been swept or not as shown in Formulas 1 and 2. Let x be the data of one's fingerprint segment, x_i be the gray level of the i^{th} pixel, $\bar{x}$ be the average of the total gray levels, n be the size of the segment, and std be SD.

$$std = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n}} \qquad (1)$$

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n} \qquad (2)$$

SD of the non-sweeping and sweeping fingerprint images is evaluated separately. The experiment reveals that, when one did not sweep one's fingerprint, the SD value range is [5.6, 5.8]. In contrary, when this user is sweeping his/her fingerprint, the SD value range is [6.1, 12.5] as shown in Fig. 2. Thus, when SD is below 6, the system considers that none has swept; as SD is higher than 6, the system recognizes that someone is sweeping one's fingerprint at this time.

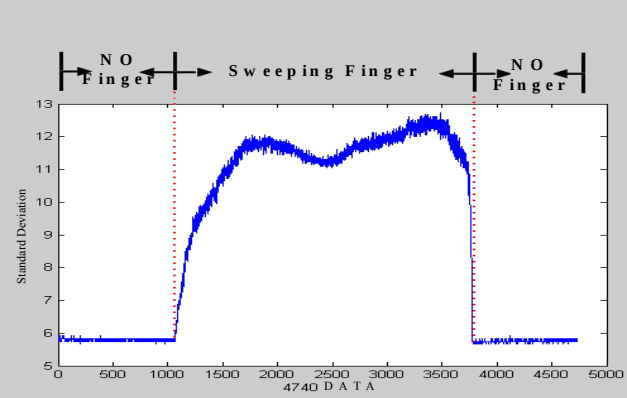

Fig. 1. Distribution of SD for non-sweeping and sweeping fingerprint images.

Reconstruction Fingerprint Segments

Before reconstructing the fingerprint segments, FastNCC is used to find the cumulating positions. Reading from the bottom of the image, two neighbor segments are retrieved and process the first row of segment 1 with every row of segment 2 by FastNCC to get the consistent positions as shown as Fig. 3. And then, the non-cumulating block of segment 2 is added to the top of segment 1, and the first iteration is completed. Next, the first row of this image is processed with every rows of the next segment by FastNCC again. The system will repeat this process until all segments are reconstructed. The whole process is shown in Fig. 4.

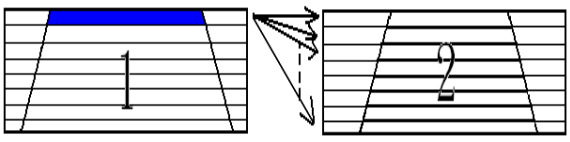

Fig. 2. Finding the cumulative positions.

3.2 Image Enhancement

After reconstructing the fingerprint image, the image enhancements are employed to enhance the image as follows. The image may be some spots on it, or the varied press strength will result in the gray area of the grain for the fingerprint image inconsistent. To reduce these problems, the clearness of the fingerprint image needs to be enhanced after reconstructing the image.

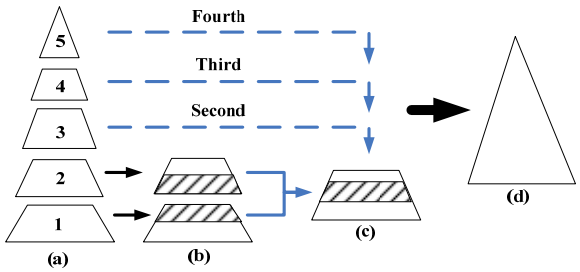

Fig. 3. Process of reconstruct image.

Histogram Equalization

The gray area of the sweeping fingerprint is always distributed to a centralized area. This histogram equalization will be used to decentralize its area.

We use histogram equalization to decentralize the area. Fig. 5 shows the distribution of gray area before histogram equalization.

The values of the gray area are centralized on a small range. After histogram equalization, the distribution is obviously decentralized and this result is helpful for next process as shown in Fig. 6 with the distribution of gray area.

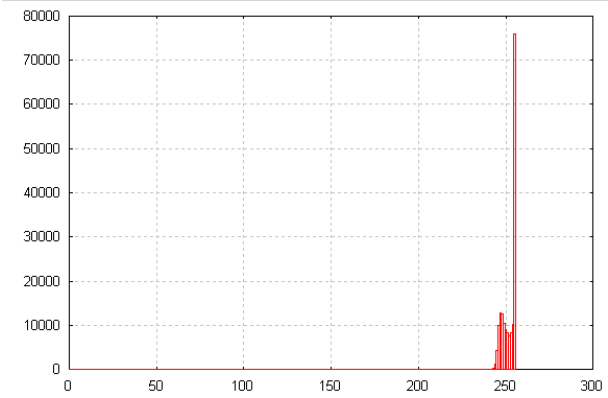

Fig. 5. (org) The distribution of gray area before histogram equalization.

Gaussian Smoothing

Gaussian Smoothing, also called low-pass filter, is used to eliminate the noise of images. The dimension formula is presented in Formula 3 and let σ be SD.

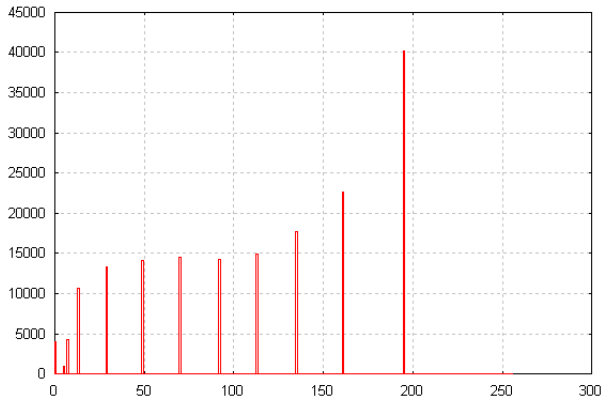

Fig. 6. Distribution of gray area after histogram equalization.

$$G(x) = \frac{1}{\sqrt{2\Pi\sigma}} e^{\frac{x^2}{2\sigma}} \qquad (3)$$

Fig. 7 illustrates the results when SD is separately substituted by 0.5, 1, and 1.5. Through the three curve lines, we know that the curve goes smoothly when σis bigger. Otherwise, the curve goes sharply. It also shows that an appropriate σ will result in a better filter effect.

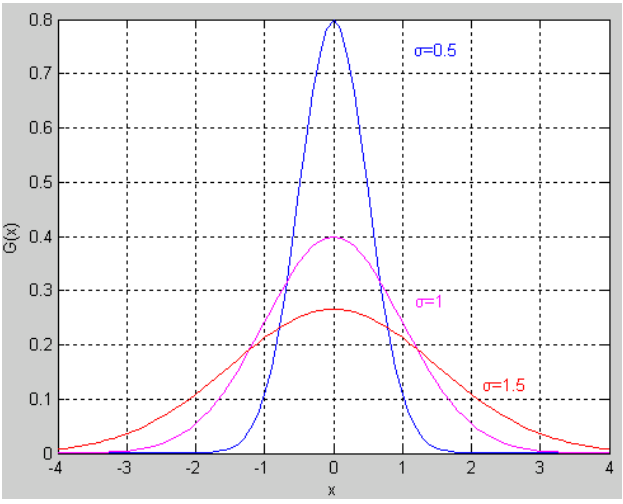

Fig .7. Gaussian Smoothing.

From this formula, we learn that the curve will be smooth, when σ is bigger. Otherwise, the curve goes sharply. Hence, it also shows that an appropriate σ will result in a better filter effect. Formulas 4 and 5 are two-dimension formulas used to process images. First, formula 4 is used to eliminate the noise of point (x, y).

$$G(x, y) = \frac{1}{\sqrt{2\Pi\sigma}} e^{\frac{x^2+y^2}{2\sigma}} \qquad (4)$$

$$P(x, y) = \sum_{j=y-1}^{y+1} \sum_{i=x-1}^{x+1} P(i, j) G(i - x, j - y) \qquad (5)$$

The new gray level can be received through Formula 5 in which P(i, j) means the pixel value of the point (i, j). The point (x, y) is adopted to be the central point, and, from four directions, up and down, right and left, the point is further extended one point outward. There are total 3x3 points to be collected for the calculation. Finally, each of the values is multiplied by the gray level and, then, added all together to be the new pixel value of the point (x, y).

3.3 Image Compression

A fast algorithm, Bresenham algorithm, is use to implement the image compression [8]. The Bresenham algorithm finishes the line-compression first and then row-compression. This algorithm takes the amount of adjustment for length and width according to user setting compression ratio to do pixels compression. And even the compression ratio is not an integer; it still should not result in the loss of image information.

3.4 Retrieving Characteristic value

The characteristic value is retrieved by the GDS method [10] [11], which is used to control the breadth and the concentric circle of the fingerprint, and has the effect of enhancing the spectrum peak value. Before using GDS, the Linear Predict Coding (LPC) should be introduced first. LPC is a mechanism of phonic analysis. Through this mathematical model, LCP transfers the phonic signals and reduces the information loading resulted from encoding. This mechanism is applied to image processing. When running LCP, the linear prediction is exploited to

predict the amplitude of the filter H(z) as shown in Formula 6.

$$H(z)=\frac{1}{1+\sum_{K=1}^{P}a_k z^{-k}} \qquad (6)$$

Let a_k be the coefficient of the linear prediction, p be the predicting level, and z be the transferring formula $z=e^{jwT}$, where T is the sampling cycle. GDS is defined as a frequency differentiation of the frequency spectrum function of H(z). The frequency spectrum function of H(z) and $\theta(\omega_i)$ is as Formula 7.

$$\theta(\omega_i)=-\tan^{-1}\left[\frac{\mathrm{Im}\{H'(z)\}}{\mathrm{Re}\{H'(z)\}}\right] \qquad (7)$$

GDS $T_g(\omega_i)$ is defined as formula 8. Let N be the point number taken for Fourier transfer, L(=N/2) be the divided number for frequency domain, and ω_i be the i^{th} frequency (i=1,…,L).

$$T_g(\omega_i)=-\frac{\Delta\theta(\omega_i)}{\Delta\omega} \qquad (8)$$

where $\Delta\theta(\omega_i)=\theta(\omega_i)-\theta(\omega_{i-1})$, $\Delta\omega=\frac{\pi}{L}=\frac{2\pi}{N}$.

GDS contains the characteristic in which the spectrum peak value may be too sharpened. Therefore, before doing a match, it is necessary to smooth the spectrum peak value. The way is to weight the coefficient a_k which is replaced by as follows.

$$a_k{}' = c^k a_k (c^k \le 1) \qquad (9)$$

3.5 DP Matching Mechanism

DP is the algorithm to get the optimal path to match two different fingerprints 0. In the DP processing, GDS are extracted from the registered image and the login image, and the values are posited on the x and y axis respectively. FastNcc is first used to get which row of login image is most similar to the first row of registered image, and set this point to starting point. If the starting point is origin, the optimal path will fall in central of coordinate system because of the GDS property. Therefore, a range for operation is set to save the calculation load and time as shown in Fig. 8.

The breadth of the range $[i_1, j_1]$ is set as 20. Let CLI be the column number of the login image, IP be the initial position, and CRI be the column number of the registered image. Thus, the slope of the range can be calculated in Formula 10.

$$m=\frac{CLI-IP}{CRI} \qquad (10)$$

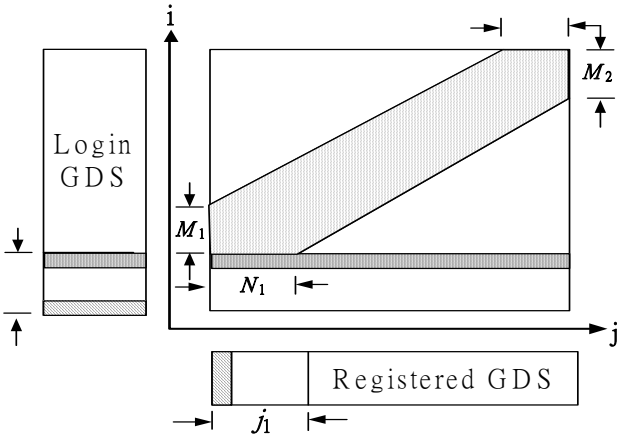

Fig.8. DP Matching stage.

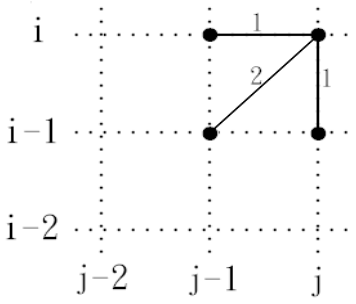

Fig. 9. DP matching path.

The optimal path searching is shown in Fig. 9. This type of path searching is not only a simpler framework but also having a lighter calculation load. For convenient, the operation range of the minimal cumulative path is divided into three segments for calculation as shown in Fig. 10.

The minimal cumulative distance of DP is calculated based on the processing model as well as the Formulas 11-14. And then, the optimal path is derived. Let d(i, j) be the distance between the column j of the registered pattern and the column i of login pattern, g(i, j) be the minimal cumulative distance

between the column j of the registered pattern and the column i of login pattern, and G(i, j) be the minimal cumulative distance after normalization.

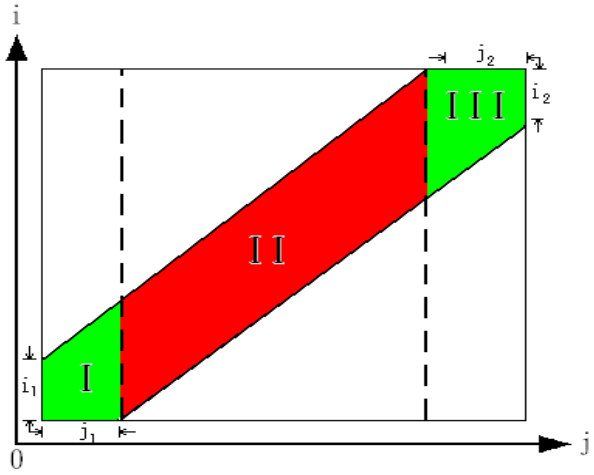

Fig. 10. Calculation segments for the minimal cumulative path

$$\text{Initial conditions：} \begin{cases} g(0,0)=0 \\ g(i,0)=0 \\ g(j,0)=0 \end{cases} \qquad (11)$$

Let i=1,2,…,M_1 and j=1,2,…M_2, where M_1 and M_2 refer to the size of fingerprints, (M_1-20) ≤ i ≤ M_1 and (M_2-20) ≤ j ≤ M_2 .

$$g(i,j)=\max\begin{Bmatrix} g(i-1,j)+d(i,j) \\ g(i-1,j-1)+2\cdot d(i,j) \\ g(i,j-1)+d(i,j) \end{Bmatrix} \qquad (12)$$

$$G(i,j)=\frac{g(i,j)}{i+j} \qquad (13)$$

$$D=\min\begin{Bmatrix} G(i,J) \\ G(I,j) \end{Bmatrix} \qquad (14)$$

3.6 Threshold setting

In the fingerprint recognition system, the threshold value plays a crucial role 0. The values will affect the verification rate. Thus, in this study, experiments are done to set the threshold value of FastNCC [4][5][9].

The values will be converged to a certain interval after the same fingerprints are matched. The threshold value can be figured out as long as the interval is defined. The fingerprints of 20 persons are used to get the interval. The matching was done for these fingerprints pair wisely, and calculated all correlation coefficients and standard deviations. From the distribution of these values, the convergence interval is defined here. After finding the interval, the FastNCC threshold is defined as follows. Let $T_{FastNCC}$ be the threshold of FastNCC, ARC be the average of correlation coefficients, and ASD be the average of SD.

$$T_{FastNCC}=ARC-ASD \qquad (15)$$

4 Experimental results

MBF310 of Fujitsu Japan [2]-[5] is a type Line-sensor and employed in this study. Moreover, the Line-Sensor MBF310 and USB module of SMI combined into a USB fingerprint sensor is our experiment platform [2]-[5] as shown in Fig. 11. Then, 80 person temples are built as fingerprint database. Every person was swept 20 times for one's fingerprints, and 1600 fingerprints are gotten totally. Then, the fingerprints of 20 persons are taken in this database to set thresholds, and the rest are taken to do the matching experiments to get the verification rate, error acceptance rate and error rejection rate.

Fig. 11. MBF310 and USB module of SMI combined into a USB fingerprint system.

For judging the fingerprint sweeping rate, an average overlap amount AOL is defined as follow.

$$\text{AOL}=\frac{1}{SN-1}\sum_{k=1}^{SN-1}R_k \qquad (16)$$

We took 100 fingerprints including fast and slow sweeping rate, and drew the time of reconstruct image and the average overlap amount in coordinate system after running the FastNCC. The result is shown in Fig. 12.

Let *SN* be the number of available segment and R_k be the number of overlapping rows between the k^{th} and $(k+1)^{th}$ segments. 100 fingerprints were taken to include the fast and slow sweeping rate, and drew the time of reconstruction image and the average overlap amount in coordinate system after running FastNCC. From the results, we know that they are the exponential relation. If the average overlap amount is less than 4, between [5.2, 6.3], or bigger than 6.6, the sweeping rate is fast, normal, or slow, respectively.

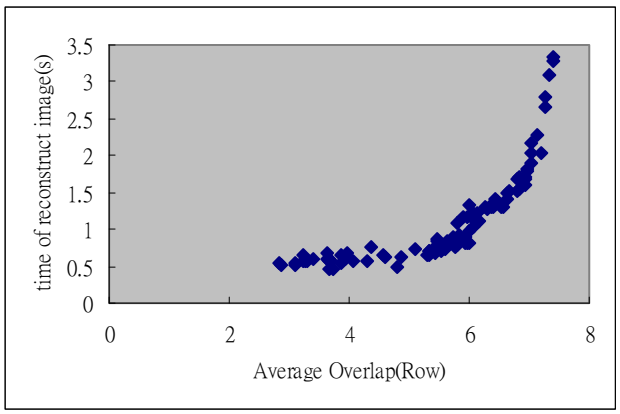

Fig. 12. The relation of the time of reconstruct image and the average overlap amount

4.1 Experiments for threshold setting

The fingerprint data of 20 persons are taken from fingerprint database and did DP matching mutually with the FastNCC algorithm. 190 correlation coefficients and standard deviations are gotten and then calculated their average as shown in Figures 13 and 14.

We took the fingerprint datum of 20 persons from fingerprint database and did DP matching mutually with FastNCC algorithm. We got 190 correlation coefficients and standard deviations, and calculated their average. Fig. 15 is the distribution of standard deviations average.

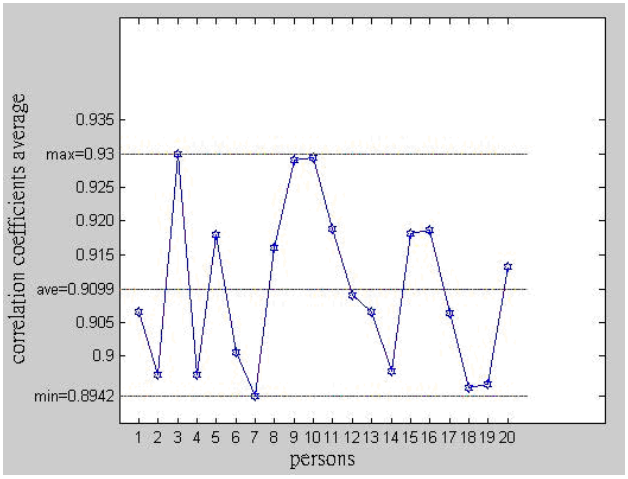

Fig. 13. Distribution of correlation coefficients average.

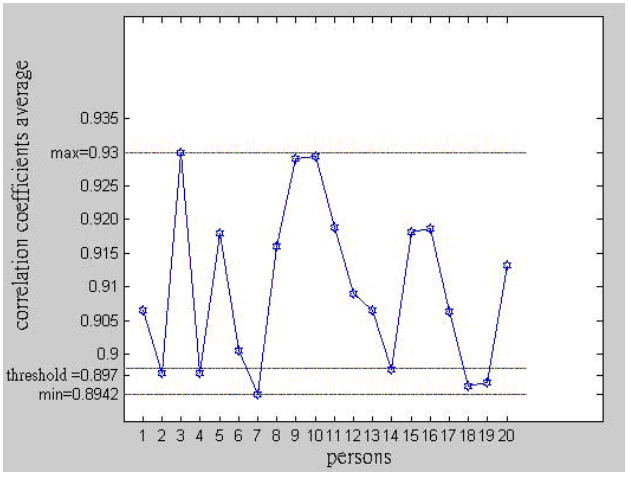

Fig. 14. Distribution of correlation coefficients and threshold.

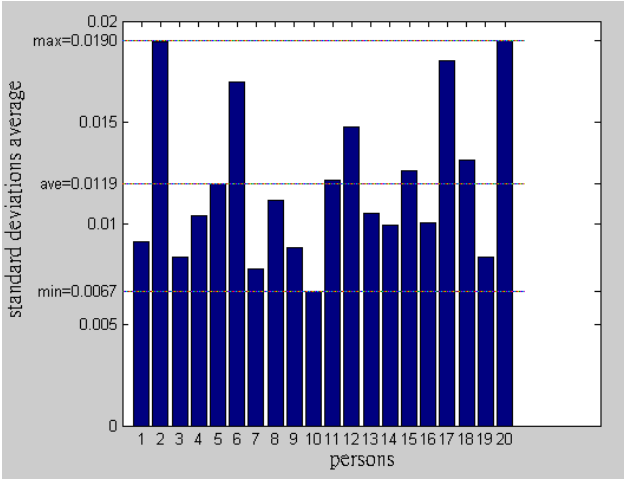

Fig. 15. The distribution of standard deviations average

The thresholds of Fig. 14 are calculated using formula 13. From Figures 13 and 14, we know that the most averages of correlation coefficients are bigger than thresholds. Those correlation coefficients less than threshold may result from skew, blur, or something else when fingerprint swept.

4.2 Experiments for matching

The Fingerprints of 60 persons are taken from the fingerprint database, and 3 of 20 fingerprints of each person were taken as registered fingerprints, the rest 17 fingerprints of each person were for login matching. The verification of the error acceptance and rejection rate is used as the performance indices to test and verify the performance of the system.

The matching experiments with both FastNCC and Sum of Squared Difference (SSD) [3][4][9] algorithms were done for comparing issue. Because fingerprints with three fingerprints of everyone have been registered, the performance indices of one pass, two pass, and three pass were calculated. Three, one, two, and three, passes as the login fingerprint matching one of the registered fingerprints were defined here. The performance indices, verification rate, error acceptance rate, and error rejection rate, are defined as follows.

$$VR = \frac{\text{Corrected Verifications Pieces}}{\text{Total Pieces Of Personal}} \times 100\% = \frac{\text{Corrected Verifications Pieces}}{17 \times 60} \times 100\% \quad (17)$$

$$EAR = \frac{\text{Error Acceptances Pieces}}{\text{Total Pieces Of Other Persons}} \times 100\% = \frac{\text{error acceptances}}{20 \times 59 \times 60} \times 100\% \quad (18)$$

$$ERR = \frac{\text{Error Rejections}}{\text{Total Pieces Of Personal}} \times 100\% = \frac{\text{Error Rejection}}{17 \times 60} \times 100\% \quad (19)$$

Figures 16, 17, 18 show the number of corrected verifications, error acceptances, and the performance indices respectively using the FastNCC algorithm; Figures 19, 20, and 21 show the number of corrected verifications, the number of error acceptances, and the performance indices respectively using the SSD algorithm.

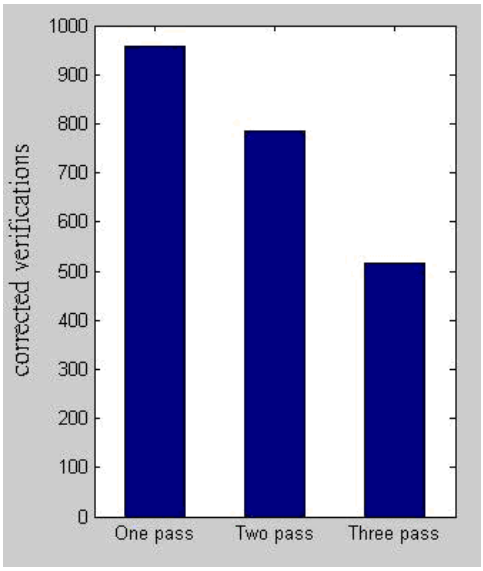

Fig. 16. Number of corrected verifications using FastNCC.

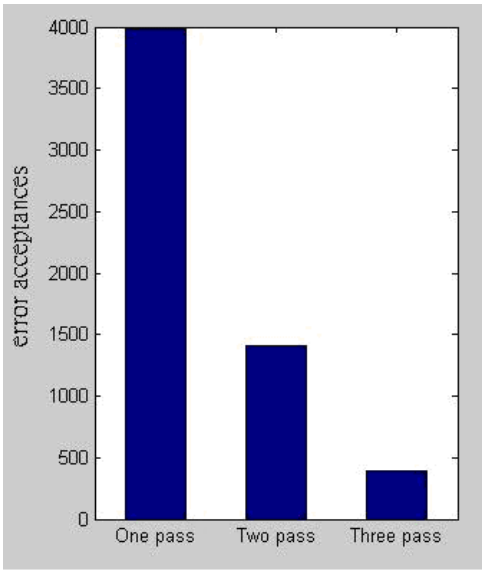

Fig. 17. Nnumber of error acceptances using FastNCC.

In our experiments, two different fingerprint matching algorithms are to exploit and demonstrate the verification rate of the system. FastNCC results in 93.8% of the verification rate, while SSD obtains 92.5% of the verification rate.

Therefore, the use of FastNCC receives a higher fingerprint verification rate.

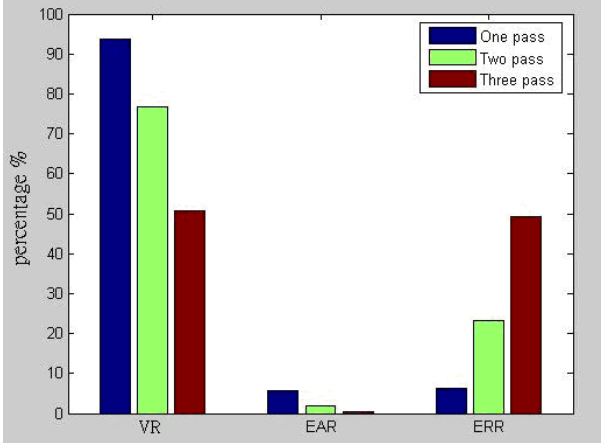

Fig. 18. Performance indices using FastNCC.

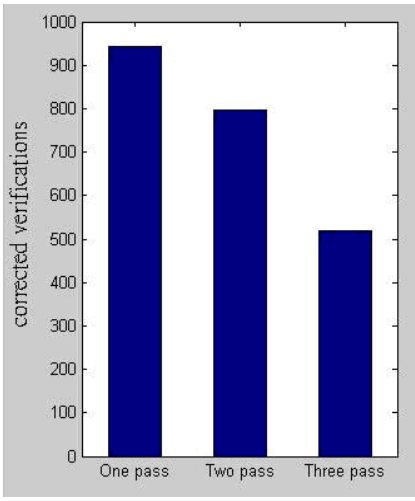

Fig. 19. Number of corrected verifications using SSD.

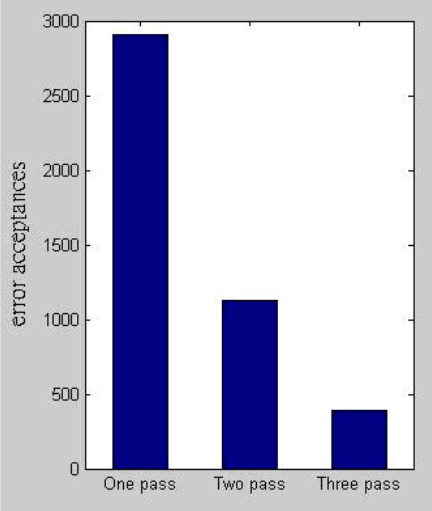

Fig. 20. Nnumber of error acceptances using SSD.

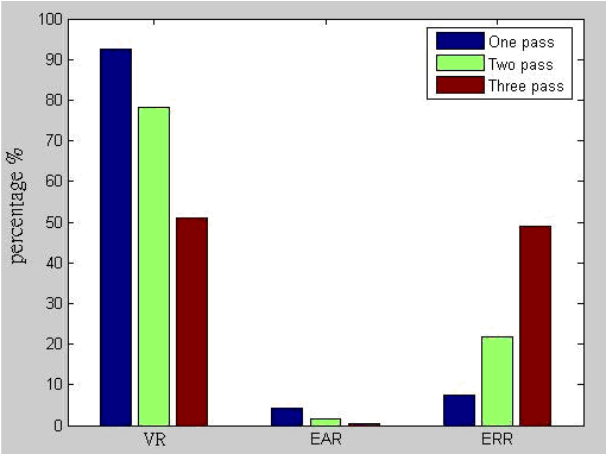

Fig. 21. Performance indices using SSD.

5 Conclusion

The Line-Sensors, MBF310 of Fujitsu Japan, have been adopted to accomplish a fingerprint verification system. Since it requires lower power, smaller size, and fewer costs than the Block-Sensors, it satisfies the requirements of fewer costs and lower power for the system and applies to embedded system.

Two different fingerprint matching algorithms are to exploit and demonstrate the verification rate of the system. FastNCC results in 93.8% of the verification rate, while SSD obtains 92.5% of the verification rate. The use of FastNCC receives a higher fingerprint verification rate.

Acknowledgment

The authors would like to thank the National Science Council of the Republic of China for financially supporting this research under Contract NSC 98-2218-E-002-010- and 98-2221-E-166-008-.
This paper is much indebted to the full support of the NSC HW/SW Co-design of Sweeping Fingerprint System with Template Matching Algorithms Project.

References:

[1] T. H. Coremen, C. E. Leiserson, and R. L. Rivest, Introduction to Algorithms, McGraw-Hill, 2001.

[2] Fujitsu, http://www.fujitsu.com/.

[3] Fujitsu MBF310 Data Sheet.

[4] W. T. Huang, H. W. Yu, Y. W. Shen, H. C. Ling C. C. Chiu, "To Study and Implement the Low-Cost Low-Power and small-size of Finger Printer System," Vol. 38-2, Mar. 2005, pp.23-38.

[5] W. T. Huang, H. W. Yu, W. Y. Chen, and C. H. Chen, "Verification System of Sweeping Fingerprint Base on Template Matching," Vol. 39-1, Oct. 2005, pp.55-70.

[6] J. P. Lewis, "Fast Normalized Cross-Correlation," Industrial Light & Magic.

[7] L. Liu, T. Jiang, J. Yang, and C. Zhu, "Fingerprint registration by maximization of mutual information," IEEE Transactions on Image Processing, Vol. 15, No. 5, May 2006, pp. 1100- 1110.

[8] X. Li , "Low bit rate image coding in the scale space," Data Compression Conference 2002. Proceedings, Apr. 2002, pp. 33-42.

[9] N. Otsu, "A threshold selection method from gray level histogram," IEEE Trans. On Systems, Man, and Cybernetics, SMC-8, 1978, pp. 62-66.

[10] N. Matsumoyo, S. Sato, H. Fujiyoshi, and T. Nmezaki, "Evalution of a Fingerprint Verification Method on LPC Analysis," T .IEE Japan, Vol. 122-C, No.5, 2002, pp.799-806.

[11] Taizo Nmezaki, Shozo Sato, Akihito Kimura, and Noriyuki Matsumoyo, "An Effectively verification Method for Varying the Quality of Fingerprint Image," T. IEE Japan, Vol. 122-C, No.7, 2002, pp.1127-1136.

[12] S.-Y. Tan, W.-T. Huang, C.-H Chen, Y.-J. Chang, "Sweeping Fingerprint Verification System Based on Template Matching", VLSI and Signal Processing (ICNVS '10), Feb. 2010, pp.44-48.

[13] E. Dovgan, M. Gams, "Intelligent Entry Control", WSEAS Transactions on Computers, Issue 2, Volume 8, February 2009, pp. 344-354.

[14] M. Meghdadi, S. Jalilzadeh, "Validity and Acceptability of Results in Fingerprint Scanners", 7th WSEAS Int. Conf. on MATHEMATICAL METHODS and COMPUTATIONAL TECHNIQUES IN ELECTRICAL ENGINEERING, Sofia, 27-29 Oct. 2005, pp. 259-266.

[15] F. Su, X. Xie and A. Cai, "A Hierachical Fingerprint Matching Method Based on the Minutiae and Ridge Image", Proc. of the 6th WSEAS Int. Conf. on Signal Processing, Computational Geometry & Artificial Vision, Elounda, Greece, August 21-23, 2006, pp.83-88.

[16] M.A. Dabbah, W.L. Woo, and S.S. Dlay, "Computationally Efficient Fingerprint Algorithm for Automatic Recognition" Proceedings of the 5th WSEAS Int. Conf. on SIGNAL, SPEECH and IMAGE PROCESSING, Corfu, Greece, August 17-19, 2005, pp. 90-95.