International Journal of Information and Communication Technology Research

http://www.esjournals.org

A New Approach in Fingerprint Matching based on a Competitive Learning Algorithm

A.N. Ouzounoglou[1], P.A. Asvestas[2] and G.K. Matsopoulos[1]

[1]National Technical University of Athens, School of Electrical and Computer Engineering,
9 Iroon Polytechniou str, 157 80, Zografos, Greece
[2] Department of Medical Instruments Technology, School of Technological Applications,
Technological Educational Institute of Athens, Ag. Spyridonos str, 122 10, Egaleo, Greece

ABSTRACT

A fingerprint identification algorithm based on a modification of the Competitive Learning Algorithm developed originally by T. Kohonen is presented. Given a pair of fingerprint images (template and input image) and a set of minutiae in the template image, the algorithm attempts to extract the corresponding points (if exist) in the input image. This is accomplished by determining the parameters of local transformations that match regions around the interest points of the template image to their corresponding regions in the input image. A properly defined similarity measure is used to conclude whether the two fingerprint images are from the same person or not. The advantage of the proposed fingerprint matching method, compared to corresponding previous similar methods, is based on the non-necessity of constructing minutiae points in the input fingerprint image. The proposed fingerprint matching algorithm was evaluated using the DB3 database of Finger Print Verification Competition (FVC2004). Image pairs of both known and unknown transformations were used for the testing. The overall performance for fingerprints of the same and different fingers was calculated in terms of Equal Error Rate (EER) and was equal to 0.0586.

Keywords— Fingerprint image identification; minutiae point correspondence; self-organizing maps; similarity measure.

1. INTRODUCTION

Fingerprints are flow-like ridges existing on human fingers and are one of the basic biometric features used in personal identification processes. An identification system recognizes an individual by searching the entire enrollment template database to achieve matching. This process is also known as fingerprint matching. In this paper a Self-Organizing Maps (SOMs) Algorithm is described to achieve the matching of an input and a template fingerprint image. Minutiae of the template fingerprint images are used as an input to the neural map.

Most fingerprint matching algorithms do not operate directly on grayscale fingerprint images, but require the derivation of an intermediate fingerprint representation. This is accomplished by means of features implemented through an extraction stage. These features are categorized in three different levels: 1) global feature characteristics of a fingerprint (e.g. ridge flow), 2) minutiae (e.g. ridge bifurcations and endings) and 3) features that include all dimensional attributes of the ridge (e.g. ridge path deviation, width, shape, pores, edge contours, breaks), [1]. The fingerprint matching techniques either use one or a combination of the aforementioned levels of fingerprint characteristics.

Jea et al, have developed a minutiae based fingerprint recognition system [2] to address the cases when only partial fingerprint image is available by using a flow network-based matching technique. In ref.[3], a verification method is proposed based on the matching of minutiae located in a region around a core point. A core point corresponds to the center of the north most loop type singularity. This method is faster since during the matching process only a portion of the minutiae of the image were used, although the calculation of the core point is most of the times a hard or even an unaccomplished task. He et al have also developed a minutiae based matching method [4]. This fingerprint verification method uses a fingerprint representation using minutiae and local ridge information. Gu et al, [5], presented a fingerprint matching technique based on the ridge orientation field and the minutiae. Finally, a fingerprint matching system that makes use of all three levels of the fingerprint characteristics has been proposed in ref.[6].

Furthermore, much research effort has been focused on the problem of non-linear distortion in fingerprint matching. Chen et al, [7], developed a two stage fingerprint matching and verification method based on normalized fuzzy-similarity measure. In the first stage, the input and template fingerprint images are aligned using a minutiae-based method and in the second stage the similarity between the two images is assessed using a similarity measure based on fuzzy theory. Ross et al, [8], addressed a thin-plate spline (TPS) function in order to estimate a ridge curve correspondence deformation model for a specific finger. This model requires that several recordings of that finger are available.

A number of other techniques have been also proposed. In ref. [9], the two images are converted to feature vectors, the so called "Finger-Codes", by applying Gabor filters around the detected core points in two images. The estimation of the Euclidean Distance between these "Finger-Codes" was used to achieve matching of the two fingerprint images under comparison. The main drawback of this method is the difficulty to estimate accurately the core point for low quality fingerprint images. Another method has been proposed by Liu et al, [10], according to which the orientation field of the fingerprint images was initially estimated and then the registration was carried out by maximizing the mutual information of the ridge orientation fields. This method performs satisfactorily in cases where the orientation field has been estimated with high accuracy and therefore very high quality fingerprint images are required.

The aim of this paper is to provide a general framework in registering fingerprint images by means of Self Organizing Maps (SOMs) network theory. In the first place, minutiae points are extracted only on the template image and then a novel implementation of the SOM network is utilized to determine the corresponding points on the input image. It should be emphasized that the main advantage of the proposed method, as compared to previous methods, is based on the fact that interest points (minutiae) need to be extracted once only on the template image. This leads to an error-tolerant procedure in estimating precisely the minutiae co-ordinates.

2. METHODOLOGY

The implementation steps of the proposed method are shown in the block diagram of Fig. 1. Hereafter, without loss of generality, we denote the image of the fingerprint acquired during enrollment as the template (Itmp) and the representation of the fingerprint to be matched, that is acquired during identification, as the input image (Iinp).

The proposed Automatic Fingerprint Matching (Identification) Scheme involves:

- Extraction of minutiae only on the template image.
-
- Automatic computation of the corresponding of minutiae in the fingerprint to be matched.

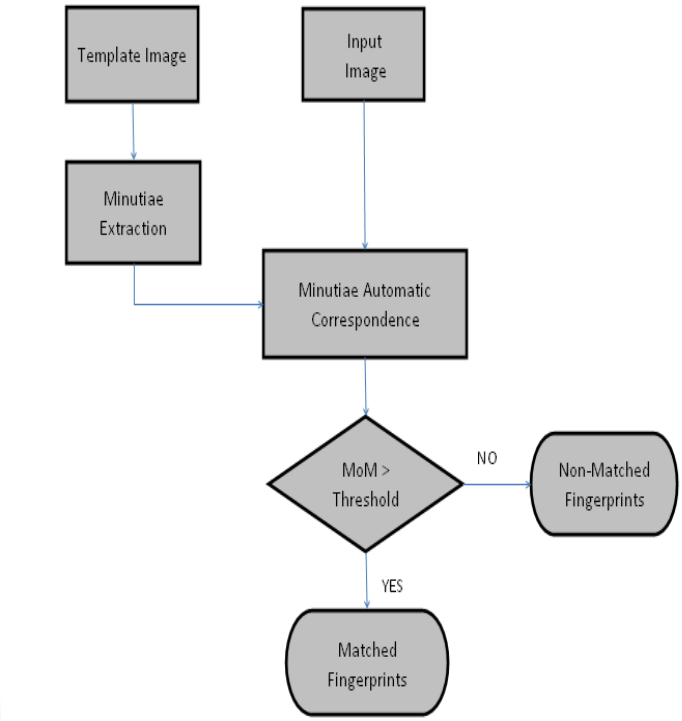


Fig. 1 Block diagram of the proposed Automatic Fingerprint Identification Scheme.

2.1 Minutiae Extraction

In order to extract the minutiae (ridge bifurcation points) in a template fingerprint image, the Crossing Numbers (CN) are calculated [1] around each point and then a minutiae verification method, as proposed by Tico et al, [12], is applied. Firstly, the ridge map fingerprint image has to be computed.

The ridge map is obtained by means of the following procedure:

1. Fingerprint image enhancement, using the method proposed by Hong et al., [13] based on Gabor Filters. (Fig. 2(b)).
2. Binarization of the enhanced fingerprint image, [14]. The binarization is based on the fact that the second derivative of the filtered image (in a direction normal to the orientation field) is positive in the regions of fingerprint ridges. (Fig. 2(c)).
3. Thinning of the fingerprint image is finally achieved by applying the two-subiteration thinning algorithm described in [15] on the binarized image. This algorithm uses a 3×3 thinning operator in the area of pixel P. (see Fig. 2(d)).

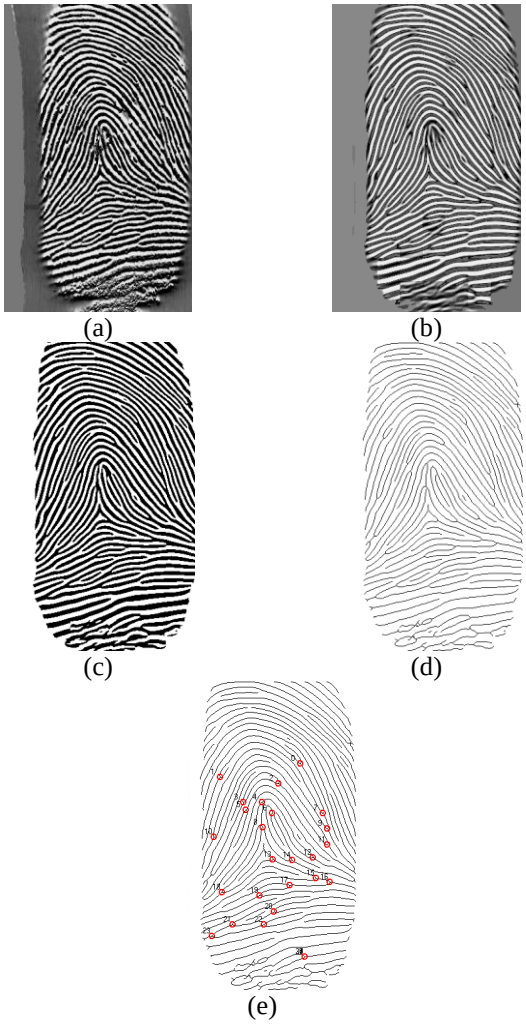


Fig. 2: Extraction of minutiae bifurcation points. (a) Typical fingerprint image. (b) Enhanced fingerprint image. (c) Binarized fingerprint image. (d) Thinned fingerprint image. (e) Extraction of minutiae bifurcation points (red dots).

The Crossing Number around a pixel p in a binary image is defined as the half sum of the differences between pairs of adjacent pixels in the eight points- neighborhood of p:

$$cn(p) = \frac{1}{2} \cdot \sum_{i=1..8} |val(p_{i \bmod 8}) - val(p_{i-1})|$$

where p0, p1 .. p7 are the pixels belonging to an ordered sequence of pixels defining the eight neighborhood of p and val(p) is the pixel value. A pixel p with val(p) = 1 corresponds to bifurcation minutiae if cn(p) = 3. The ridge fingerprint image corresponds to a bit map image which has a pixel value 0 for the background points and 1 for the ridges of the fingerprint. A computed image case is given in Fig. 2(e), where the extracted minutiae (ridge bifurcations) of the template image (Fig.2(a)) are shown.

2.2 Minutiae Correspondence based on SOMs Algorithm

The proposed method to establish correspondences between minutiae of the template image and the input image is based on Competitive Learning Algorithm. The first researcher who established this method and explored its possible applications is T. Kohonen [11]. The Kohonen model can be used for a wide range of scales. In this paper a SOM model is applied to fingerprint matching.

The Kohonen model is comprised of N neuron layers, which is structured usually as a one- or two-dimensional grid. Each neuron j is assigned a vector of weights $\mathbf{w}_j \in ¡^m$. The training of the network takes place in an iterative way. At each iteration n, an input vector is fed to the network. For each neuron, the distance of its weight vector from the input vector is calculated using the following equation:

$$D_j = \|\boldsymbol{x} - \boldsymbol{w}_j\| \ (j = 1, 2, \ldots, N) \quad (1)$$

The neuron with the smallest distance is declared as "winner". Then, the update of the weight vectors of a winning neuron j and its neighboring neurons i takes place as follows:

$$\mathbf{w}_i(n+1) = \mathbf{w}_i(n) + h_{ij}(n)\left[\mathbf{x}(n) - \mathbf{w}_i(n)\right] \quad (2)$$

where $h_{ij}(n) = h\left(\|\mathbf{r}_i - \mathbf{r}_j\|, n\right)$ is a Gaussian kernel that depends on the distance distance $\|\mathbf{r}_i - \mathbf{r}_j\|$ between neurons *i* and *j* and the iteration number *n*. This kernel receives its maximum value when $\|\mathbf{r}_i - \mathbf{r}_j\| = 0$. The width of this function decreases monotonically as the iteration number increases. In this way, convergence to the global optimum is attempted during the early phases of the self-training process, whereas the convergence gradually becomes more local as the size of the kernel decreases. In particular, the kernel is as follows:

$$h_{ij}(n) = L^n e^{-\frac{\|\mathbf{r}_i - \mathbf{r}_j\|^2}{2\sigma^2(n)}} \quad (3)$$

In the above equation, the parameter L $(L \le 1.0)$ acts like a gain factor for the magnitude of the update applied to the weights of the neurons. This parameter decreases geometrically as the iteration variable n evolves. The parameter $\sigma(n)$ determines the spatial extent of the neighborhood over which the update of the neurons will take place. It's also a decreasing function of the iteration variable. In this work, $\sigma(n) = r_0 a^n$, where r_0 is the maximum distance between minutiae and $a \in (0,1)$ a real number that determines the rate of decrease of $\sigma(n)$.

The theory of the SOMs is adapted to the task of minutiae correspondence as follows: Each minutiae i $(i=1,..,N)$ of the template image with coordinates (x_i, y_i) is considered as a neuron with weight vector $\mathbf{w}_i = (r_i, dx_i, dy_i, \theta_i)$, which holds the parameters of a local similarity transformation of the square region $A_i = [x_i - R, x_i + R] \times [y_i - R, y_i + R]$ of $(2R+1)^2$ pixels. The similarity transformation is given by the following equation:

$$T_{\mathbf{w}_i}(A_i) = \left\{ \begin{array}{l} (s,t) \in ¡^2 : \\ s = r\cos\theta \cdot x - r\sin\theta \cdot y + dx, \ \forall (x,y) \in A_i \\ t = r\sin\theta \cdot x + r\cos\theta \cdot y + dy \end{array} \right\} \quad (4)$$

The objective of the algorithm is to find the best weight vector for each minutia such that:

$$SM_i(n) = \text{maximum} \quad (5)$$

where $SM_i(n) = SM\left(I_{tmp}(A_i), I_{inp}\left(T_{\mathbf{w}_i(n)}(A_i)\right)\right)$, $I_{tmp}(A_i)$ is the restriction of the template image in region A_i, $\mathbf{w}_i(n)$ is the weight vector of the minutia i at iteration n and SM is a Similarity Measure between the two images. Towards this direction, a pool of input vectors is created dynamically, as will be explained below. At each iteration n, a sample from this pool is presented to the SOMs network and the winning neuron is determined by sorting the Similarity Measure in descending order and selecting the neuron with the largest Similarity Measure. Without loss of generality, the Similarity Measure is a positive number between 0 and 1, which takes its maximum value when the two images are similar. More specifically the two images are similar then the Average Similarity Measure of all Neurons is maximum. In this work, in order to cope with the differences in contrast and/or brightness, the squared correlation coefficient between the template and the input image was the selected Similarity Measure. For two gray-level images I_1 and I_2, the specific Similarity Measure is given by the following equation:

$$SM(I_1, I_2) = \frac{\left(\sum_{x,y} \left[I_1(x,y) - \bar{I}_1 \right] \left[I_2(x,y) - \bar{I}_2 \right] \right)^2}{\sum_{x,y} \left[I_1(x,y) - \bar{I}_1 \right]^2 \sum_{x,y} \left[I_2(x,y) - \bar{I}_2 \right]^2} \quad (6)$$

where $I_1(x,y)$ and $I_2(x,y)$ are the pixel values and $\bar{I}_1$ and $\bar{I}_2$ are the mean pixel value for image I_1 and I_2, respectively.

Regarding the creation of the input vectors, the procedure is as follows:

If i^*_{n-1} is the winner with the highest Similarity Measure at the previous iteration, then a random input vector $\mathbf{s}(n) = (dx(n), dy(n), d\theta(n), r(n))$ is created according to a multidimensional Cauchy distribution with location parameter vector $\mathbf{w}_{i^*_{n-1}}$ and scale parameter γ_n which evolves with the iteration variable as follows:

$$\gamma_n = \left\{ \begin{array}{ll} \gamma_0, & n = 0 \\ \gamma_0 / \sqrt[4]{\lfloor n/p \rfloor}, & n > 0 \end{array} \right. \quad (7)$$

where γ_0 is the initial value of γ_n and the parameter p is a constant that determines the number of iterations that are executed before an adjustment of the values of the parameter γ_n takes place. A typical value for the parameter γ_0 is 2 and for p is 200. The parameter γ_n controls how far from the weights of the current winning neuron the input vector can reach. The magnitude of γ_n falls as the iteration variable evolves and the generated input signals are more localized around the weights of the current winning neuron. This is a desired property since the weights of the current winning neuron get closer to the parameters of the solution of the correspondence problem as the number of iterations grows.

Finally, the update of the weights of the neurons is performed by means of the following equation:

$$\mathbf{w}_i(n) = \mathbf{w}_i(n-1) + h_{ij}(n)[\mathbf{s}(n) - \mathbf{w}_i(n-1)] \quad (8)$$

where $h(k_{mn}, n, i)$ is defined in (3).

An index of the matching between the two images is provided by the best average value of the Similarity Measure, $SM_{best} = \max_n \left\{ \frac{1}{N} \sum_{i=1}^{N} SM_i(n) \right\}$. Since the transformed region $T_{\mathbf{w}_i}(A_i)$ does not have integer coordinates, bilinear interpolation [16] is used in order to calculate $SM_i(n)$.

It should be pointed out that in order to achieve an accurate registration result a sufficient number of minutiae should be extracted. Moreover, minutiae must be distributed over the whole template image (if possible). The degree of sparseness of the mentioned points can be determined by checking if the standard deviation of the x – y coordinates is above a predefined threshold. Experiments have shown that best results are achieved when the number of minutiae is above 12 and standard deviation of the $x-y$ coordinates exceeds 100.

3. RESULTS

In order to assess the performance of the proposed automatic fingerprint identification scheme, the database of fingerprint images used was the DB3 of FVC2004 that was used as reference database of the Fingerprint Verification Competition of 2004 . This specific database contains fingerprint images of

100 different fingers and for eight impressions of each finger; thus in total 800 images. The size of each fingerprint image is 300×480 pixels. Two different data sets were used: the first data set (SET I) comprises fingerprint images subject to known transformations and the second one (SET II) consists of fingerprint image pairs from the database (unknown transformation).

3.1 SET I: Data Subject to known Transformations

The data of SET I is comprised of 10 typical fingerprint images from the DB3 FVC2004 database. Each image from SET I is considered as a template image and is transformed according to two affine transformations (*Affine-1*, *Affine-2*) into two different input images; thus, 20 image pairs were finally obtained.

The affine transformation function is described as follows:

$$\begin{aligned} X &= s\cos\theta\left(x-x_{cm}\right)-s\sin\theta\left(y-y_{cm}\right)+t_1+x_{cm} \\ Y &= s\sin\theta\left(x-x_{cm}\right)+s\cos\theta\left(y-y_{cm}\right)+t_2+y_{cm} \end{aligned} \quad (9)$$

where $x(X)$, $y(Y)$ are the column and row indices, respectively, in the template (input) image and (x_{cm}, y_{cm}) are the coordinates of the geometric centre of the template image, s is the scaling parameter, θ the rotation angle and t_1, t_2 the translation along *x*-axis and *y*-axis, respectively. For *Affine-1*, the parameters used were: $s=1$, $\theta=5°$, $t_1=5$ and $t_2=5$, while for *Affine-2:* $s=1.05$, $\theta=10°$, $t_1=10$ and $t_2=10$.

The number of the extracted minutiae in the template images of the data SET I varied from 15 to 30 points. The actual corresponding points, $\mathbf{Q}_i$ in the input image of each affine transformed pair were calculated by transforming the points of the template image according to (9).

For the quantitative assessment of the performance of the proposed automatic point correspondence detection algorithm Root Mean Square Error (RMS) was used. The RMS was calculated between the detected points $\hat{\mathbf{Q}}_i$, using the proposed methodology, and the actual corresponding points $\mathbf{Q}_i$, $i=1,2,...,N$, according to the equation:

$$RMS=\sqrt{\frac{1}{N}\sum_{i=1}^{N}\left\|\mathbf{Q}_i-\hat{\mathbf{Q}}_i\right\|^2} \quad (10)$$

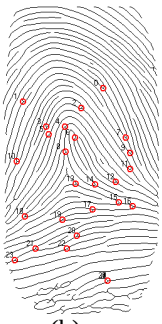
For the implementation of the algorithm described above a number of parameters must beforehand be determined. The values of the parameters used are listed in Table. 1.

Table 1: Implementation parameters of the proposed automatic fingerprint identification scheme

Equation	Description	Symbol	Value
(3)	Initial Neighbourhood Size of Winning Neuron	r_0	max distance between neurons
(3)	Rate of Change of r_0	α	0.70
(3)	Gain constant for the magnitude of the update that is applied to the weights of the neurons - Learning Rate	L	0.994
(7)	Rate of change of input vector range	p	200
	Half Size of Square Region of each neuron	R	10
	Number of Iterations	$n_{\max}$	3,000

An example of the performance of the proposed algorithm in defining automatic correspondence for typical fingerprint images of SET I is shown in Fig.3. In Fig. 3(a) a template image of the SET I is depicted along with its input images transformed by the *Affine-1* (Fig. 3(c)) and *Affine -2* (Fig. 3(e)) transformations. The red dots on the template thinned image in Fig. 3(b) correspond to the minutiae points. The actual corresponding points, calculated by the *Affine-1* (Fig. 3(d)) and *Affine-2* (Fig. 3(f)) transformations of the minutiae in the template image according to (9) are marked with red dots, while the yellow dots indicate the corresponding points detected by the SOM algorithm. From this figure, we can deduce that the detected points by the proposed algorithm correctly match the actual corresponding points.

(a)



(b)

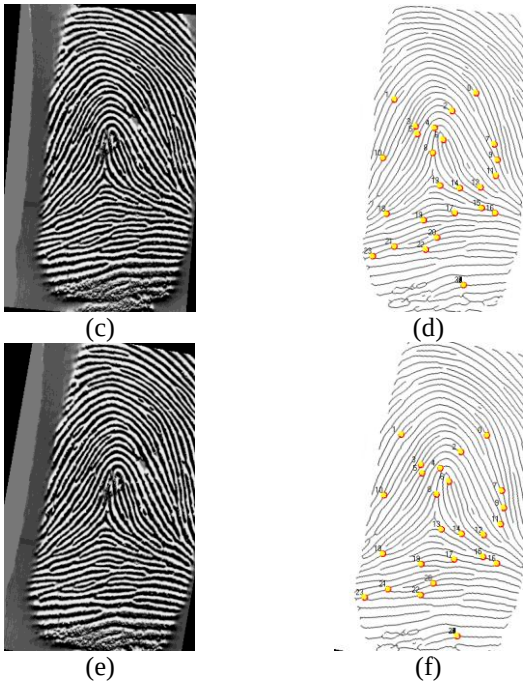


Fig. 3 The performance of the proposed SOM algorithm in defining automatic point correspondence of a typical fingerprint image of SET I and its transformed images. Red dots indicate the actual corresponding points, while yellow dots indicate the detected points using the SOM algorithm. (a) Typical fingerprint template image. (b) Minutiae of the template image. (c) Transformed template image using the *Affine-1* transformation. (d) Minutiae correspondences of the template and *Affine-1* transformed image using the SOM algorithm. (e) Transformed template image using the *Affine-2* transformation. (f) Minutiae correspondences of the template and *Affine-2* transformed image using the SOM algorithm.

The performance of the proposed SOM algorithm was tested for a typical fingerprint image of SET I and its transformed image (*Affine-1* transformation) against the number of iterations. It was observed that as the number of iteration increased, the value of the average RMSE dropped from its initial value of 10.02 pixels to the value of 5.34 pixels, after 200 iterations and to the value of 0.950 pixels after 400 iterations. The value of the Similarity Measure increased from its initial value to the value of 0.421 after 200 iterations and to the value of 0.887 after 400 iterations. The average RMSE got its best value (0.40 pixels) after 800 iterations and then it remained stable (up to 1,200 iterations), while the average Similarity Measure got its best value of 0.994 after 700 iterations.

Furthermore, the performance of the SOM algorithm in terms of the average RMS was studied for different values of the parameter *L* and the parameter *a* for the image pairs of SET I, transformed by the Affine-1 transformation. As mentioned before, the parameter *L* acts like a gain constant for the magnitude of the update that is applied to the weights of the neurons and its value was in the range between 0.99 and 1.0. The parameter *a* determines the reduction rate of the distance of the initial radius around the winning neuron and the range of its values was between 0.1 and 1.0. All the other parameters of the SOM algorithm were set to values included in Table. 1. The lowest value of the average RMSE was for $L = 0.994$ while the performance of the SOM algorithm is relative stable for different values of *L*. Similarly, the best performance of the SOM algorithm in terms of average RMSE was obtained for $a = 0.70$.

Quantitative results from the application of the proposed automatic fingerprint identification scheme to SET I are presented in terms of the best values of the Similarity Measure SM and the RMSE (in pixels) in Table. 2. For the *Affine-1* transformation, the RMS lies between 0.4 and 0.725 pixels while for the *Affine-2* transformation, the RMSE lies between 0.706 and 1.633 pixels. From Table 2, it is evident that the proposed algorithm is capable of determining correct correspondences with subpixel accuracy for the *Affine-1* transformation. However, as is expected, its performance deteriorates for *Affine-2* transformation. Yet again, the best values of the RMS for the transformations remain relatively low, while the low values of the standard deviation indicate high reproducibility of the proposed algorithm for all transformations used.

Table 2: Quantitative results obtained by the proposed automatic fingerprint identification scheme to SET I in terms of best value of SM and the RMSE (in pixels)

Pairs of SET I	Affine-1 transformation		Affine-2 transformation	
	RMSE	SM	RMSE	SM
1	0.400	0.913	1.109	0.670
2	0.456	0.916	0.778	0.821
4	0.519	0.917	0.910	0.813
5	0.545	0.865	0.706	0.682
6	0.552	0.903	1.060	0.725
7	0.632	0.883	1.145	0.634
8	0.725	0.886	1.633	0.600
9	0.490	0.935	0.930	0.711
10	0.473	0.934	0.952	0.852

Mean	**0.532**	**0.906**	**1.025**	**0.723**
STD	**0.0978**	**0.024**	**0.270**	**0.089**

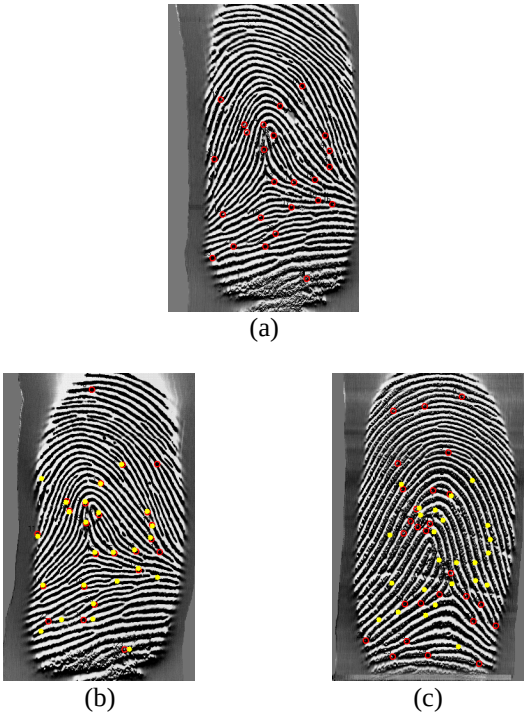
3.2 SET II: Data subject to unknown transformations

The SET II comprises of all the fingerprint images of the DB3 FVC2004. This database consists of 800 fingerprint images. Tests have been carried out on 2,800 image pairs of same fingers and on 4,950 image pairs of different fingers. Each pair consists of a template and an input image, and the transformation that associates these two images is, unlike the previous case, unknown. Furthermore it must be noted that no images are rejected on enrolment. The size of each fingerprint image is 300×480 pixels.

The proposed automatic fingerprint identification scheme was applied to all image pairs using the same values of the various parameters listed in Table 1. Since the actual corresponding points $\mathbf{Q}_i$, $i = 1, 2, ..., N$ cannot be directly calculated by transforming the minutiae of the template image, a different procedure for the visual evaluation of the proposed algorithm had to be applied. Initially, minutiae were extracted in both the template and input images, according to the aforementioned procedures. Then, new minutiae of the input image were obtained by the application of the SOM algorithm. These two sets of minutiae on the input image were visually compared.

In Fig. 4, the performance of the proposed automatic fingerprint identification scheme is visually assessed. The template image, shown in Fig.4(a) along with its extracted bifurcations, is compared with an input image of the same finger and with an input image of a different finger. In Fig. 4(b) is displayed the input image along with its extracted bifurcations (red dots) and the bifurcations (yellow dots) as obtained by the application of the proposed scheme, when the fingers compared are the same. It can be noticed that a satisfactory correspondence has been achieved for the majority of the minutiae. For this particular pair, the best value of

Similarity Measure was 0.690. In the case of applying the proposed automatic fingerprint identification scheme in pairs of different fingers, the results are considerably deteriorated, as was expected. Fig. 4(c) shows the input image of the different fingers pair along with its extracted minutiae (red dots) and the minutiae (yellow dots), as obtained by the application of the proposed scheme, where it is evident the failure of minutiae correspondence. For this particular example, the best value of Similarity Measure was 0.121.



(a) (b) (c)

Fig. 4 The performance of the proposed SOM algorithm in defining automatic point correspondence for a fingerprint image pair of SET II (same finger under different acquisition settings). (a) Initial template image. (b) The input image (same finger under different acquisition settings) along with its extracted minutiae (red dots) and the estimated minutiae as obtained by the proposed scheme (yellow dots). (c) The input image (finger of different person under different acquisition settings) along with its extracted minutiae (red dots) and the estimated minutiae as obtained by the proposed scheme (yellow dots).

The performance of the proposed scheme on this dataset was computed using the aforementioned Similarity Measure between the fingerprint image and the stored template. If the value of Similarity Measure approaches 0 (if normalized between in the range [0,1]), the more likely that both fingerprints originate from different finger. On the other hand, if Similarity Measure is near 1, it will be quite probable that both fingerprints are from same fingers.

The decision of the system is determined by a threshold *T*. In order for two fingerprints to be regarded as matched the Similarity Measure must pass the threshold, whereas if the Similarity Measure is below the threshold the fingerprints are regarded as being non-matched. The performance of a biometric system for various values of the threshold is assessed by means of the False Acceptance Rate (FAR), False Rejection Rate (FRR) and Equal Error Rate (EER):

$$FAR(T) = \frac{\text{ndiff\_match}}{ndiff\_total}$$
$$FRR(T) = \frac{nsm\_non\_match}{nsm\_total} \quad (11)$$
$$ERR = FAR(T^*)$$

where *ndiff_match* is the number of comparisons of different fingers resulting in a match, *ndiff_total* is the total number of different fingers, *nsm_match* is the number of comparisons of the same fingers resulting in a non-match, *nsm_total* is the total number of the same fingers and T^* is the value of the threshold, such that $FAR(T^*) = FRR(T^*)$.

For the calculation of the FAR and FRR the best Similarity Measure obtained by the SOM-based algorithm was calculated for the data of SET II. *ERR* corresponds to the intersection point of the two curves. The value of the ERR is 0.0586 and obtained for T^* =0.202. So the overall performance of the proposed method is 94.1%.In Fig.5 the FAR/FRR curves for various threshold values of the SM is shown, while Table 3 corresponds to the confusion matrix of the obtained results.

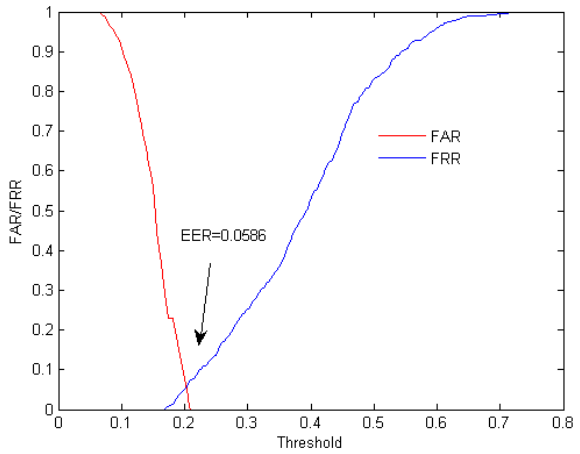


Fig. 5 The FAR/FRR curves for various threshold values of the SM. The intersection point corresponds to the ERR.

Table 3: Confusion Matrix

		Actual Value		
		Positive	**Negative**	Total
Prediction Outcome	**Positive**	2636	290	2,926
	Negative	164	4660	4,824
	Total	2,800	4,950	

Furthermore, the performance of the proposed automatic fingerprint identification scheme is above the average EER of existing methods that included in the FVC2004 [17] for the specific database DB3 and for the open category. (http://bias.csr.unibo.it/fvc2004/results/O_res_db3_a.asp).The results are shown in Table 4.

Table 4: Results of FVC2004 compared to SOMS for the database DB3

Method	EER	Method	EER
P047	1.18%	P050	5.97%
P101	1.20%	P118	6.15%
P071	1.64%	P104	6.77%
P039	1.78%	P120	6.97%
P075	1.85%	P051	7.07%
P004	1.89%	P027	7.07%
P113	2.05%	P072	7.11%
P016	2.39%	P006	7.30%
P048	2.70%	P078	7.56%
P068	3.39%	P105	8.22%
P108	3.54%	P087	8.73%
P009	3.74%	P099	9.46%
P002	3.82%	P112	9.60%
P067	3.86%	P011	10.07%
P049	4.03%	P080	11.41%
P097	4.16%	P093	11.46%
P041	4.19%	P083	11.64%
P111	4.65%	P026	14.39%
P103	5.38%	P106	19.74%
SOMS	5.86%	P079	37.92%

4. CONCLUSION

In this paper a Self Organizing Map (SOM)-based algorithm for fingerprint identification is presented. The neurons of the neural network are the minutiae of the template image and the proposed algorithm detects the set of minutiae in the input image in an iterative way. The main advantage of this method,

against other minutiae-based fingerprint recognition methods, is that the method is error-tolerant in the estimation of the minutiae of the template image. This is a matter of high importance since the precise estimation of minutiae is a difficult task, especially for low-quality fingerprint images. The overall performance of the proposed method applied on SET II was 94.1%.

Furthermore, the present report demonstrates the feasibility of applying the SOM theory to achieve the registration of fingerprint images. This approach of implementing the SOM model is to find the optimum value of an objective function. Therefore, the proposed registration scheme uses several "degrees of freedom" regarding its parameters. For example, another measure of match (such as mutual information), as well as other characteristic points of the fingerprint images (such as the points with high value of deviation), could be used. Finally, it should noticed that the proposed algorithm has not been tested on images with high degree of distortion or partial fingerprint images.

REFERENCES

[1] Maltoni D, Maio D, Jain A.K, Prabhakar S 2009 Handbook of Fingerprint Recognition Second Edition, Springer, London.

[2] Jea T.Y, Govindaraju V 2005 A minutia-based partial fingerprint recognition system. Elsevier Pattern Recognition 38:1672– 1684.

[3] Chan K.C, Moon Y.S, Cheng P.S 2004 Fast Fingerprint Verification Using Subregions of Fingerprint Images. IEEE Transactions on Circuits and Systems For Video Technology 14 : 95-101.

[4] He Y, Tian J, Li L, Chen H, Yang X 2006 Fingerprint Matching Based on Global Comprehensive Similarity. IEEE Transactions on Pattern Analysis and Machine Intelligence 28: 850-862.

[5] Gu J, Zhou J, Yang C 2006 Fingerprint Recognition by Combining Global Structure and Local Cues. IEEE Transactions on Image Processing 15: 1952-1964.

[6] Jain S.K, Chen Y, Demirkus M 2007 Pores and Ridges: High-Resolution Fingerprint Matching Using Level 3 Features. IEEE Transactions on Pattern Analysis and Machine Intelligence 29: 15-27.

[18]

[7] Chen X, Tian J, Yang X 2006 A New Algorithm for Distorted Fingerprints Matching Based on Normalized Fuzzy Similarity Measure. IEEE Transactions on Image Processing 15: 767-776.

[8] Ross A, Dass S.C, Jain A.K 2006 Fingerprint Warping Using Ridge Curve Correspondences. IEEE Transactions on Pattern Analysis and Machine Intelligence 28: 19-30.

[9] Jain S.K, Prabhakar S, Hong L, Pankanti S 2000 Filterbank-Based Fingerprint Matching. IEEE Transactions on Image Processing 9: 846-859.

[10] Liu L, Jiang T, Yang J, Zhu C 2006 Fingerprint Registration by Maximization of Mutual Information. IEEE Transactions on Image Processing 15: 1100-1110.

[11] Kohonen T 2000 Self-Organizing Maps, Third Edition, Springer-Verlag, Berlin.

[12] Tico M, Kuosmanen P 2000 An algorithm for fingerprint image postprocessing. Proceedings of the Thirty-Fourth Asilomar Conference on Signals, Systems and Computers 2:1735–1739.

[13] Hong L, Wan Y, Jain A 1998 Fingerprint Image Enhancement : Algorithm and Performance Evaluation. IEEE Transactions on Pattern Analysis and Machine Intelligence 20: 777-789.

[14] Hastings R 2007 Ridge Enhancement in Fingerprint Images Using Oriented Diffusion. Proceedings of the 9th Biennial Conference of the Australian Pattern Recognition Society on Digital Image Computing Techniques and Applications 245-252.

[15] Guo Z, Hall R 1989 Parallel Thinning with Two-Subiteration Algorithms. Communications Association of Computer Machinery, New York USA 32:359-373.

[16] Press W, Flannery B, Teukolsky S, Vetterling W 1992 Numerical Recipes in C: The Art of Scientific Computing, Second Edition, Cambridge Univ. Press, Cambridge U.K.

[17] Cappelli R, Maio D, Maltoni D, Wayman J.L, Jain A.K 2006 Performance Evaluation of Fingerprint Verification Systems. IEEE Transactions of Pattern Analysis and Machine Intelligence 28: 3-18.