

Measurement of the Principal Singular Point in Contact and Contactless Fingerprint Images by using Computational Intelligence Techniques

Ruggero Donida Labati, Angelo Genovese, Vincenzo Piuri, *IEEE, Fellow*, Fabio Scotti, *IEEE, Member*
Department of Information Technologies
Università degli Studi di Milano
Milano, 20122, Italy.
(ruggero.donida, angelo.genovese, vincenzo.piuri, fabio.scotti)@unimi.it

Abstract—Biometric systems identify individuals by comparison of the individual biometric traits, such as the fingerprint patterns. In the literature, many relevant methods are based on the localization of a reference “pivot” point of the fingerprint, called principal singular point (PSP). Most of the time, the PSP is selected from the list of the estimated singular points (SPs) that are identified by specific local patterns of the fingerprint ridges called cores and deltas. The challenge is to provide an automatic method capable to select the same PSP from different images of the same fingertip. In this paper, we propose a technique that estimates the position of all the singular points by processing the global structure of the ridges and extracting a specific set of features. The selection of the reference point from the candidate list is then obtained by processing the extracted features with computational intelligence classification techniques. Experiments show that the method is accurate and it can be applied on contact and contact-less image types.

I. INTRODUCTION

Many biometric fingerprint systems compare fingerprint images using a specific reference point of the image called Principal Singular Point (PSP). All subsequent processing steps are related to the position of the PSP, and the accuracy of its localization is a critical factor that strongly impacts on the overall accuracy of these biometric systems.

The PSP must be extracted from a list of the SPs detected on the image. The list can be obtained by different methods that study the local/global pattern of ridges of the fingerprint [1]. The real difficulty of this problem is to select the same PSP on different images produced by the same individual. At the best of our knowledge, many method has been proposed in the literature for the estimation and selection of the PSP, but there is no evidence of a preferable method. Similar method are used also for the classification of the fingerprint images [2].

The selection of the SPs is usually achieved by selecting a core point or a delta point in the fingerprint image. The core points are the center point of a loop of the ridges. The delta points are the top vertex of a triangle of ridges. The cores and deltas can be detected with different methods [3 - 5]. The estimation of the PSP is a complex task. The main problem consists in the fact that the fingerprints can present a different number of SPs. Also, some fingerprints do not present any SP.

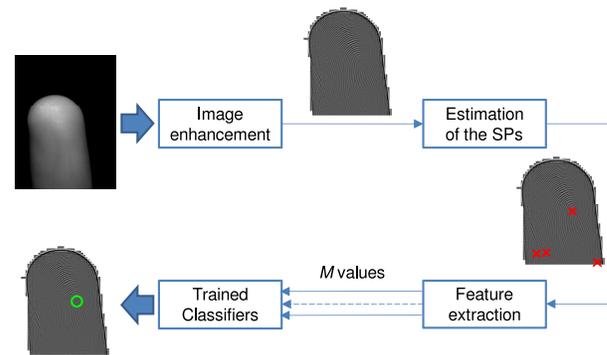


Fig. 1. Schema of the proposed method.

In literature, this problem is treated in different ways [6 - 11] but, at best of our knowledge, there are not methods for the estimation of a reference point in fingerprint images captured with touch-less technologies.

In this paper we present a method for the estimation of the PSP that can work properly with fingerprint images captured with touch-based and touch-less technologies. The proposed method can be divided in four different steps: (i) image enhancement, (ii) estimation of the SPs, (iii) feature extraction and (iv) PSP estimation. As first, the image enhancement improves the contrast of the pattern of the ridges. This step is particularly important for the images captured with touch-less technologies. Then a list of SPs and some characteristics are extracted from the fingerprint image. These data are the input of the computational intelligence techniques that estimate the position of the PSP. Fig. 1 shows the schema of the method.

This approach has been verified for touch-based and touch-less technologies fingerprint images. The obtained results are encouraging and they show that the method is feasible and it provides a relevant accuracy.

The paper is structured as follows. In the next session the related works are discussed. In section III, the proposed approach is presented and detailed. In section IV, the creation of training and test datasets is discussed, it is presented the creation of the computational intelligence techniques and the

overall results are given and compared with other techniques present in the literature. In the last section it is discussed the overall behavior of the proposed method and the future work.

II. PREVIOUS WORK

Many feature extraction algorithms for fingerprint recognition are based on a reference point (the most famous of them is the method based on the template called ‘‘Fingercode’’ [12]). In the literature, are available different methods for the estimation of a reference point in fingerprint images. The most of them are based on algorithms for the estimation of the SPs and other local features. The most important algorithm for the estimation of the SPs exploits the Poincare technique [3] that is based on the computation of the local orientation map of the ridges. Differently, in [4] is proposed a method that estimates the orientation of the ridges by an algorithm based on the Zero-Pole Model and in [5] is presented a method based on the topological structure of the fingerprint image. These algorithms can work properly with images captured with techniques based on the contact of the finger with a sensor. In [13] is proposed a method that can estimate the position of the core in touch-less fingerprint images.

Many methods for the selection of the reference point computes the local orientation of the ridges like Poincare’s method. In [6] the reference point is obtained by selecting the point in the image with the best local orientation consistency. The local orientation consistency can filter the singular points obtained by the computation of the local orientation of the ridges. In [7] the Poincare method is applied and the obtained information is filtered by computing the local orientation variance. In [10] the reference point is computed by searching the most significant points in a directional image and selecting the best point by weighting the obtained set of points with an algorithm based on the orientation of the adjacent areas.

Other methods are not based on the searching of the singularity in the map of the local orientation of the ridges. In [8] the reference point is the point in the middle of two center of gravity points obtained by computing the local axial symmetry of the image. In [11] the directional map of the ridges is quantized in four different binary images related to the four cardinal points. The reference point is obtained by applying a set of rules and morphological operations to these images. All the In [9] is proposed a solution for the estimation of the reference point for arch type fingerprints, that do not present any singular point. This class of fingerprint is uncommon. The local orientation of the ridge is computed, the radial symmetry line is estimated. The reference point consists in the point with the maximum value in the radial symmetry line.

III. THE PROPOSED APPROACH

Our approach selects the PSP from a list of SPs obtained by the Poincare method. This task is difficult because the Poincare method returns many false SPs. When there are not SPs in the fingerprint image, we select the maximum curvature point in the image. The proposed solution consists in computational

intelligence techniques capable to estimate the position of the PSP in fingerprint images that are captured with touch-based technologies and touch-less technologies.

As first, the input images are processed with an enhancement algorithm based on an analysis in the frequency domain, in order to improve the pattern structure of the ridges. This step is particularly critical for the images captured with touch-less technologies.

Then, a list of SPs is obtained by a method based on the Poincare algorithm. For the fingerprint without singular points, the method returns the list of the maximum curvature points.

The third step consists in the feature extraction. In this step, the fingerprint images are categorized with the software PCASYS [14] of the National Institute of Standard and Technologies (NIST) into the classes of (1) arch, (2) left or (3) right loop, (4) scar, (5) tented arch, and (6) whorl. The class of a fingerprint is a very important information for the selection of the PSP because it describes the global curvature of the ridges by using a single integer value. Other features consist in the position of the SPs in different coordinate systems and in a measure of the quantity of information contained in the local region of the point.

Finally, the position of the PSP is estimated by computation intelligence techniques. This methods extract the PSP from the list of the SPs by processing the input features. Fig. 2 shows an application of the methods with a touch-based fingerprint image and a touch-less image.

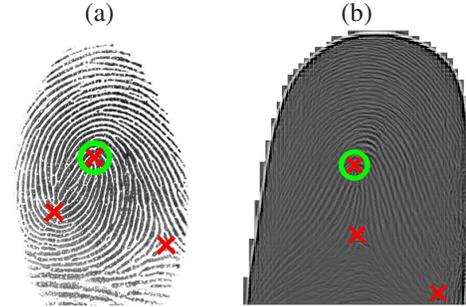


Fig. 2. Application of the proposed method on: (a) a fingerprint image captured with touch-based technologies; (b) a fingerprint image captured with touch-less technologies. The X markers represent the SPs and the O markers represent the PSPs.

A. Image enhancement

We applied a classical method for the enhancement of the fingerprint images that is described in [15]. This method computes two images that describe the local orientation O_R and the local frequency F_R of the ridges, and then it applies a convolution to the local areas of the original images with Gabor filters with the frequency and the orientation of the ridges. The even-symmetric Gabor filter has the general form

$$h(x, y : \phi, f) = \exp \left\{ -\frac{1}{2} \left[\frac{x_\phi^2}{\sigma_x^2} + \frac{y_\phi^2}{\sigma_y^2} \right] \right\} \cos(2\pi f x_\phi), \quad (1)$$

$$x_\phi = x \cos(\phi) + y \sin(\phi) \quad (2)$$

$$y_\phi = -x \cos(\phi) + y \sin(\phi) \quad (3)$$

where ϕ is the orientation of the Gabor filter, f is the frequency of a sinusoidal plane wave, and σ_x and σ_y are the space constants of the Gaussian envelope along x and y axes, respectively. The parameters ϕ and f of the Gabor filter applied to the pixel (x, y) of the image I are selected accordingly to the $O_R(x, y)$ and $F_R(x, y)$ respectively. The parameters σ_x and σ_y were empirically tuned.

However, the images captured with touch-less techniques are different from the images captured with dedicated sensors because they describe a bigger portion of the finger, the ridge-valley contrast is lower, and these images present many reflections. For this reason, we apply a segmentation method, an algorithm for removing the reflections and contrast transformations before applying the method based on Gabor filters on touch-less images.

The segmentation is based on the variance of the image. We compute a binary image I_B that represents the area of the image occupied by the finger by dividing the image I in i squared areas with size $l \times l$ and applying a threshold T_{Std} on the local standard deviation σ_i :

$$I_B(x, y) = \begin{cases} 0 & \text{if } \sigma_i(x, y) > T_{Std} \\ 1 & \text{otherwise} \end{cases} \quad (4)$$

The ROI of the images captured with dedicated sensor is equal to I_B . Considering the images captured by touchless technologies, it is necessary to reduce the ROI because the area of the finger before the last phalanx is very noisy. We consider only the region from the top of the finger y_{top} to a specific value of the y axis y_{down} . The value of y_{top} is computed as

$$y_{top} = \min_y \{y : I_B(x, y) = 0\}, \quad (5)$$

and y_{down} as

$$y_{down} = y_{top} - S_H, \quad (6)$$

where S_H is an empirically estimated constant. The ROI is computed as

$$\text{ROI} = \{I_B(x, y) \mid y \in (y_{top}, \dots, y_{down})\}. \quad (7)$$

The most important reflections are small white dots in the image. As first, the proposed algorithm estimates the areas of the reflections and then it substitutes these area in the image with interpolated values.

The first step of the estimation of the reflections consists in the computation of the gradient of the image with the Sobel operator, obtaining the image G_x and G_y :

$$G_x(x, y) = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix} * I(x, y), \quad (8)$$

$$G_y(x, y) = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} * I(x, y), \quad (9)$$

Then the image of the gradient module G_M is computed as

$$G_M(x, y) = \sqrt{G_x(x, y)^2 + G_y(x, y)^2}. \quad (10)$$

A binary image that represents a map of the reflection M_R is then obtained applying a threshold operation on G_M .

$$M_R(x, y) = \begin{cases} 0 & \text{if } G_M(x, y) > S_R \\ 1 & \text{otherwise} \end{cases} \quad (11)$$

where S_R is an empirically estimated value.

The interpolation of the intensity values of the reflection regions is computed by the algorithm proposed in [16]. For a point $M_R(x_0, y_0) > 0$, four points $\{P^{left}(x_l, y_0), P^{right}(x_r, y_0), P^{top}(x_0, y_t), P^{down}(x_0, y_d)\}$ are defined as the points with a fixed distance d in the main four directions from the borders of the reflection area that contains the point $M_R(x_0, y_0)$. The point of the final image $I_F(x_0, y_0)$ is computed as

$$I_F(x_0, y_0) = \frac{I(P^{left})(x_r - x_0) + I(P^{right})(x_0 - x_l)}{2(x_r - x_l)} + \frac{I(P^{top})(x_d - x_0) + I(P^{down})(x_0 - x_d)}{2(x_t - x_d)}. \quad (12)$$

The contrast enhancement applied to the images captured with touch-less technologies consists only in the computation of the exponential of the fingerprint image

$$I_C(x, y) = \exp(I_F(x, y)). \quad (13)$$

B. Estimation of the SPs

The following algorithm returns the list of the coordinates of the SPs and the type of each SP for the input image. It is based on the Poincare method and can be divided in the sequent steps.

- A map of the orientation of the ridges Ω is computed with the algorithm proposed in [17].
- The centroid $c_{ROI}(x, y)$ of the ROI is computed.
- The map of the Poincare indexes P of the image I is computed.
- A binary image B_C of the candidate core points is computed as:

$$B_C(x, y) = \begin{cases} 0 & \text{if } 180 - t_C < I_P(x, y) < 180 + t_C \\ 1 & \text{otherwise} \end{cases}, \quad (14)$$

where t_C is a threshold value.

- A binary image B_D of the candidate delta points is computed as:

$$B_D(x, y) = \begin{cases} 0 & \text{if } -180 - t_D < I_P(x, y) < -180 + t_D \\ 1 & \text{otherwise} \end{cases}, \quad (15)$$

where t_D is a threshold value.

- The vector of the core points V_C is computed as the centroids of the eight-connect regions in B_C .
- The vector of the core points V_D is computed as the centroids of the eight-connect regions in B_D .
- The final vector of the SPs S is obtained by sorting V_C and V_D by the distance from $c_{ROI}(x, y)$ and merging them.
- If S is empty, the coordinates of the maximum value of P nearest to $c_{ROI}(x, y)$ are putted in S .

C. Feature extraction

We computed two different features from the input images. The first of them consists in the fingerprint class obtained by the software PCASYS [14] of the National Institute of Standard and Technologies (NIST). This information describes the global structure of the ridges. The software categorizes the fingerprint images in six different classes: (1) arch, (2) left or (3) right loop, (4) scar, (5) tented arch, and (6) whorl. The method is based on a neural network classifier. We ordered the possible results in a numeric vector by similarity for reducing the effect of the errors on the final system. We also used the returned probability of the classification as a feature.

For each image, we consider the vector V_{SP} as the vector that contains all the SPs of the fingerprint. As the first step, the vector V_{SP} is sorted by type of SP (core and delta) then, the vector is sorted again by considering the proximity of the singular point from the centroid of the ROI. Only the first M SPs nearest of V_{SP} are considered in the subsequent processing steps. As a consequence, if the algorithm estimates the presence of more SPs far from the centroid, with high probability they can be considered as false positive and then discarded from further processing. For each of these points we compute the following features.

- The point coordinates (x_O, y_O) in the original image.
- The point's relative coordinates (x_R, y_R) expressed in percentage with respect the maximum dimension of the ROI.
- The polar coordinates (ρ_O, θ_O) with the origin in the centroid of the ROI.
- The angle θ_R of the SP relative to the orientation of the fingerprint α as

$$\theta_R = \theta_O - \alpha; \quad (16)$$

where α is computed as the angle from the ROI the y axis of the image.

- The mean and the standard deviation of the intensity of the pixels in the circle with radius r and center in the coordinates (x, y) of the SP. This features permit to evaluate the presence of the ridges in the area next to the SP.
- The coordinates (x_c, y_c) of the centroid of the fingerprint and the angle α .
- The index of the SP that is the nearest to the top of the finger.
- The index of the SP that is the nearest to the centroid of the ROI.

The total number of the features computed for each image is $7 + 11 \times M$. These features are then used as inputs of the computational intelligence techniques. The description of the final neural module is given in the experimental result section.

IV. EXPERIMENTAL RESULTS

In this section we describe the creation of the training and test datasets and the results obtained by training the computational intelligence techniques compared with heuristic methods.

A. Creation of the training and test datasets

We had used two different dataset of fingerprint images for the training and testing steps. The first dataset (Dataset A) contains images captured with a touch-based sensor. Dataset A is composed by 498 grayscale fingerprint images acquired by a CrossMatch V300 sensor [18, 19]. There are 8 images for each individual. These images have a resolution equal to 500 DPI with the dimension of 512×480 pixel.

The second dataset (Dataset B) contains images captured with a touch-less sensor. At best of our knowledge, there are not available any public datasets collecting touch-less fingerprint images captured in unconstrained conditions. For this reason, we created the dataset in our laboratory. The Dataset B is composed by 71 grayscale images with different fingers, captured with a Sony XCD-V90 camera. The size of each image is 1920×1024 pixels, the illumination is controlled by a led and the used focal is 25mm. The distance from the finger to the camera is about 200mm.

For each dataset we have manually estimated the PSP position of each image by selecting the position of the best SP from the list extracted by the algorithms described in section III-B. Fig. 3 shows two examples of images of the Dataset A and two examples of images of the Dataset B.

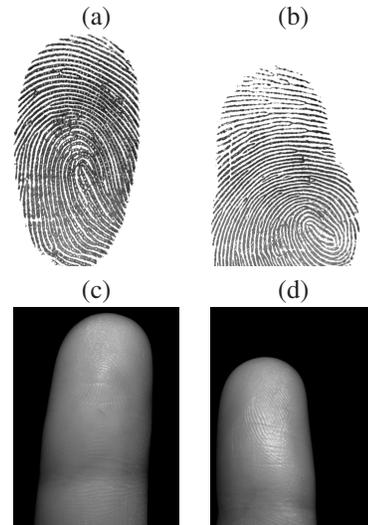


Fig. 3. Examples of images produce by contact (a,b) and contact-less (c,d) sensors used in our experiments.

B. The computational intelligence techniques and obtained results

In our experiments we tested different classification paradigms in order to better enlighten the complexity of the learning problem embedded in the datasets. In particular, we adopted the k-Nearest Neighbor classifier with odd values of the parameter k (1, 3, 5) and feed forward neural networks with different number of neurons in the hidden layer. The topology of the neural networks has been design as follows: we used a linear node for the output layer of the neural networks

TABLE I
TESTED CLASSIFIERS.

Classifier	Dataset A		Dataset B	
	Mean	Std	Mean	Std
NN-1	0.078	0.032	0.014	0.045
NN-3	0.044	0.052	0.057	0.100
NN-5	0.048	0.025	0.029	0.060
NN-10	0.048	0.025	0.157	0.171
kNN-1	0.013	0.004	0.014	0.000
kNN-3	0.016	0.004	0.028	0.000
kNN-5	0.023	0.002	0.032	0.018
kNN-10	0.028	0.002	0.056	0.000
LDC	0.03	0.002	0.032	0.007
KL-LDC	0.038	0.006	0.039	0.013
PCA-LDC	0.035	0.007	0.028	0.000
Quadratic	0.049	0.006	0.944	0.000
Heuristic 1	0.065	0.248	0.113	0.318
Heuristic 2	0.156	0.364	0.127	0.335

Notes. Classification error obtained on the different datasets with the methods: Feed-Forward Neural Network with one hidden layer composed by 1 node (NN-1), 3 nodes (NN-3), 5 nodes (NN-5), 10 nodes (NN-10); k Nearest Neighbor with $k = 1$ (kNN-1), $k = 3$ (kNN-3), $k = 5$ (kNN-5), $k = 10$ (kNN-10); Normal densities based linear classifier (Ldc); Linear classifier by KL expansion of common covariance matrix (KL-LDC); Linear classifier by PCA expansion on the joint data (PCA-LDC); Quadratic classifier (Quadratic); Heuristic 1; Heuristic 2.

and we tested different configurations for the hidden layer. In particular, we have tested one or two levels of nodes with different topologies: log-sigmoidal and tan-sigmoidal. The algorithm used for the train of the neural networks is the back-propagation algorithm. Other classical inductive classification system has been considered in our tests such as the Fisher linear discriminant classifier and quadratic classifiers with different preprocessing methods. As reference, we used also two heuristic methods proposed in the literature: the method Heuristic 1 is based on the selection the PSP from the available SPs list by the estimation of the point that is the nearest from the top of the finger; the method Heuristic 2 considers the SP that is nearest to the centroid of the ROI as the PSP.

In order to train the computational intelligence methods, we adopted a simple N -fold cross validation technique with $N = 10$ [20]. Table I reports the classification accuracy of tested classifiers and the relative standard deviation over the Dataset A and Dataset B.

Results show that the proposed method offers a remarkable accuracy compared with the reference methods in all datasets. In fact, for the tested datasets, the usage of the trained classifiers produced a better classification accuracy than the heuristic-based classification methods (last two rows in Table I).

In addition, in the classification of the Dataset B, the neural-based classifier shows a very good accuracy compared with other classical inductive classification system.

The only classification family showing a similar accuracy (on the considered datasets) are the kNN classifiers, but the neural network approach offers a relevant gain in the computational complexity (up to a factor 10 in the classifier execution time in our datasets). Experiments showed that the Method A applied with the neural-based quality classification system is the most suitable for the real-time applications.

The presented methods had been tested on contact and contact-less image databases. The images captured from a

multiple view system are comparable to the images discussed in this paper. For each view produced by a N -view system, the coordinates of the PSP point can be processed by using the proposed approach. The PSP $p_i = (x_i, y_i)$ extracted from in the I_i image can be collected in the final coordinate vector $C = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_M)\}$. From the vector C , it is possible to obtain a three-dimensional projection of the PSP $P(x_p, y_p, z_p)$. In the case of a 2-view system, this representation can be computed by applying the pinhole camera model [21]:

$$p_i = \begin{pmatrix} x_i \\ y_i \\ 1 \end{pmatrix} = K \begin{bmatrix} 1000 \\ 0100 \\ 0010 \end{bmatrix} \begin{bmatrix} R & t \\ 0^\top & 1 \end{bmatrix} \begin{pmatrix} x_p \\ y_p \\ z_p \\ 1 \end{pmatrix} = K [R|t] P, \quad (17)$$

where K is the camera calibration matrix, R the rotation matrix, and t the translation vector.

We tested a similar approach in images captured by 2-view system and our preliminary results are encouraging. Fig. 4 shows an example of the obtained results.

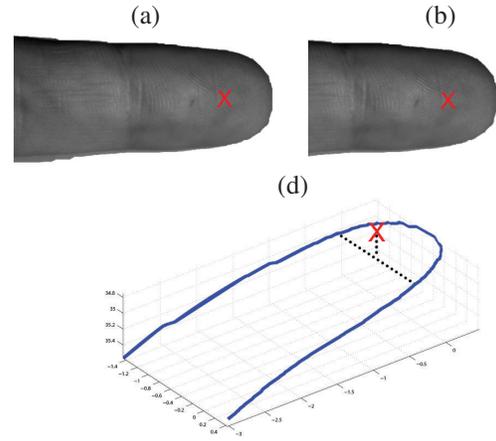


Fig. 4. Estimation of the PSP in stereo vision system; (a) left image; (b) right image; (c) PSP in the three-dimensional space.

In future work we will study the performances varying operational conditions, with different strategies for the feature selection, optimizing the method in the different conditions, as well as the estimation of the obtainable accuracy with enlarged dataset size.

V. CONCLUSION

The paper presented an innovative method based on computational intelligence techniques capable to deal with the problem of Principal Singular Point selection in fingerprint images. The method exploits a specific set of features mapping the global ridge pattern of the fingerprint and a trained classifier capable to select from the list of candidate singular points the Principal Singular Point. The proposed method is general and it can deal with contact and contactless images types. Further studies will be focused on the optimization of the computational complexity of the method as well as on the study of the effect of different environmental conditions on enlarged datasets.

ACKNOWLEDGMENT

This work was supported in part by the MIUR (Ministero dell'Università e della Ricerca) under Grant 2007JXH7ET (Priv-Ware, Privacy aware processing of encrypted signals for treating sensitive information).

REFERENCES

- [1] G. Drets and H. Liljenstrom, "Fingerprint sub-classification and singular point detection," *PRAI*, vol. 12, no. 4, pp. 407–422, June 1998.
- [2] Q. Zhang and H. Yan, "Fingerprint classification based on extraction and analysis of singularities and pseudo ridges," *Pattern Recognition*, vol. 37, no. 11, pp. 2233 – 2243, 2004.
- [3] A. K. Jain and D. Maltoni, *Handbook of Fingerprint Recognition*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2003.
- [4] L. Fan, S. Wang, H. Wang, and T. Guo, "Singular points detection based on zero-pole model in fingerprint images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, pp. 929–940, 2008.
- [5] J. Zhou, J. Gu, and D. Zhang, "Singular points analysis in fingerprints based on topological structure and orientation field," 2007, pp. 261–270.
- [6] G. Eason, B. Noble, and I. N. Sneddon, "On Certain Integrals of Lipschitz-Hankel Type Involving Products of Bessel Functions," *Royal Society of London Philosophical Transactions Series A*, vol. 247, pp. 529–551, Apr. 1955.
- [7] J. C. Maxwell, *A treatise on electricity and magnetism / by James Clerk Maxwell*, unabridged 3rd ed. ed. Dover Publications, New York :, 1954.
- [8] T. Liu, C. Zhang, and P. Hao, "Fingerprint reference point detection based on local axial symmetry," in *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*. Washington, DC, USA: IEEE Computer Society, 2006, pp. 1050–1053.
- [9] H. Lam, Z. Hou, W. Yau, T. Chen, J. Li, and K. Sim, "Reference point detection for arch type fingerprints," 2009, pp. 666–674.
- [10] P. Porwik and K. Wrobel, "The new algorithm of fingerprint reference point location based on identification masks," in *4th Int. Conf. on Comp. Recognition Systems CORES'05*, 2005, pp. 807–814.
- [11] J. L. A. Samatelo and E. O. T. Salles, "Determination of the reference point of a fingerprint based on multiple levels of representation," in *SIBGRAPI '09: Proceedings of the 2009 XXII Brazilian Symposium on Computer Graphics and Image Processing*, 2009, pp. 209–215.
- [12] A. K. Jain, S. Prabhakar, L. Hong, and S. Pankanti, "Filterbank-based fingerprint matching," *IEEE Transactions on Image Processing*, vol. 9, pp. 846–859, 2000.
- [13] B. Hiew, A. Teoh, and D. Ngo, "Automatic digital camera based fingerprint image preprocessing," in *Computer Graphics, Imaging and Visualisation, 2006 International Conference on*, 26-28 2006, pp. 182 –189.
- [14] C. I. Watson, M. D. Garris, E. Tabassi, C. L. Wilson, R. M. McCabe, S. Janet, and K. Ko, "User's guide to nist biometric image software (nbis)," 2007.
- [15] L. Hong, Y. Wan, and A. Jain, "Fingerprint image enhancement: algorithm and performance evaluation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 20, no. 8, aug 1998.
- [16] Z. He, T. Tan, Z. Sun, and X. Qiu, "Toward accurate and fast iris segmentation for iris biometrics," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 31, no. 9, pp. 1670 –1684, sept. 2009.
- [17] L. Hong, Y. Wan, and A. Jain, "Fingerprint image enhancement: Algorithm and performance evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 777–789, 1998.
- [18] Neurotechnology, <http://www.neurotechnology.com>.
- [19] CROSSMATCH Technologies, Verifier 300, <http://www.neurotechnology.com>.
- [20] R. Duda, P. Hart, and D. Stork, *Pattern Classification*. New York: John Wiley & Sons, 2001.
- [21] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.