Ridge Pattern Representation for Fingerprint Indexing

M. Uysal[1], S. Gorgunoglu[1]

[1]Department of Computer Engineering, Karabuk University,
Karabuk, Turkey
uysal.tr@gmail.com

[1]Abstract—**Fingerprint is a widely used distinguishing biometric characteristic and indexing has an important role in fingerprint identification for large databases. A method to represent ridge pattern is proposed for fingerprint indexing. Ridge pattern within a triangular area is represented by a numerical value, which is suitable to use as an index and very robust to elastic distortion. Experimental results are promising. The proposed representation method can be utilized to further narrow down the search space of databases by the other indexing methods, especially the ones based on minutiae triplets.**

　Index Terms—**Fingerprint indexing, ridge pattern, minutia triplet.**

I. INTRODUCTION

It is impractical to scan entire database for fingerprint identification. As a solution, exclusive classification approaches distribute fingerprints into distinct classes. Nevertheless, number of classes, which is generally small and fingerprints are unevenly distributed among them, limits the selectivity of fingerprint classification. The other approach is continuous classification [1], that is indexing.

Fingerprint indexing assigns numerical values extracted from their features to each fingerprint in the database. At the retrieval stage, queried fingerprint, too, is assigned numbers which are used to calculate similarities between the query fingerprint and the template fingerprints in the database. At the end, similar fingerprints are retrieved. So, only this set of fingerprints are compared with the query fingerprint.

Some of the fingerprint indexing methods are mainly based on orientation fields, such as [2] and [3], while the most are based on minutiae, for example [4]–[10], among them [4]–[9] use minutia triplets. Few studies are based on ridges. The paper [9] extends its minutia triplet based indexing framework by utilizing ridge curve parameters, and [11] combines minutiae with surrounding ridges to form substructures. Another study [12] analyses the usefulness of ridge curvature information. Also, many different techniques, which cannot be categorized into above groups, exist. For instance, [13] compares indexing performances of finger code with directional image and minutiae triplets, [14] uses three kinds of symmetrical filter responses and [15] uses Minimum Average Correlation Energy filter response.

Ridges effectively represent fingerprints [16]. However,

most of the indexing algorithms use minutiae [17]. This study proposes a method to represent ridge patterns in fingerprints. To index a fingerprint, triangular areas are extracted from the fingerprint image. Within each triangular area, ridge lines are traced to form a binary value. These binary values are used for indexing.

The details of getting index values from ridge patterns are illustrated in Section II. An implementation of ridge pattern representation is described in Section III and Section IV gives the experiment results of the implementation. Section V concludes the paper.

II. PROPOSED METHOD

To represent ridge patterns, ridges enclosed by triangles are used. Delaunay triangulation on minutiae can give the necessary triangle set. So, each Delaunay triangle encloses a ridge pattern to be represented.

Ridges are traced on binarized fingerprint images. For minutiae detection, an open source program, called MINDTCT, is used. It is distributed with nbis [18] (NIST Biometric Image Software) suite by NIST (National Institute of Standards and Technology). Prior to minutiae detection, MINDTC binarizes the input image and the implementation of the proposed method, which is described in Section III, used its binarization output. MINDTCT gives minutiae positions consistent with their types. All ridge ends are at black points and all bifurcation positions are at white points.

A triangular ridge pattern is processed as in the following way:

The 3 edges, starting from the shortest one, are examined with their crossing ridges in the counter-clockwise order. For each triangle edge, 32 bits are reserved from the index and for each ridge crossing that edge, 2 bits are reserved from the 32 bits in the order sorted by the ridges' positions

Ridges crossing the examined edge are traced starting from the intersection point at the edge until exiting the triangle. If the other point at which the ridge crosses the triangle is:

　− at the same edge, 00,

　− at the left edge, 01,

　− at the right edge, 10,

　− both at the left and right edge, 11,

binary values are written at the reserved position of the index. If the ridge does not cross the triangle any more, 00 binary values are written.

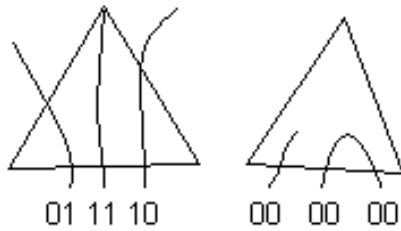Example situations are shown in Fig. 1.

Fig. 1. Getting index values from ridge lines.

If there are less than 16 ridges, as usual, 00 binary values are written in empty bit positions. In case, there are more than 16 ridges, only the first 16 ridges are considered, and the rest are discarded.

Repeating the above procedure on all the ridges of the 3 edges in counter-clockwise order, 96 bits of the index value are found. For the minutiae types and other features, which can be added later, one more 32 bit section is used. Binary 0 is written for ridge end, while binary 1 is written for ridge bifurcation to this index section.

For the sample triangular ridge pattern in Fig. 2, a 128 bit representative value is shown as follows:

01 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00
10 10 10 10 10 00 00 00 00 00 00 00 00 00 00 00
01 01 01 01 01 10 10 00 00 00 00 00 00 00 00 00
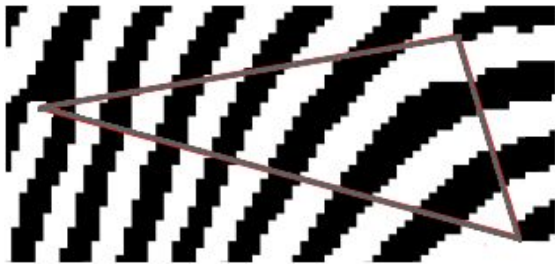00 10 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Fig. 2. Sample triangular ridge pattern.

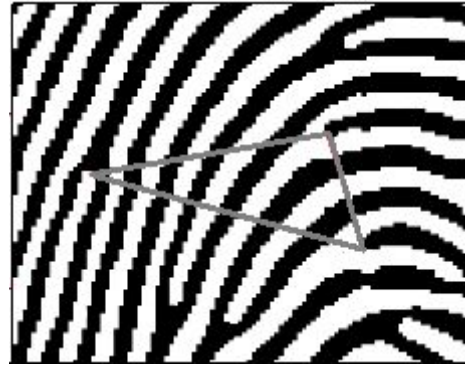Ridge pattern representation has the following properties:

It represents a ridge pattern with a scalar value, which is an advantage for indexing.

Adjacent ridges usually cross same edges. Therefore, the proposed ridge pattern representation has many repeated 2 bit elements, which makes this representation suitable for compression.

It is rotation and translation invariant, because the extraction of representation value always starts with the shortest side of a triangle and it does not need a reference point

It is robust to elastic distortion. Even if the edges of the triangular area change with elastic distortion, ridge pattern representation does not change, because ridge pattern inside the triangular area does not change. For example, Fig. 3 shows the non-stretched and stretched impressions of the same finger. Ridge pattern representations are the same for the both impressions.

Instead of delaunay triangles, more stable structures can be preferred. For example, low order (0 and 1) delaunay triangles are reported to be more robust to elastic distortions [7]. In addition, expanded delaunay triangles can be utilized against non-detected minutiae [8].

a)

b)

Fig. 3. Non-stretched impression (a), stretched impression (b).

Although ridge pattern representation robust to elastic distortion, it is sensitive to other effects causing distorted ridge pattern. For instance, even partially non-detected ridges or false minutiae can change ridge patterns.

III. IMPLEMENTATION

Ridge pattern representation was implemented to test its selectivity. Minutiae triplets are extracted by using Delaunay triangulation. Index values are extracted from Delaunay triangles using the proposed representation scheme.

Three of the remaining bits of the last 32 bits are used for binarized relative minutiae directions. For example, if direction of a minutia is between 0 and   radian relative to its neighbouring minutia in clockwise order, its corresponding bit is 0. Otherwise it is 1.

One more bit is used in such a way that it shows whether the left side is smaller than the right side, relative to the shortest side of the triangle.

Areas of triangles are calculated and stored with indices. They are used for similarity calculation. For each index in a query fingerprint, scores of all fingerprints pointed by the indices with the same value are incremented. Score calculation is done as follows:

Let $Tq = \{Tq1, Tq2, Tq3, \ldots, Tqm\}$ be the set of minutiae triplets of a queried fingerprint fq and for each Tqi with $i \in \{1, 2, 3, \ldots, m\}$ Aqi be its area and Iqi be its corresponding index value.

For a minutiae triplet Tkj of a fingerprint fk in the

database, let Ikj = Iqi be its index and Akj be its area. (j ∈ {1, 2, 3, …, n}).

A dissimilarity function ds(Aqi, Akj) is defined as in (1)

$$ds(Aqi, Akj) = \begin{cases} \dfrac{Aqi}{Akj}, & Aqi > Akj, \\ \dfrac{Akj}{Aqi}, & Aqi \le Akj. \end{cases} \quad (1)$$

The score increment Skj of fk due to Tqi is given by (2)

$$Skj = \begin{cases} \dfrac{Akj}{m.n}, & ds(Aqi, Akj) < 1.07, \\ \dfrac{Akj}{2.m.n}, & 1.07 \le ds(Aqi, Akj) < 1.11, \\ \dfrac{Akj}{4.m.n}, & 1.11 \le ds(Aqi, Akj) < 1.15, \\ \dfrac{Akj}{8.m.n}, & 1.15 \le ds(Aqi, Akj) < 1.19, \\ \dfrac{Akj}{16.m.n}, & 1.19 \le ds(Aqi, Akj) < 1.23, \\ \dfrac{Akj}{32.m.n}, & 1.23 \le ds(Aqi, Akj) < 1.27. \end{cases} \quad (2)$$

The score Sk of fk is calculated as in (3)

$$Sk = \sum_i \sum_j Skj(Iqi = Ikj). \quad (3)$$

Fingerprints are ordered by their scores. At the end, a permitted number of fingerprints with the highest scores are retrieved.

IV. EXPERIMENTAL RESULTS

The application of the proposed representation method was tested on FVC 2000 DB2, FVC2000 DB3 [19] and FVC 2000 DB1 [20] databases.

Penetration rate refers to the ratio of the retrieved fingerprint count to the number of fingerprints in the database. Hit rate means ratio of successful trials to all trials. It shows the probability that a searched fingerprint exists in the result set of fingerprint indexing system.
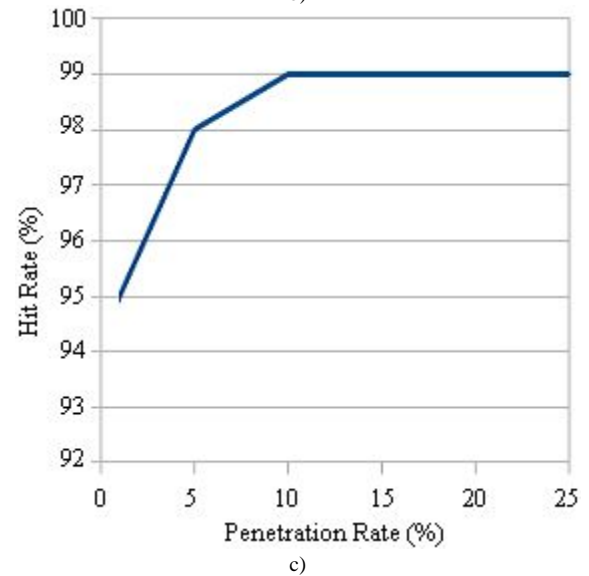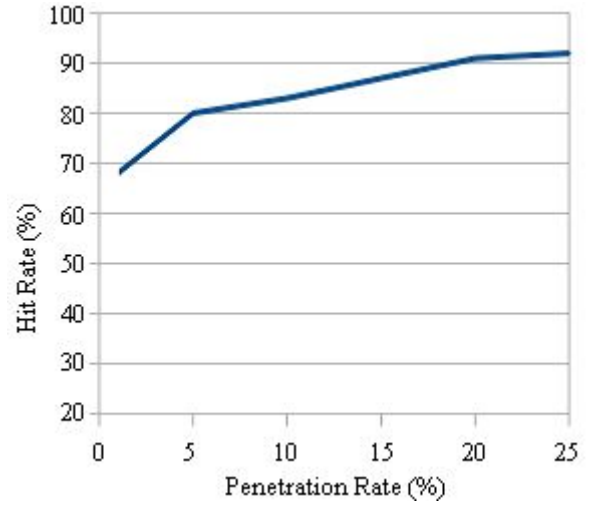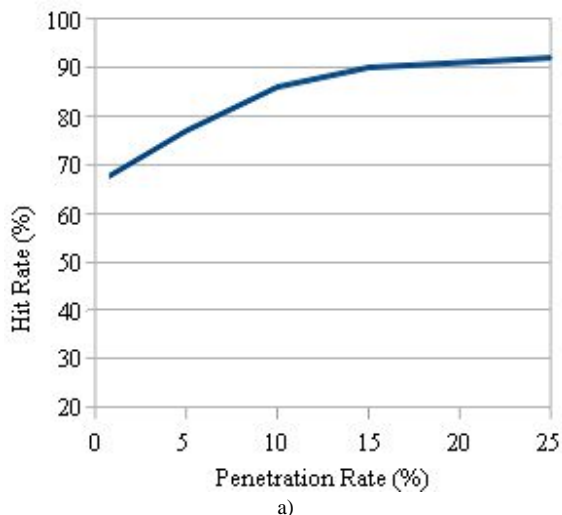
a)

b)

c)

Fig. 4. Hit Rate versus Penetration Rate on various FVC Databases: FVC2000 DB2(a), FVC2000 DB3(b), and FVC2002 DB1 (c).

The results are shown in Fig. 4.

State of the art indexing methods, e.g. [10], [11], [17], reported in literature have hit rates up to 97 % on FVC2000 DB2, 92 % on FVC2000 DB3 and 99 % on FVC2002 DB1 for a penetration rate of 5 %.

The implementation using the ridge pattern representation as described in the previous section, has hit rates lower than the state of the art indexing method for FVC2000 databases but its performance is comparable to the other indexing methods for FVC2002 DB1.

V. CONCLUSIONS

The applied model relies on both ridge pattern and stability of Delaunay triangles. More stable structures, such as low order Delaunay triangles [7] or expanded Delaunay triangles [8], can be used to increase hit rates.

MINDTCT is used for minutiae detection. Although its binarization method has enhancing effect, it does not enhance images prior to binarization. Fingerprint images should be enhanced before MINDTCT's processing steps.

All fingerprints with a specified number of highest scores are obtained as a result set. If a score evaluation step can be applied such that fingerprints with insufficient score are eliminated, more successful hit rates will be exhibited.

Other geometric features of minutiae triplets, such as length of triangle edges and angles between them have not been added to test the performance. Binarized relative minutiae directions were used as part of the index values. Features extracted from minutia and giving more successful results should be sought. Adding the other fingerprint features a more successful indexing technique based on the ridge pattern representation can be developed.

REFERENCES

[1] A. Lumini, D. Maio, D. Maltoni, "Continuous versus exclusive classification for fingerprint retrieval", *Pattern Recognition Letters*, vol. 18, pp. 1027–1034, 1997. [Online]. Available: http://dx.doi.org/10.1016/S0167-8655(97)00127-X

[2] Y. Wang, J. Hu, D. Phillips, "A fingerprint orientation model based on 2D fourier expansion (FOMFE) and its application to singular-point detection and fingerprint indexing", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 573–585, 2007. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2007.1003

[3] M. Liu, P. T. Yap, "Invariant representation of orientation fields for fingerprint indexing", *Pattern Recognition*, vol. 45, no. 7, pp. 2532–2542, 2012. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2012.01.014

[4] R. S. Germain, A. Califano, S. Colville, "Fingerprint matching using transformation parameter clustering", *IEEE Computational Science and Engineering*, vol. 4, no. 4, pp. 42–49, 1997. [Online]. Available: http://dx.doi.org/10.1109/99.641608

[5] B. Bhanu, X. Tan, "A Triplet based approach for indexing of fingerprint database for identification", In *Proc. Third Int. Conf. Audio- and Video-Based Biometric Person Authentication*, pp. 205–210, 2001.

[6] G. Bebis, T. Deaconu, M. Georgiopoulos, "Fingerprint identification using delaunay triangulation", In *Proc. 1999 IEEE Int. Conf. Intelligence, Information, and Systems*, 1999, pp. 452–459.

[7] X. Liang, A. Bishnu, T. Asano, "A robust fingerprint indexing scheme using minutia neighborhood structure and low-order delaunay triangles", *IEEE Trans. Information Forensics and Security*, vol. 2, no. 4, pp. 721–733, 2007. [Online]. Available: http://dx.doi.org/10.1109/TIFS.2007.910242

[8] A. Munoz-Briseno, A. Gago-Alonso, J. Hernandez-Palancar, "Fingerprint indexing with bad quality areas", *Expert Systems with Applications*, vol. 40, no. 5, pp. 1839–1846, 2013. [Online]. Available: http://dx.doi.org/10.1016/j.eswa.2012.09.018

[9] A. Ross, R. Mukherjee, "Augmenting ridge curves with minutiae triplets for fingerprint indexing", in *Proc. SPIE Conf. Biometric Technology for Human Identification IV*, 2007.

[10] R. Cappelli, M. Ferrara, D. Maltoni, "Fingerprint indexing based on minutia cylinder-code", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 33, no. 5, pp. 1051–1057, 2011. [Online]. Available: http://dx.doi.org/10.1109/TPAMI.2010.228

[11] J. Feng, A. Cai, "Fingerprint indexing using ridge invariants", in *Proc. 18th Int. Conf. Pattern Recognition*, vol. 4, 2006, pp. 433–436.

[12] S. Biswas, N. K. Ratha, G. Aggarwal, J. Connell, "Exploring ridge curvature for fingerprint indexing", in *Proc. 2nd IEEE Int. Conf. Biometrics: Theory, Applications and Systems*, 2008, pp. 1–6.

[13] J. De Boer, A. M. Bazen, S. H. Gerez, "Indexing fingerprint databases based on multiple features", in *ProRISC 2001, Annual Workshop on Circuits, Systems and Signal Processing*, 2001, pp. 300–306.

[14] J. Li, W. Y. Yau, H. Wang, "Fingerprint indexing based on symmetrical measurement", in *Proc. 18th Int. Conf. Pattern Recognition*, 2006, vol. 1, pp. 1038–1041.

[15] T. Liu, G. Zhu, C. Zang, P. Hao, "Fingerprint indexing based on singular point correlation", in *Proc. IEEE Int. Conf. Image Processing*, vol. 3, pp. 293–296, 2005.

[16] J. Feng, Z. Ouyang, A. Cai, "Fingerprint matching using ridges", *Pattern Recognition*, vol. 39, no. 11, pp. 2131–2140, 2006. [Online]. Available: http://dx.doi.org/10.1016/j.patcog.2006.05.001

[17] A. Gago-Alonso, J. Hernandez-Palancar, E. Rodriguez-Reina, A. Munoz-Briseno, "Indexing and retrieving in fingerprint databases under structural distortions", *Expert Systems with Applications*, vol. 40, no. 8, pp. 2858–2871, 2013. [Online]. Available: http://dx.doi.org/10.1016/j.eswa.2012.12.004

[18] NBIS Biometric Image Software. [Online]. Available: http://www.nist.gov/itl/iad/ig/nbis.cfm

[19] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, A. K. Jain, "FVC2000: Fingerprint verification completion", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 402–412, 2002. [Online]. Available: http://dx.doi.org/10.1109/34.990140

[20] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman, A. K. Jain, "FVC2002: Second fingerprint verification completion", in *Proc. 16th Int. Conf. Pattern Recognition*, vol. 3, pp. 811–814, 2002.