Improved FingerCode Matching Function

Gustavo de Sá, and Roberto Lotufo
School of Electrical and Computer Engineering, University of Campinas (UNICAMP),
C.P. 6101, 13083-970, Campinas, SP, Brazil
gsa@dca.fee.unicamp.br, and www.dca.fee.unicamp.br/~lotufo

Abstract

FingerCode is a fingerprint correlation matching scheme that relies on texture information. In this scheme, the oriented components are extracted from a fingerprint image using a bank of Gabor filters, and a directional texture feature vector is computed for each oriented component. The feature vectors from the input and template images are compared and a matching score is obtained. Here we explore ways to improve the matching score for the FingerCode method by using more complex matching functions. The best results were obtained by applying a nonlinear function to the texture values and weighting the texture vectors based on the spatial distribution.

1. Introduction

The use of biometric systems, in particular fingerprint-based systems, is increasing. Because fingerprints are unique, permanent, and easy to collect identifiers of individuals, systems based on fingerprint patterns show good performance and reliability [1]. These characteristics, which underpin the push towards exploiting fingerprint systems in civilian applications, are the same as have been used for more than a century in forensic applications.

A fingerprint is the pattern of alternating ridges and valleys on the finger surface. At the global level this pattern has distinct shapes and characteristics called singularities, such as high curvature and predominant orientation. At the local level, anomalies in the pattern, called minutiae, can be found; examples of minutiae are terminations and bifurcations. Fingerprint systems can be broadly categorized into correlation-based techniques [2-4] and minutiae-based techniques [5-6] depending on whether they consider singularities or minutiae respectively. The majority of fingerprint applications use minutiae in the matching process, while the correlation-based technique is mostly used to classify fingerprints [7-9]. Each of these techniques does not account for information on the other level: minutiae based methods do not utilize the full information from the global level and correlation based methods do not use the detailed local level information. Hence, systems that combine the two techniques could potentially be of great utility [9].

Based on a model for iris recognition [11-13], Jain et al. [14] developed a texture-based method for fingerprint recognition called FingerCode. In their method, the pair of fingerprints under consideration is first aligned according to their core points, then oriented texture vector sets are extracted by a bank of Gabor filters, and finally the Euclidean distance of the oriented texture vector sets is computed. Subsequently, Jain et al. [15] proposed a hybrid FingerCode method that combines minutiae and texture (i.e. local and global information). In this variation, the minutiae points are used to align the fingerprints and the texture vector sets are used to obtain the final match score [16].

Motivated by our perception from past work that certain aspects of fingerprints have greater discriminating power than others, in the present study we explored the use of more complex FingerCode matching functions. The best results were obtained using two types of function: a nonlinear function (e.g., an exponential) as a substitute for the texture values; and applying weights to the texture vectors using statistical measures extracted from the set of texture vectors (e.g., mean and standard deviation) based on the spatial distribution. The statistical measures were obtained in two ways: from an area around a cell in a specific oriented component, or from a cell in the same position in all oriented components.

In our experiments, we use the hybrid FingerCode of Ross et al. [16] because it is more accurate than the FingerCode method alone. This paper is organized as follows: Section 2 presents the FingerCode method and its hybrid variation; the alternative matching score functions are described in Section 3; the experimental results obtained using the new matching score functions are given in Section 4; and finally, our conclusions are presented in Section 5.

2. FingerCode

The FingerCode method is a correlation based technique, where a small circular area around the core point is tessellated in an arc fashion and filtered by an oriented Gabor filter bank. For each cell in the oriented components, a value is computed that constitutes the texture feature vector. Two fingerprints are aligned by their core points and their texture vectors are compared, generating a score that indicates whether they belong to the same finger. The FingerCode hybrid variation differs from the FingerCode method in that it uses the entire fingerprint area, which is tessellated in a square rather than an arc fashion, and, most importantly, the minutia points are used to align the fingerprints.

In the hybrid variation, the fingerprint image is tessellated in rectangular cells. The block size used in the present work was 16x16 pixels, which corresponds to approximately 1.5 times the width of the ridge-valley structure in a 500 dpi image. Then, the fingerprint is separated from the background in the segmentation process, while avoiding extraction of features from invalid cells (cells in which one or more pixels belong to the background). In this process, a morphological scheme is used to establish the fingerprint mask. To reduce intensity variations due to differences in finger pressure, the fingerprint image is normalized using the approach presented in [17].

After extracting the minutiae points from the input image, the rotation and translation parameters are generated that give the best correspondence with the set of minutiae points from the template image. The aligned image is filtered by an oriented Gabor filter bank. The Gabor filter is frequency- and orientation-selective and consists of a sinusoidal wave modulated by a Gaussian envelope. The bank is formed by eight Gabor filters in directions Θ = {0°, 22.5°, 45°, 67.5°, 90°, 112.5°, 135°, 157.5°}. The chosen frequency (f_0 = 0.1 pixel^{-1}) was based on the mean inter-ridge distance of approximately 10 pixels for a 500 dpi image. The real part of the Gabor filter is given by:

$$G(x,y)=\exp\left(-\frac{1}{2}\left(\frac{x_\theta^2}{\sigma_x^2}+\frac{y_\theta^2}{\sigma_y^2}\right)\right)\cos\left(\omega_0 x_\theta\right)$$
$$x_\theta = x\sin\theta + y\cos\theta \qquad (1)$$
$$y_\theta = x\cos\theta - y\sin\theta$$

where σ_x and σ_y are the standard deviations of the Gaussian envelope along the x- and y-axes respectively, θ is the direction of the filter, and $\omega_0 = 2\pi f_0$. The filtering extracts eight oriented image components in each direction. For each cell, C, the feature value, $F_{\rho\theta}$, is computed:

$$F_{\rho\theta}=\sqrt{\frac{1}{K_{\rho\theta}}\sum_{(x,y)\in C_{\rho\theta}}\left(C_{\rho\theta}(x,y)-\overline{C}_{\rho\theta}\right)^2}$$
$$\overline{C}_{\rho\theta}=\frac{1}{K_{\rho\theta}}\sum_{(x,y)\in C_{\rho\theta}}C_{\rho\theta}(x,y) \qquad (2)$$

where $C_{\rho\theta}$ is the cell at position ρ in the oriented component θ, $C_{\rho\theta}(x,y)$ is the pixel value at position (x,y) in cell $C_{\rho\theta}$, $K_{\rho\theta}$ is the number of pixels from cell $C_{\rho\theta}$, and $\overline{C}_{\rho\theta}$ is the mean value of the pixels from cell $C_{\rho\theta}$. FingerCode is the set of eight feature vectors composed of all cell values, $F_{\rho\theta}$. The difference vector, E, is formed by the absolute difference between corresponding cells from the input, F, and template, T, FingerCode feature vectors:

$$E_{\rho\theta}=\left|F_{\rho\theta}-T_{\rho\theta}\right| \qquad (3)$$

where $E_{\rho\theta}$, $F_{\rho\theta}$ and $T_{\rho\theta}$ are the cell values at position ρ in direction θ from the difference, input and template feature vectors, respectively. The matching score S is expressed as the sum of all differences normalized by the number of valid cells, V, given by the intersection of the two fingerprint masks:

$$S=\frac{1}{V}\sum_{\rho\in \mathrm{P}}\sum_{\theta\in\Theta}V_{\rho\theta}E_{\rho\theta} \qquad (4)$$

where $V_{\rho\theta}$ indicates whether the cell at position ρ in direction θ is valid ($V_{\rho\theta}$ = 1) or invalid ($V_{\rho\theta}$ = 0), and P is the set of all cell positions in one oriented component. If the score S is below a predetermined threshold, the two fingerprints are considered to belong to the same finger; otherwise they are considered to belong to different fingers.

3. Matching Functions

The use of more complex functions to compute the difference feature vector can improve the discrimination of fingerprint matching. In preliminary testing, we found that among the various approaches that can be used to calculate the matching score, the best results were obtained by the following two methods: replacing the difference feature cell values with new values given by a nonlinear function; and applying weights to the texture or difference feature cells using statistical measures extracted from them. In the second case the statistical measures pick up characteristics of spatial distribution of the feature vector.

3.1. Nonlinear Functions

One way to improve the matching score is to apply a nonlinear function, f_α, to the difference feature values so as to create new ones. The new matching score is:

$$S_f = \frac{1}{V} \sum_{\rho \in \mathrm{P}} \sum_{\theta \in \Theta} V_{\rho\theta} f_\alpha \left(E_{\rho\theta} \right) \tag{5}$$

The functions chosen give more importance of differences than of similarities between the features vectors. Two of the best functions, f_α, identified in the present work were a polynomial function of the form:

$$f_\alpha \left(x \right) = x^\alpha \tag{6}$$

where α is the polynomial degree; and an exponential function of the form:

$$f_\alpha \left(x \right) = \frac{e^{\alpha x} - 1}{e^\alpha - 1} \tag{7}$$

where α is the coefficient degree. In these equations, $f_\alpha(x)$ gives the new values from the difference cells.

3.2. Statistical Weighting

Statistical measures were used to weight the feature vectors based on the spatial distribution of the features. These measures can be extracted from and applied to each of the difference feature vector, Equation (8).

$$E'_{\rho\theta} = R_{\rho\theta} \left| F_{\rho\theta} - T_{\rho\theta} \right| \tag{8}$$

Two ways to extract the measures from the features vectors were explored: from the neighborhood of a cell in a specific oriented component, or from cells in the same position across all oriented components. The neighborhood, N_ρ, of cell ρ is given by the eight cells surrounding that cell and by cell ρ itself in a specific oriented component $\theta \in \Theta$. Thus, as shown in Figure 1, the neighborhood of a cell is $N_\rho = \{\rho, \rho_1, \rho_2, \rho_3, \rho_4, \rho_5, \rho_6, \rho_7, \rho_8\}$. The measures extracted from cells at the same position across all directions are obtained from the values of P_ρ, where $P_\rho = \{\rho_{\theta 1}, \rho_{\theta 2}, \rho_{\theta 3}, \rho_{\theta 4}, \rho_{\theta 5}, \rho_{\theta 6}, \rho_{\theta 7}, \rho_{\theta 8}\}$ as shown in Figure 2.

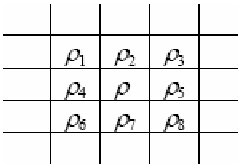


Figure 1. Neighborhood of cell ρ, $N_\rho = \{\rho, \rho_1, \rho_2, \rho_3, \rho_4, \rho_5, \rho_6, \rho_7, \rho_8\}$.

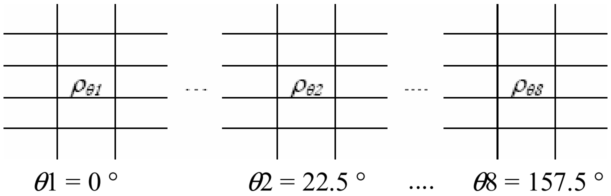


Figure 2. All cells ρ across all directions θ, $P_\rho = \{\rho_{\theta 1}, \rho_{\theta 2}, \rho_{\theta 3}, \rho_{\theta 4}, \rho_{\theta 5}, \rho_{\theta 6}, \rho_{\theta 7}, \rho_{\theta 8}\}$.

The weights, R, can be given by the mean or standard deviation of the feature values from the neighborhood of a cell (Equations (9) and (10), respectively); or the mean or standard deviation of the feature values from the cells in the same position (Equations (11) and (12), respectively).

$$\overline{R}_{\rho\theta} = \frac{1}{9} \sum_{r \in N_\rho} E_{r\theta} \tag{9}$$

$$\hat{R}_{\rho\theta} = \sqrt{\frac{1}{9} \sum_{r \in N_\rho} \left(E_{r\theta} - \overline{R}_{r\theta} \right)^2} \tag{10}$$

$$\overline{R}_\rho = \frac{1}{8} \sum_{q \in \Theta} E_{\rho q} \tag{11}$$

$$\hat{R}_\rho = \sqrt{\frac{1}{8} \sum_{q \in \Theta} \left(E_{\rho q} - \overline{R}_\rho \right)^2} \tag{12}$$

Other measures tested were the median, maximum, minimum, and combinations of two of these measures. When the measures are extracted from cells in the same position across all directions, the weight R is the same for all oriented components.

4. Experimental Results

We used the databases of the Fingerprint Verification Competition 2002 (FVC2002) to test the gain obtained using the proposed matching functions. The FVC2002 databases consist of four distinct databases (Db1–Db4), each of which contains data collected using a different sensor/technology.

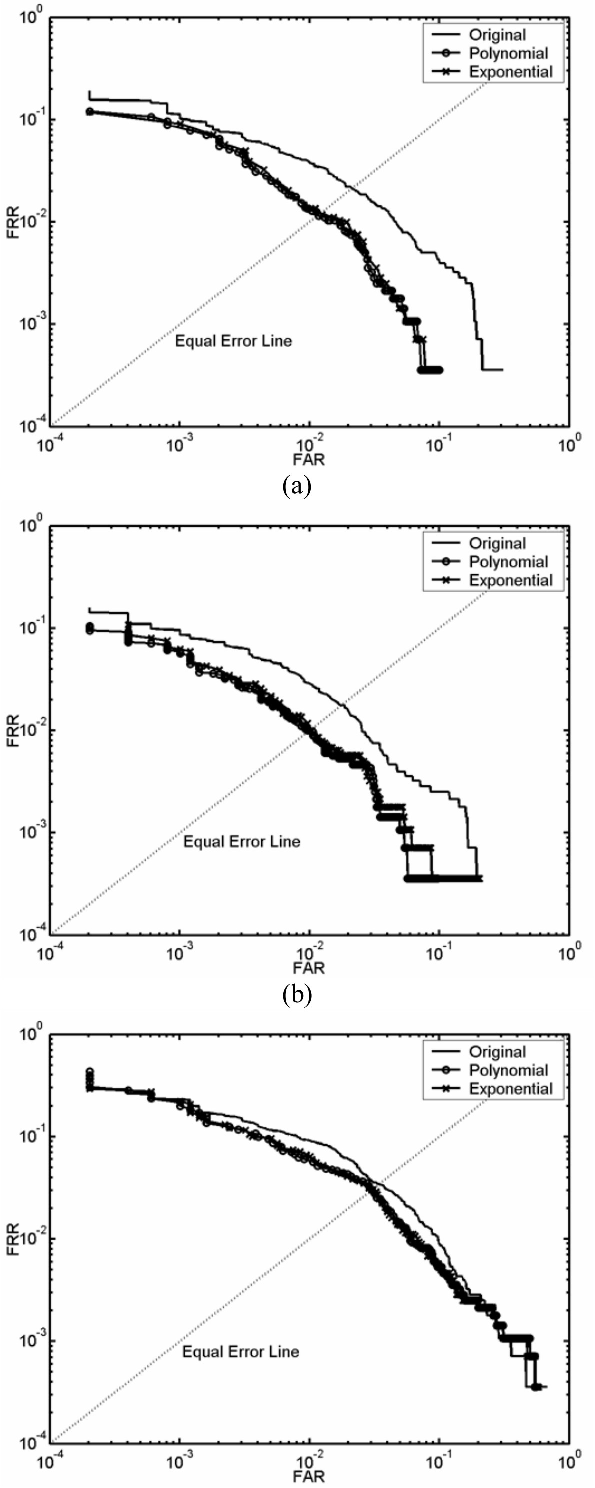
Each database contains 8 fingerprint samples for each of the 100 distinct fingers. In the present work, we compared the eight samples of the same finger with each other, and also compared the first sample of each finger with the first sample of all others fingers. These combinations result in 2800 comparisons between different samples of the same finger and 4950 comparisons between the pairs of first samples of different fingers.

To compare the new matching functions, we used the Receiver Operating Characteristic curve (ROC), which took the form of a plot of the False Accept Rate (FAR) against the False Reject Rate (FRR) for various thresholds. The FAR is given by the percentage of comparisons between different fingers where the matching score is below the threshold, i.e., false matching. The FRR is given by the percentage of comparisons between different samples of the same finger where the matching score is above the threshold, i.e., false non-matching. The matching scores were normalized to [0 1]. The equal error rate (EER) is the point where the FRR and FAR are equal.

The first experiment replaces the difference cell values using nonlinear functions. Among the various functions tested, the best results were obtained using functions for which the effective replacement values had an exponential shape. These functions emphasize the contrast of the difference feature vector. Table 1 shows the Equal Error Rate (EER) for the hybrid FingerCode method, as well as for the polynomial and exponential replacing functions (Equations (6) and (7), respectively). Figure 3 shows the ROC curves obtained for the functions listed in Table 1.

Table 1. Equal Error Rate (EER) for non-linear functions, where α is the polynomial degree or exponential coefficient degree used.

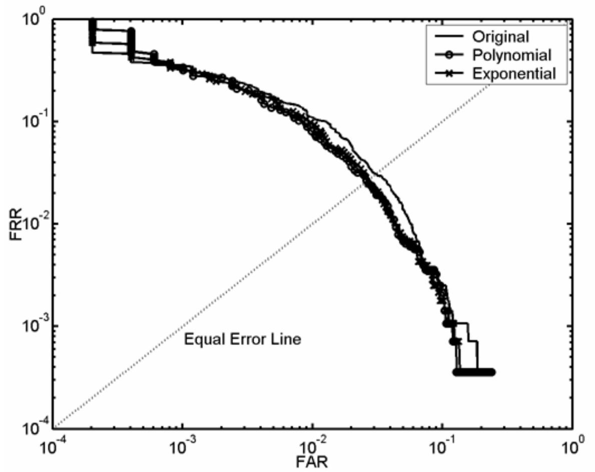
	Original	Polynomial	Exponential
Db1	2.12%	1.17% (α= 4)	1.19% (α= 17)
Db2	1.76%	0.99% (α= 3)	1.01% (α= 14)
Db3	3.43%	3.04% (α= 4)	3.05% (α= 14)
Db4	3.06%	2.66% (α= 3)	2.67% (α= 12)



(a)

(b)

(c)



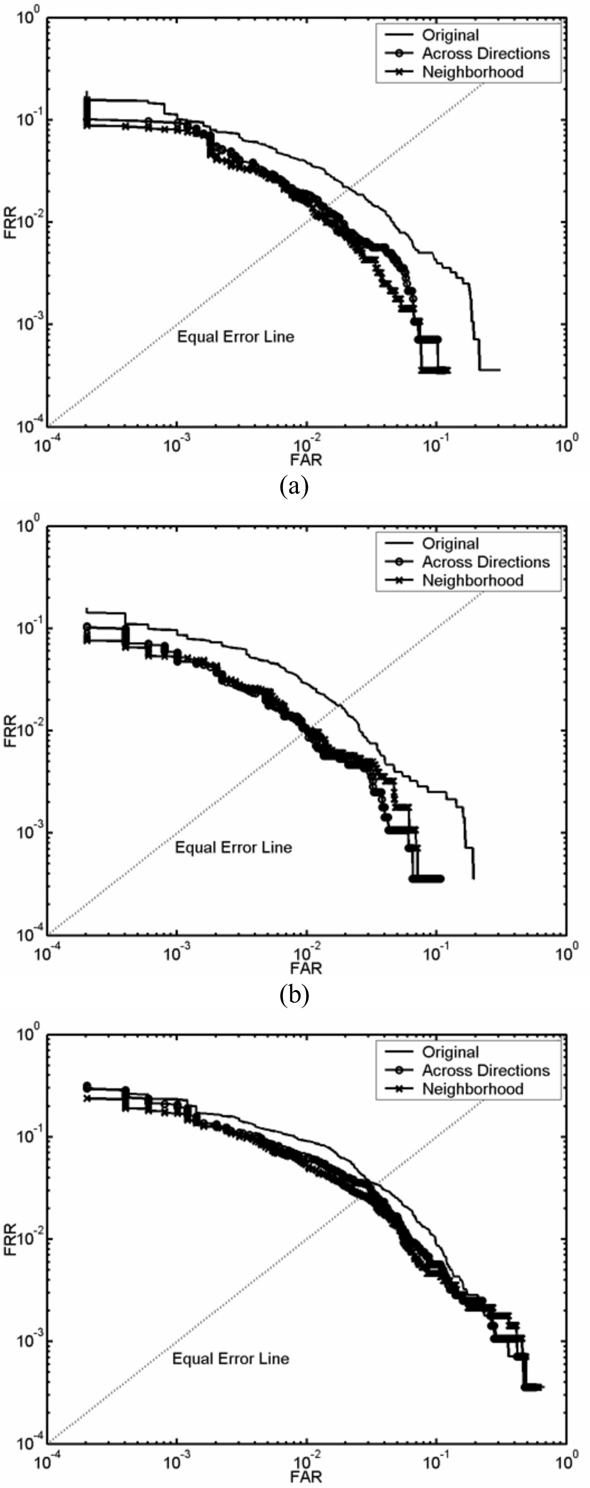
(d)

Figure 3. ROC curves for non-linear functions: (a) Db1; (b) Db2; (c) Db3; and (d) Db4.

In the second experiment, we tested several approaches to weighting the cells using statistical measures extracted from the texture or difference feature vectors: mean, standard deviation, median, maximum, minimum, and combinations of two of these measures. These measures were used directly, either by giving more weight to cells with high values of the measure, or inversely, giving more weight to cells with low values. For Db1, Db2, and Db3, the statistical measures for cell weighting that gave the best responses were the standard deviation of the neighborhood around a cell in a specific oriented component (Equation 10) and the standard deviation of the cells in the same position across all oriented components (Equation 12), extracted from and applied to the difference feature vector (Equation 8), used in the direct way. The first one analyzes the uniformity of the surrounding area of the point, while the second one analyzes the direction force in the point. Table 2 shows the EERs for the hybrid FingerCode method (Original), the standard deviation of the neighborhood of a cell in a specific direction (Neighborhood), the standard deviation of the cells in the same position across all directions (Across Directions). Figure 4 shows the ROC curves for the cases listed in Table 2.

Table 2. Equal Error Rate (EER) for statistical weighting.

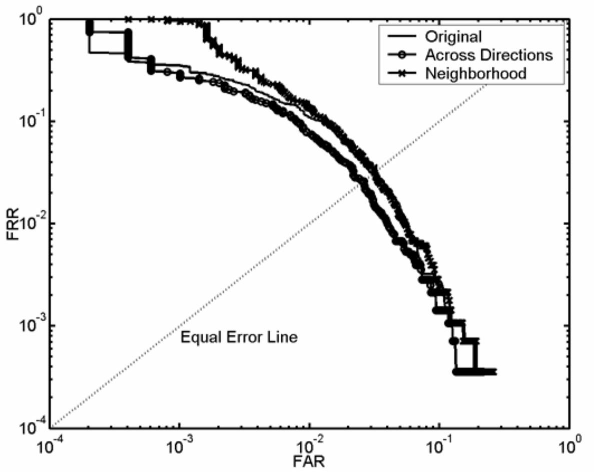
	Original	Neighborhood	Across Direction
Db1	2.12%	1.19%	1.35%
Db2	1.76%	0.99%	1.01%
Db3	3.43%	2.69%	3.11%
Db4	3.06%	3.17%	2.61%



(a)

(b)

(c)



(d)

Figure 4. ROC curves for statistical weighting: (a) Db1; (b) Db2; (c) Db3; and (d) Db4.

In Db4, the Neighborhood case worsened the response, and only the Across Directions case worked well. For Db4, the best results (EER = 1.80%) were obtained by weighting the cells using the maximum cell value from the neighborhood of a cell in a specific direction, which was extracted from and applied to each of the texture vectors; by contrast, the gain obtained using this measure was very small for the other databases. However, given that Db4 is composed of synthetic fingerprints, we do not recommend the use of this measure.

5. Conclusions

In the present work we have established that using more complex functions to match the texture feature vectors can improve the discrimination of the FingerCode method. A large range of matching functions could potentially be used for this purpose; here we have presented two such functions: replacing the cell values of the difference vector by a non-linear function; and weighting the cells using statistical measures extracted from the feature vector. Application of the method to the four data sets within the FVC2002 database showed that the gain achieved depends on the database used. Db1 and Db2 (optical sensors) showed the greatest gain (~40% improvement in the EER) for both approaches. Db3 (capacitive sensor) showed a lower gain than Db1 and Db2 (~10% improvement in the EER). Db4 (synthetic fingerprints) showed behavior that was very different from that observed for the other databases in the experiments using statistical measures.

The process of replacing cell values with new values given by a nonlinear function is faster and easier than the process of weighting cells based on statistical measures. In the nonlinear function approach, however, it is necessary to find the optimum parameters of the function to be used, which requires a good representative sample of the database. If new input fingerprints diverge from this representative sample, the response of the function will diminish. Here we found that the best fitting functions had exponential-like characteristics that cause low values to be decreased to near zero, and high values to be markedly increased.

Compared to the nonlinear function approach, the use of statistical measures has the advantage of being less sensitive to the specifics of the database. The same measures work well for almost all types of fingerprints. Here we found that the two best statistical measures were the standard deviation from the neighborhood of a cell in a specific direction, and the standard deviation from the cells in the same position across all oriented components, extracted from and applied to the difference feature vector. The former measure gave superior results for all databases except the database of synthetic fingerprints. The gain achieved using a combination of statistical measures was small compared to that achieved using a single measure, and the use of multiple measures has the disadvantage of being more complex.

6. References

[1] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, “Handbook of Fingerprint Recognition”, Springer, 2003.

[2] M. U. Munir and M. Y. Javed, “Fingerprint Matching Using Gabor Filters”, National Conference on Emerging Technologies, pp. 147-151, 2004.

[3] K. Venkataramani, and B. V. K. V. Kumar, “Fingerprint Verification Using Correlation Filters”, Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication (AVBPA), pp. 886-894, 2003.

[4] M. Tico and P. Kuosmanen, “Fingerprint Matching Using an Orientation-Based Minutia Descriptor”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 25, no. 8, pp. 1009-1014, 2003.

[5] X. Jiang and W. Y. Yau, “Fingerprint Minutiae Matching Based on the Local and Global Structures”, Proc. Int. Conf. Pattern Recognition, vol. 2, pp. 1038-1041, 2000.

[6] D. Maio and D. Maltoni, “Direct Gray-Scale Minutiae Detection in Fingerprints”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 19, no. 1, pp. 27-40, 1997.

[7] J. K. Min, J. H. Hong, and S. B. Cho, “Effective Fingerprint Classification by Localized Models of Support Vector Machine”, International Conference on Biometrics (ICB), pp. 287-293, 2006.

[8] Q. Zhang, K. Huang, and H. Yan, “Fingerprint Classification Based on Extraction and Analysis of

Singularities and Pseudoridges", Pan-Sydney Area Workshop on Visual Information Processing (VIP), 2001.

[9] A. Senior, "A Combination Fingerprint Classifier", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 10, pp. 1165-1174, 2001.

[10] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "FingerCode: A Filterbank for Fingerprint Representation and Matching", Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, pp. 187-193, 1999.

[11] J. G. Daugman, "High Confidence Recognition of Persons by a Test of Statistical Independence", IEEE Trans. Pattern Anal. Machine Intell., vol. 15, no. 11, pp. 1148-1161, 1993.

[12] J. Daugamn, "Iris Recognition", American Scientist, vol. 89, pp. 326-333, 2001.

[13] J. Daugamn, "How Iris Recognition Works", IEEE Transactions on Circuits and Systems for Video Technology, vol. 14, no. 1, pp. 21-30, 2004.

[14] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-Based Fingerprint Matching", IEEE Trans. on Image Processing, vol. 9, no. 5, pp. 846-859, 2000.

[15] A. Jain, A. Ross, and S. Prabhakar, "Fingerprint Matching Using Minutiae and Texture Features", Proc. Int. Conf. on Image Processing (ICIP), pp. 282-285, 2001.

[16] A. Ross, A. Jain, and J. Reisman, "A Hybrid Fingerprint Matcher", Pattern Recognition, vol. 36, pp. 1661-1673, 2003.

[17] L. Hong, Y. Wan, and A. K. Jain, "Fingerprint Image Enhancement: Algorithms and Performance Evaluation", IEEE Transactions on PAMI, vol. 20, pp. 777-789, 1998.