A new fingerprint matching approach using level 2 and level 3 features
Bahram Sadeghi Bigham, Samaneh Mazaheri**
Department of Computer Science & Information Technology,
Institute for Advanced Studies in Basic Science (IASBS), Zanjan, Iran

Abstract. Fingerprint recognition is considered as one of the most popular and reliable techniques for automatic identification due to the well-known distinctiveness and persistence of fingerprints. In an automated fingerprint recognition system (AFRS), fingerprint matching is an important and essential step. The distortion and noise of given fingerprints and the inaccurate feature extraction make fingerprint matching a very difficult problem. With the advent of high-resolution fingerprint imaging techniques and the increasing demand for high security, sweat pores have been recently attracting increasing attention in automatic fingerprint recognition. Therefore, this paper takes fingerprint pore matching as an example to show the robustness of our proposed matching method to the errors due to fingerprint representation. This method using Onion Peeling algorithm of computational geometry, utilizes Level 3 features in conjunction with Level 2 features for matching. The experimental results on two databases of high-resolution fingerprints demonstrate that the proposed method can achieve much higher recognition accuracy compared with other state-of-the-art pore matching methods. The algorithm has been tested on two databases (NIST 9 database and FVC2006). We have measured the performance of our proposed algorithm in terms of receiver operating characteristics (ROC). For the NIST9 database, at ~11% false acceptance rate (FAR) the genuine acceptance rate (GAR) is ~ 92% and at ~0% FAR, observed GAR is ~74%. For the FVC2006 database at ~0% FAR the GAR observed is 71% and at ~19% FAR, GAR is 89%. It is concluded that with proposed technique we can have better GAR at low FAR with reduced computational complexity.

Keywords: Fingerprint Recognition, Fingerprint Matching, Pores, Level 3 Features, Onion Peeling Algorithm, Computational Geometry, and Nested Convex Polygons.

1 Introduction
Fingerprint recognition is a widely popular but a complex pattern recognition Problem. It is difficult to design accurate algorithms capable of extracting salient features and matching them in a robust way. There are two main applications involving fingerprints: fingerprint verification and fingerprint identification. While the goal of fingerprint verification is to verify the identity of a person, the goal of fingerprint identification is to establish the identity of a person. Specifically, fingerprint identification involves matching a query fingerprint against a fingerprint database to establish the identity for an individual. To reduce search time and computational complexity, fingerprint classification is usually employed to reduce the search space by splitting the database into smaller parts (fingerprint classes) [1]. There is a popular misconception that automatic fingerprint recognition is a fully solved problem since it was one of the first applications of machine pattern recognition (it dates back to more than 50
years ago). On the contrary, fingerprint recognition is still a challenging and important pattern recognition problem. The real challenge is matching fingerprints affected by:

- High displacement or rotation which results in smaller overlap between template and query fingerprints (this case can be treated as similar to matching partial fingerprints),
- Non-linear distortion caused by the finger plasticity,
- Different pressure and skin condition and
- Feature extraction errors which may result in spurious or missing features.

The approaches to fingerprint matching can be coarsely classified into three classes: Correlation-based matching, Minutiae-based matching and Ridge-feature-based matching. In correlation-based matching, two fingerprint images are superimposed and the correlation between corresponding pixels is computed for different alignments. During minutiae-based matching, the set of minutiae are extracted from the two fingerprints and stored as a sets of points in the two dimensional plane. Ridge-feature-based matching is based on such feature as orientation map, ridge lines and ridge geometry.

![Fingerprint features](image)

The information contained in a fingerprint can be categorized into three different Levels, namely, Level 1 (pattern), Level 2 (minutiae points), and Level 3 (pores and ridge contours).

The vast majority of contemporary automated fingerprint authentication systems (AFAS\footnote{1}) are minutiae (Level 2 features) based\footnote{2}. Minutiae-based systems generally rely on finding correspondences\footnote{2} between the minutiae points present in “query” and “reference” fingerprint images. These systems normally perform well with high-quality fingerprint images and a sufficient fingerprint surface area. These conditions, however, may not always be attainable.

---

1 Automated Fingerprint Authentication Systems
2 A minutiae in the "query" fingerprint and a minutiae in the "reference" fingerprint are said to be corresponding if they represent the identical minutiae scanned from the same finger
Figure 2: Example of skin distortions.

As shown in Figure 2, even though these two fingerprint images are from the same individual, the relative positions of the minutiae are very different due to skin distortions. This distortion is an inevitable problem since it is usually associated with several parameters [28], [29], including skin elasticity, nonuniform pressure applied by the subject, different finger placement with the sensor, etc.

In many cases, only a small portion of the “query” fingerprint can be compared with the “reference” fingerprints as a result, the number of minutiae correspondences might significantly decrease and the matching algorithm would not be able to make a decision with high certainty. This effect is even more marked on intrinsically poor quality fingerprints, where only a subset of the minutiae can be extracted and used with sufficient reliability. Although minutiae may carry most of the fingerprint’s discriminatory information, they do not always constitute the best trade-off between accuracy and robustness. This has led the designers of fingerprint recognition techniques to search for other fingerprint distinguishing features, beyond minutiae which may be used in conjunction with minutiae (and not as an alternative) to increase the system accuracy and robustness. It is a known fact that the presence of Level 3 features in fingerprints provides minutiae detail for matching and the potential for increased accuracy.

Ray et al. [5] have presented a means of modeling and extracting pores (which are considered as highly distinctive Level 3 features) from 500 ppi fingerprint images. This study showed that while not every fingerprint image obtained with a 500 ppi scanner has evident pores, a substantial number of them do have. Thus, it is a natural step to try to extract Level 3 information, and use them in conjunction with minutiae to achieve robust matching decisions. In addition, the fine details of Level 3 features could potentially be exploited in circumstances that require high-confidence matches. Moosavi [21] developed a matcher that utilizes Level 3 features, including pores and ridge contours, in conjunction with Level 2 features (minutiae) for matching.

The types of information that can be collected from a fingerprint’s friction ridge impression can be categorized as Level 1, Level 2, or Level 3 features as shown in Figure 1. At the global Level, the fingerprint pattern exhibits one or more regions where the ridge lines assume distinctive shapes characterized by high curvature, frequent termination, etc. These regions are broadly classified into arch, loop, and whorl. The arch, loop, and whorl can further be classified into various subcategories, by noticing Delta and core.

Features of Level 1 comprise these global patterns and morphological information. They alone do not contain sufficient information to uniquely identify fingerprints but are used for broad classification of fingerprints.
Level 2 features or minutiae refer to the various ways that the ridges can be discontinuous. These are essentially Galton characteristics, namely ridge endings and ridge bifurcations. A ridge ending is defined as the ridge point where a ridge ends abruptly. A bifurcation is defined as the ridge point where a ridge bifurcates into two ridges.

![Figure 3: Singular Points (Core & Delta) and Minutiae (ridge ending & ridge bifurcation)](image-url)

Minutiae are the most prominent features, generally stable and robust to fingerprint impression conditions. Statistical analysis has shown that Level 2 features have sufficient discriminating power to establish the individuality of fingerprints [6]. Level 3 features are the extremely fine intra ridge details present in fingerprints [7]. These are essentially the sweat pores and ridge contours. Pores are the openings of the sweat glands and they are distributed along the ridges.

Studies [8] and [12] have shown that density of pores on a ridge varies from 23 to 45 pores per inch and 20 to 40 pores should be sufficient to determine the identity of an individual. A pore can be either open or closed, based on its perspiration activity. A closed pore is entirely enclosed by a ridge, while an open pore intersects with the valley lying between two ridges as shown in Figure 4.

The pore information (position, number and shape) are considered to be permanent, immutable and highly distinctive but very few automatic matching techniques use pores since their reliable extraction requires high resolution and good quality fingerprint images. Ridge contours contain valuable Level 3 information including ridge width and edge shape. Various shapes on the friction ridge edges can be classified into eight categories, namely, straight, convex, peak, table, pocket, concave, angle, and others as shown in Figure 5. The shapes and relative position of ridge edges are considered as permanent and unique.

![Figure 4: Open and closed pores](image-url)

![Figure 5: Characteristics of ridge contours and edges](image-url)
On the perpetual quest for perfection, a number of techniques devised for reducing FAR\textsuperscript{3} and FRR\textsuperscript{4} were developed; computational geometry being one of such techniques [10]. Matching is usually based on lower-Level features determined by singularities in the finger ridge pattern known as minutiae. Given the minutiae representation of fingerprints, fingerprint matching can simply be seen as a point matching problem. As mentioned before, two kinds of minutiae are adopted in matching: ridge ending and ridge bifurcation. For each minutia usually extract three features: type, the coordinates and the orientation.

![Figure 6: Two types of minutiae, ridge ending and ridge bifurcation with their orientations](image)

Where \( \theta \) is the orientation and \((x_0, y_0)\) is the coordinate of minutiae. M. Poulos et al. develop an approach that constructs nested polygons based on pixels brightness; this method needs some image processing techniques [11].

Another geometric topologic structure, Nested Convex Polygons (NCP) [13], used in [17], Khazaee and others establish a matching using minutiae. This approach was invariant from translation and rotation. They also had a local matching with using of the most interior polygon (Reference Polygon) and then apply global matching. They use reference polygon that is unique for every fingerprint; this uniqueness helps to reject non matching fingerprints with minimum process and time. We use in our approach, this point of view and continue this idea to use it for pores and ridges in Level 3 features. In this paper, we proposed a new fingerprint matching method that utilizes Level 3 features (pores and ridge contour) in conjunction with Level 2 features (minutiae) for matching, using of the most interior polygon (reference polygon) and apply matching in 2 Levels. Three main steps of our proposed method are:

1) Minutiae extraction and matching in Level 2
2) Pores extraction and matching in Level 3
3) Fingerprint recognition

\textsuperscript{3} False Acceptance Rate
\textsuperscript{4} False Rejection Rate
In Section 2, we review some feature extraction methods. In Section 3, we introduce nested convex polygons [13]. In section 4, we describe matching approach. Then, in section 5, show how we can construct a NCP with minutiae and pores. A new approach for fingerprint matching using NCP is described in section 6. Experimental results on FVC2006 and NIST 9 are presented in Section 7. The paper is concluded in Section 8.

2 Feature Extraction

To compensate for the variations in lighting, contrast and other inconsistencies three preprocessing steps are used: Gaussian blur, sliding window contrast adjustment, and histogram based intensity Level correction. Gaussian blurring is used to remove any noise introduced by the sensor. The lighting inconsistencies are adjusted by using sliding-window contrast adjustment on the Gaussian blurred image. To further enhance the ridges and valley a final intensity correction is made by using Histogram-based Intensity Level Adjustment.

The preprocessed image is enhanced using a popular fingerprint enhancement technique by Sharat [14] which uses contextual filtering in Fourier domain. The enhanced fingerprint image is not suitable for extracting pores as the pore information is lost during enhancement. Thus for pore extraction the preprocessed image is used.

The minutiae points are extracted from the enhanced image by using NIST’s5 minutia extraction software (NFIS). The method first generates the image quality maps by checking the regions with high curvature, low flow and low contrast. And then, a binary representation of the fingerprint is constructed. Minutiae are generated by comparing each pixel neighborhood with a family of minutiae templates. Finally, spurious minutiae are removed by using a set of heuristic rules. The NFIS also counts the neighbor ridges and assigns each minutia point a quality (in the range 0 to 100) determined from image quality maps. The minutia representation generated by NFIS consists of the location information, orientation, minutia type (bifurcation or ending) and minutia quality [26]. The proposed minutia matcher does not differentiate between different minutiae types. This is because the minutia types are difficult to distinguish when the applied finger pressure during acquisition varies. Figure 9

---

5NIST stands for National Institute of Standards and Technology and is a federal technology agency that develops and promotes measurement, standards, and technology.
shows one such example where the same minutiae extracted from two different impressions appears as a bifurcation in one image and as a ridge ending in other.

![Image](image1.png)

(a)                                (b)

Figure 9: Effect of differential finger pressure [16]

The Level 3 features are extracted only when the minutiae based matching fails to match the query fingerprint image with the template image. The ridge contours are extracted from the enhanced fingerprint image by using the approach proposed by Jain et al. [9]. The extracted ridge contours by using this approach is shown in Figure 10.

![Image](image2.png)

(a) Sample 500 dpi fingerprint (b) Enhanced fingerprint (c) Traced Ridge Contours fingerprint

Figure 10: Ridge Contour extraction process in an image scanned with Cross Match Verifier 300 scanner at 500 dpi

For extracting pores the technique proposed in [16] and [25] by International Biometric Group is used. Figure 11 show the extracted pores by using this approach. The pore information thus extracted contains the location information and orientation information of pores.

![Image](image3.png)

(a) Sample 500 dpi fingerprint  (b) extracted pore

Figure 11: Pores extracted from an image scanned with Cross Match Verifier 300 scanner at 500 dpi

2.1 Fingerprint Preprocessing
Before extracting the proposed ridge features, we need to perform some preprocessing steps (see Figure 12). These steps include typical feature extraction procedures as well as additional procedures for quality estimation and circular variance estimation. We first divide the image into 8x8 pixel blocks. Then, the mean and variance values of each block are calculated to segment the fingerprint regions in the image. We then apply the method described in [29] to estimate the ridge orientation and the ridge frequency is calculated using the method presented in [30]. The Gabor filter [31] is applied to enhance the image and obtain a skeletonized ridge image. Then, the minutiae (end points and bifurcations) are detected in the skeletonized image. The quality estimation procedure [32] is performed in order to avoid extracting false minutiae from poor quality regions and to enhance the confidence level of the extracted minutiae set.

![Figure 12: Overall preprocessing steps](image)

Furthermore, in regions where ridge flows change rapidly, such as the area around a singular point, it is hard to estimate the ridge orientations accurately or to extract the thinned ridge patterns consistently. Therefore, to detect regions which have large curvature, we apply circular variance estimation [33]. The circular variance of the ridge flows in a given block is calculated as follows:

\[
Var(\theta) = 1 - \frac{1}{n} \left[ \left( \sum_{i=1}^{n} \cos \theta_i \right)^2 + \left( \sum_{i=1}^{n} \sin \theta_i \right)^2 \right]
\]

Where \( \theta_i \) and \( n \) represent the estimated orientation of the \( i \)th block and the number of neighboring blocks around the \( i \)th block, respectively. In our experiments, we use eight neighboring blocks. Quality estimation and circular variance values are used to avoid generating feature vectors in poor quality regions or in regions around singular points. Moreover, we adopt some post-processing steps [34] to remove falsely extracted ridges, such as short ridges and bridges. We can then extract the ridge structures consistently against various noise sources.

### 3 Nested Convex Polygons

Let \( S = \{x_1, x_2, \ldots, x_n\} \) denote \( n \) points in two dimensional spaces. We use Quick Hall algorithm iteratively to construct nested polygons [13].
**Algorithm 1: Construct Nested Convex Polygons (CNCP)**

**Input:** $S = \{x_1, x_2, ..., x_n\}$, $K = \{\}$, $depth = 0$

**Output:** Nested Convex polygons with their depth

```
While ($N(S)$ > 0)
{
    $K = \{\}$;
    $K = quickhall(S)$;
    $S = S - K$;
    StorePolygonProperties($K$, depth);
    depth++;
}
```

Where $N(S)$ is the number of minutia in $S$, *quickhall()* method find the convex layer of given point set and *Store Polygon Properties* is a method that stores the reference polygon properties and its depth finally.

![Nested polygons constructed from point set](image)

**Figure 13:** Nested polygons constructed from point set

### 4 Matching Phase

The purpose of fingerprint matching is to determine whether two fingerprints are from the same finger or not. In order to this, the input fingerprint needs to align with the template fingerprint represented by its minutiae pattern [18]. The following rigid transformation can be performed:

$$
F_{S, \Delta \theta, \Delta x, \Delta y} \begin{pmatrix} x_{temp} \\ y_{temp} \end{pmatrix} = \begin{pmatrix} \cos \Delta \theta & -\sin \Delta \theta \\ \sin \Delta \theta & \cos \Delta \theta \end{pmatrix} \begin{pmatrix} x_{input} \\ y_{input} \end{pmatrix} + \begin{pmatrix} \Delta x \\ \Delta y \end{pmatrix}
$$

Where $(s, \Delta \theta, \Delta x, \Delta y)$ represent a set of rigid transformation parameters: (scale, rotation, translation). In our research, we assume that the scaling factor between input and template image is identical since both images are captured with the same device.
Suppose $Q=((x_1^Q, y_1^Q, \theta_1^Q, t_1^Q), \ldots, (x_n^Q, y_n^Q, \theta_n^Q, t_n^Q))$ denote the set of $n$ minutiae in the input image ($(x, y)$: location of minutiae, $\Theta$: orientation field of minutiae, $t$: minutiae type, end or bifurcation);

And $P=((x_1^P, y_1^P, \theta_1^P, t_1^P), \ldots, (x_n^P, y_n^P, \theta_n^P, t_n^P))$ indicate set of $m$ minutiae in template image.

5 Constructed Nested Polygons

Table 1. Fingerprint image comparison in Level two. Y: Fingerprint transformation dependent, N: independent of fingerprint transformation.

| Feature                        | $X$: (Y) | $Y$: (Y) | $\theta$: (Y) | Type: (N) |
|-------------------------------|---------|---------|-------------|----------|
| Minutiae Point                |         |         |             |          |
| Polygon Edges                 | Length: (N) | $\theta_1$: (N) | $\theta_2$: (N) | Type1: (N) / Type2: (N) |

Table 2. Fingerprint image comparison in Level three. Y: Fingerprint transformation dependent, N: independent of fingerprint transformation.

| Feature                        | $X$: (Y) | $Y$: (Y) | $\theta$: (Y) | Type: (N) |
|-------------------------------|---------|---------|-------------|----------|
| Pore Point                    |         |         |             |          |
| Polygon Edges                 | Length: (N) | $\theta_1$: (N) | $\theta_2$: (N) | Type1: (N) / Type2: (N) |
Table 1 shows features that we use in Level 2 matching and table 2 shows the features we use in Level 3 matching. In the table 1, Length is the length of edge; $\Theta_1$ is the angle between the edge and the orientation field at the first minutiae point; Type1 denote minutiae type of the first minutiae [10].

Using onion layer algorithm we construct nested polygons; for every fingerprint we store edge properties that mentioned in table 1 of the reference polygon, plus its depth, and Minutiae points features that mentioned in first row of table 1 in database as template (fingerprint Registration). Also, we construct nested polygons for pores and for every fingerprint store edge properties that mentioned in table 2 of the reference polygon, plus its depth, and pore points features that mentioned in first row of table 2 in database as template (fingerprint Registration).

At the Figure 14 the polygon at depth 6 is the reference polygon that used for Level 2 matching in order to calculate rigid transformation parameters; these parameters apply to the whole remain minutiae of input fingerprint in order to align with template fingerprint, then Level 3 matching is employed, and if the score of matching is higher than predefined threshold, two fingerprints are identical, otherwise they are from different fingers.

![Fingerprint with minutiae and its nested polygons](image)

Figure 15: Fingerprint with minutiae and its nested polygons

6 Registration Phase Based on Nested Convex Polygons

The purpose of fingerprint matching is to determine whether two fingerprints are from the same finger or not. At this step input fingerprint image goes through preprocessing, NCPs construction and determining its class like registration step. Afterward depend on Identification (1→n matching) or verification (1→1 matching) we perform matching. In verification mode we do not have to determine the class of fingerprint; retrieve fingerprint from database template and perform matching. But the purpose of identification is to identify unknown person, so in this mode, the class of input fingerprint is detect and matching of unknown fingerprint with templates at that class continue until happen a matching or rejection at the end. We divided registration in two steps: firstly, in preprocessing step, some image processing techniques that customize for fingerprint, like segmentation, normalization, Gabor filter and etc apply on fingerprint input image [19]. Next, Binarization and thinning are employed and valid minutiae are extracted from thin image.
Secondly, we apply onion layer algorithm and construct NCPs. We store invariant feature (table 1) for reference polygon plus its depth and variant features for other polygons in the database as a template. We also do the same procedure for pores and apply onion layer algorithm and construct NCPs for them too and store invariant feature (table 2) for reference polygon plus its depth and variant features for other polygons in the database as a template. Following steps elaborate our algorithm in identification mode.

Some abbreviations that we use in algorithm:

- **RP**: Reference Polygon
- **RT**: Reference Triangle
- **P_i**: RP of input fingerprint image
- **P_t**: RP of template fingerprint image
- **\( \beta_i \)**: angle between two adjacent edges of RP in RT of input fingerprint image
- **\( \beta_t \)**: angle between two adjacent edges of RP in RT of template fingerprint image
- Other abbreviations are interpreted base on table 1 and table 2.

**Algorithm 2:**
1. Compare input and template RP depth:
   \[
   D = |\text{Dept} (P_t) - \text{Dept} (P_i)|
   \] (1)
2. If \( D \geq 2 \), then two fingerprints are not from same finger, so matching reject at this step.
3. Otherwise select one of the \( P_i \) edges \( (E_i) \) and find corresponding edge in \( P_t \); two edges are corresponding (equal) if satisfy four following conditions:
   \[
   |\text{Len}(E_i) - \text{Len}(E_t)| < 1 \] (2)
\[ Type_1 (E_i) = Type_1 (E_i) \] (3)

\[ Type_2 (E_i) = Type_2 (E_i) \] (4)

\[ \theta_1 - \theta_1 - T_2 \] And \[ \theta_2 - \theta_2 - T_2 \] (5)

Where, \( T_1 \) and \( T_2 \) are respectively thresholds of length of edges, and minutia angle and orientation of edge, respectively.

4. Repeat step 3, until find two adjacent edges in \( P_i \) that have two adjacent edges corresponding in \( \hat{P}_i \). If such couple adjacent edges don’t exist in two RPs, matching reject at this step.

5. Using of such couple adjacent edges, a triangle constructed as Reference Triangle (RT). One more step needed to ensure that two triangles are exactly corresponded, that’s satisfied with following condition:

\[ |\beta_i - \beta_i| - T_3 \] (6)

Where, \( T_3 \) is threshold of angle between two adjacent edges in two RT.

6. Using of RT, calculate the rigid transformation parameters \( (\Delta \theta, \Delta x, \Delta y) \).

7. Achieve the number of corresponding minutiae pairs. Given an alignment \( (\Delta \theta, \Delta x, \Delta y) \), minutiae of input fingerprint are mapped into template fingerprint. Judging they are matched or not according to equations (7);

\[ m = \{ \text{yes} \text{, if } \sqrt{(x_i - x_{temp})^2 + (y_i - y_{temp})^2} \leq r_0 \text{ and } Type_i = Type_i \text{ and } |\theta_i - \theta_i| < \theta_0 \}
\]

\[ \text{no, otherwise} \] (7)

Where, \( r_0 \) and \( \theta_0 \) are thresholds of distance and orientation respectively.

8. Similarity calculates according to equation (8);

\[ p = \frac{2n}{m + q} \times 100 \] (8)

Where, \( m \) and \( q \) are the number of minutiae in two fingerprints and \( n \) is the number of matched minutiae. If \( p \) be greater than predefined value, so two fingerprint are the same, otherwise go back to step 3. This iteration continues until either no candidate at step 4 exists, or accepts at step 9.

7 Experimental Results

We perform experiments using the fingerprint database of FVC 2006 and NIST 9 database to evaluate the correctness of algorithm and show the results of experiments. The experiment uses DB1_a in database FVC 2006 [20]. Each database contained 800 fingerprints from 100 different fingers and in each database dry, wet, scratched, distorted and markedly rotated fingerprints were also adequately represented. The performance of the algorithm is measured in terms of receiver operating characteristics (ROC). For the NIST 9 database, at ~ 0% false acceptance rate (FAR) the genuine acceptance rate (GAR) observed is ~ 74% and at ~ 11% FAR, GAR is ~ 92%. For the FVC2006 database at ~ 0% FAR the GAR observed is 71% and at ~ 18% FAR, GAR is 89%. Proposed algorithm yields better GAR at low FAR with reduced computational complexity.
In this paper, we have designed a completely new method of fingerprint matching using an onion layer technique regarding computational geometry. This matching approach works by using Level 3 characteristics (pores along with ridge contour) jointly with Level 2 characteristics (minutiae) for matching. Using an onion layer technique, we assemble nested convex polygons regarding minutiae, after determining by polygons property, we perform matching regarding fingerprints; we use the almost all inside polygon in order to analyze the rigorous transformations parameters along with perform Level 2 matching, next we employ Levels 3 matching. The theory research regarding computational complexity demonstrates the NCP approach for fingerprint matching can be extremely effective than the normal minutiae based matching methods. Three principal phases for the presented fingerprint matching are: Minutiae extraction and perform matching throughout Level 2, Pores extraction along with perform matching throughout Level 3 and next fingerprint recognition. The most important characteristics on the presented algorithm are: rapid detection, very fast in rejection, far more accurate compared to typical minutiae matching. Another advantage of presented technique can be that none of image processing techniques are require for matching. Our next objective can be
considering brand-new computational geometry design for matching along with classification in order to far more resistant against low quality and noise in fingerprints.

9 References

1. Bebis, George, Deaconu, Taisa. and Georgiopoulos, Michael., Fingerprint Identification Using Delaunay Triangulation, IEEE International Conference on Information Intelligence and Systems, (1999)

2. Mohammad A.U. Khan , M. Khalid Khan and M. Aurangzeb Khan, Fingerprint Image Enhancement Using Decimation-free Directional Filter Bank. Information Technology Journal, 4: 16-20, (2005)

3. Nasir Rehan and Khalid Rashid , Multi-matcher Based Fingerprint Identification System. Journal of Applied Sciences, 4: 611-618, (2004)

4. D. Maio, D. Maltoni, R. Cappelli, J.L. Wayman, and A.k. Jain. FVC2000: Fingerprint Verification Competition. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(3):402–412, (2002)

5. M. Ray, P. Meenen, and R. Adhami. A novel approach to fingerprint pore extraction. Southeastern Symposium on System Theory, pages 282–286, (2005)

6. Pankanti S., Prabhakar S., and Anil K. Jain. On the Individuality of Fingerprints. IEEE Trans. Pattern Anal. Mach. Intell., 24(8):1010–1025, (2002)

7. Abdullah Cavusoglu and Salih Gorgunoglu, A Robust Correlation Based Fingerprint Matching Algorithm for Verification. Journal of Applied Sciences, 7: 3286-3291, (2007)

8. D.R. Ashbaugh. Quantitative-Quantitative Friction Ridge Analysis: An Introduction to Basic and Advanced Ridgeology. CRC Press, (1999)

9. A.K. Jain, Y. Chen, and M. Demirkus. Pores and Ridges: High-Resolution Fingerprint Matching Using Level 3 Features. PAMI, 29(1):15–27, (2007)

10. Wang, Ch. and Gavrilova, L. Delaunay Triangulation Algorithm for Fingerprint Matching, The 3rd IEEE International Symposium of Voronoi Diagrams in Science and Engineering, (2006)

11. Poulos, M., Magkos, E., Chrissikopoulos, V., Alexandris, N., Secure Fingerprint Verification Based on Image Processing Segmentation Using Computational Geometry Algorithms, Proceedings of the IASTED International Conference on Signal Processing, Pattern Recognition, and Applications, Rhodes Island, Greece, June 30- July 2, ACTA Press, (2003)

12. W. M. Z. Somarny, N. S. Mariana , V. Neela , R. Rozita and A. R. Raha , Differentiation of Pathogenic Vibrio Species by RAPD. Journal of Medical Sciences, 2: 165-169, (2002)

13. Sack, J.-R. and Urrutia, J. Handbook of Computational Geometry, Elsevier, (2000)

14. Sharat Chikkerur, Alexander N. Cartwright, and Venu Govindaraju. Fingerprint enhancement using STFT analysis. Pattern Recogn., 40(1):198–211, (2007)

15. A.K. Jain, Y. Chen, and M. Demirkus. Pores and Ridges: High-Resolution Fingerprint Matching Using Level 3 Features. PAMI, 29(1):15–27, (2007)

16. F.E. Nwilene , A. Onasanya , O. Okhidievbie , Y. Sere , M.N. Ndjiondjop and I. Ingelbrecht , Identification and Differentiation of Three Nigerian Orseolia sp. by RAPD Markers. Asian Journal of Plant Sciences, 5: 50-53, (2006)
17. Khazaei, H., Mohades, A., Fingerprint Matching and Classification using an Onion Layer algorithm of Computational Geometry, International Journal of Mathematics and Computers in Simulation, Issue 1, Volume 1, (2007)

18. Jiang, X. D. and Yau, W. Y., “Fingerprint minutiae matching based on the local and Global Structures”, Proceedings of the 15th International Conference on Pattern Recognition, (2000)

19. Maltoni, D., Maio, D., Jain, A. K. and Prabhakar, S., Handbook of Fingerprint Recognition, Springer (2003)

20. P. Vijayaprasad, Md. N. Sulaiman, N. Mustapha and R.W.O.K. Rahmat, Partial Fingerprint Recognition Using Support Vector Machine. Information Technology Journal, 9: 844-848, (2010)

21. Mazaheri S. and Moosavi Tayebi R., Sadeghi Bigham B. "Fingerprint Matching using an Onion Layer algorithm of Computational Geometry Based on Level 3 Features", DICTAP2011, ISI Proceedings, Print in Springer Journal, France, Dijon, (2011)

http://www.springerlink.com/content/l472159830jk6050/

22. Moosavi Tayebi R., Mazaheri S. and Sadeghi Bigham B. "A New Fingerprint Matching Approach Using Level 2 and Level 3 Features", C3S2E’11, Print in ACM Proceeding, Canada, Montreal, (2011)

http://dl.acm.org/citation.cfm?id=1992906

23. Md. Mamunur Rashid and A.K.M. Akatar Hossain, Fingerprint Verification System Using Artificial Neural Network.Information Technology Journal, 5: 1063-1067, (2006)

24. Md. Rajibul Islam, Md. Shohel Sayeed and Andrews Samraj, Technology Review: Image Enhancement, Feature Extraction and Template Protection of a Fingerprint Authentication System. Journal of Applied Sciences, 10: 1397-1404, (2010)

25. Ross, S. Dass, and A. K. Jain, “A deformable model for fingerprint matching,”Pattern Recognit., vol. 38, no. 1, pp. 95–103, 2005.

26. X. Chen, J. Tian, X. Yang, and Y. Zhang, “An algorithm for distorted fingerprint matching based on local triangle feature set,”IEEE Trans. Inf. Forensics Security, vol. 1, no. 2, pp. 169–177, Jun. 2006.

27. C. Lee, S. Lee, J. Kim, and S. Kim, “Preprocessing of a fingerprint image captured with a mobile camera,” in Proc. IAPR Int. Conf. Biometrics (ICB), Hong Kong, Jan. 2006, pp. 348–355, Springer LNCS-3832.

28. D. Maio and D. Maltoni, “Ridge-line density estimation in digital images,” inProc. 14th ICPR, 1998, vol. 1, pp. 534–538.

29. L. Hong, Y. Wan, and A. K. Jain, “Fingerprint image enhancement: Algorithm and performance evaluation,”IEEE Trans. Pattern Anal. Mach. Intell., vol. 20, no. 8, pp. 777–789, Aug. 1998.

30. S. Lee, H. Choi, and J. Kim, “Fingerprint quality index using gradient components,”IEEE Trans. Inf. Forensics Security, vol. 3, no. 4, pp. 792–800, Dec. 2008.

31. K.V. Mardia and P.E. Jupp, Directional Statistics. New York: Wiley, 2000.

32. K. Choi, H. Choi, S. Lee, and J. Kim, “Fingerprint image mosaicking by recursive ridge mapping,”Special Issue Recent Adv. Biometrics Syst., IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 37, no. 5, pp. 1191–1203, Oct. 2007.