HIERARCHICALLY LINKED EXTENDED FEATURES IN FINGERPRINTS

Krzysztof Mieloch∗, *Axel Munk*

University of Göttingen
Institute for Mathematical Stochastics
Maschmühlenweg 8-10
37073 Göttingen, Germany

Preda Mihăilescu†

University of Göttingen
Mathematical Institute
Bunsenstraße 3-5
37073 Göttingen, Germany

ABSTRACT

Requirements for identifying and defining additional fingerprint features beyond minutiae, even not limited to level 3 details, has already been addressed by the community [1]. However, new feature proposals are based on refinements of existing features (e.g. a finer level of classification), or on the introduction of new single features (e.g. 3-D level features such as the ridge height) [2]. The new set of features proposed in this work does not only include additional fingerprint features individually but it also contains the information about their relationships such as line adjacency information at minutiae points or links between neighbouring fingerprint lines. The dual information, that is the interference between ridges and valleys, is included as well.

For the first confirmation of the quality of extracted features in application for the matching, the goodness of the extracted minutiae has been tested. The experiments have shown a decreased error rate when applying the minutiae to a standard minutiae-based matcher.

1. INTRODUCTION

Fingerprints utilisation as a personal code has had a long tradition due to their uniqueness and permanence. Two imprints are matched by comparison of characteristic features which are generally categorised into three levels [3] (See Fig. 1):

- **level 1 features (patterns)** - macro details of a fingerprint such as ridge flow and pattern type, which are used for fingerprint classification;
- **level 2 features (points)** - minutiae which have sufficient discriminating power to establish the individuality of fingerprints;
- **level 3 features (shapes)** - all dimensional attributes of a ridge such as ridge path deviation, width, shape, sweat pores, edge contour, incipient ridges, breaks, creases, scars and other permanent details.

Like level 2 features, level 3 features are also claimed to be permanent, immutable, and unique [4]. If properly utilised, they can provide discriminatory information. Forensic experts use them together with level 2 features in latent (partial) print examination.

One reason why level 3 features are hardly used by commercial software is that for an automatic extraction of most of those features high resolution (at least 1000 dpi) devices are required [4]. However,

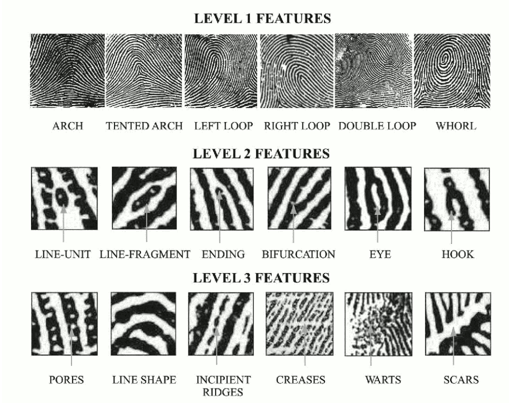


Fig. 1. Fingerprint feature levels [3].

some of the features (such as a ridge shape, breaks, creases) can also be easily found in images acquired by 500 dpi scanners which are commonly used in automatic systems. In this paper we will define such features and describe an extraction method for them. Since we will also extract additional features (such as ridge connection and neighbourhood information) we propose to call them *hierarchically linked extended features*.

This paper is organised as follows: The extraction algorithm and the extracted fingerprint structure are described in Section 2. Section 3 addresses the important duality principle applied to the minutiae validation. The neighbour information and its applications are presented in Section 4. The detailed description of extracted features is provided in Section 5. Finally, experimental results are presented in Section 6, with summary and feature work presented in Section 7.

2. FEATURES EXTRACTION AND STRUCTURE

In order to identify fingerprint features, first a fingerprint structure containing the fingerprint's crucial information has to be extracted. The essential engine which we are going to use for that task is a fingerprint line tracer. Various approaches for minutia detection which are based on following lines have been investigated in the literature [5–7]. However, there are principial differences between the existing methods and our:

- We additionally extract the so-called *extended features*, while other approaches focus solely on minutiae detection.
- Both ridge and valley lines (or simply the white and black lines) are processed. The objective of the apparently redun-

∗Supported by Graduiertenkolleg 1023 *Identification in mathematical models* of **DFG**

†Supported by the project *Explicite and algorithmic methods in number theory, cryptology and pattern recognition* of **Volkswagen foundation**

dant information is the correction of defects caused by noise. For example, in regions where black lines are broken the information from connected white lines is used to repair the black lines.

- We perform the tracing in an already binarised image. In contrast to grey-level images, in binarised images recognising the line's border while following it is much simpler and faster. Thus we can skip the costly centring on the line. In spite of the fact that we have to binarise the input image, it is still advantageous since the binarisation is much faster than centring steps. Furthermore, we do not totally dismiss the grey level image, and if necessary, one can always come back to it; for example, in regions where the binarised image is too much disturbed due to noise.
- We do not need to compute the orientation field before tracing. Moreover, the orientation field can easily be computed from the extracted features.

2.1. Preprocessing

Before we can apply the tracer algorithm to an acquired fingerprint image, some preprocessing steps on the image have to be performed.

Since common matching algorithms utilise only level 1 and 2 features, the additional features are treated as noise and are removed during the enhancement stage. Application of a directional smoothing filter, such as the Gabor filter [8], results in an enhancement of the input image. However, level 3 features, such as creases or scars, are being smoothed out at the same time. In order to preserve the information, we use a very weak smoothing Gabor filter and additional non-smoothing enhancement, such as holes filling and median filter.

Segmentation, that is the division between the containing information foreground and the background, is very important for our line following algorithm, so that only fingerprint lines and not the surrounding area is traced. For this task we apply the algorithm proposed by Bazen and Garez [9], which is based on a linear classifier.

2.2. Tracing

A fingerprint line has three possible end types: *bifurcation* (point where three lines join together), *border point* (point where a line hits the background), or *ending*. Additionally, a whorl type fingerprint can contain special lines without ends, that is closed lines.

Black and white lines are traced independently. The correspondence between the black and the white area is added in the "offline" stages after all lines have been traced.

The tracing process briefly presented in Table 1 is described in detail in the following subsections.

2.2.1. Aura

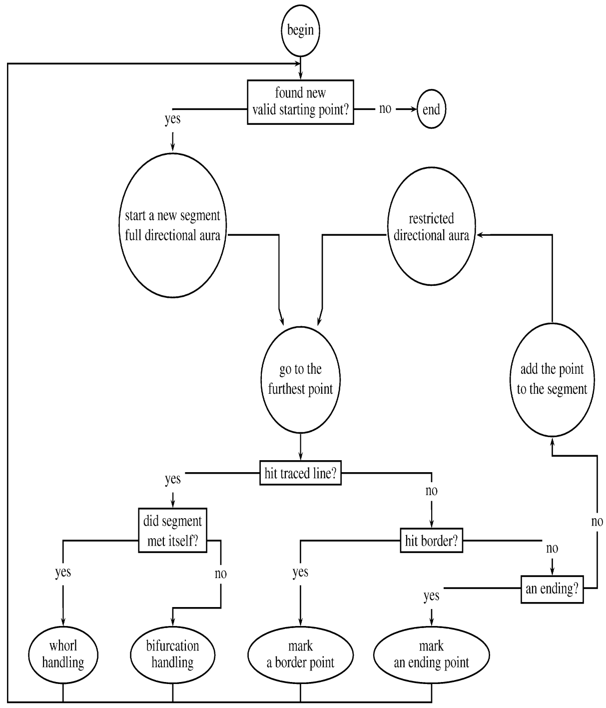
The main tracer tool is an **aura** which comprises **feelers** and **ridge border location**. The aura determines the optimal direction which the tracer should follow from a given point.

The feeler Φ_α is a directed line section with direction $\alpha \in \Omega$ starting at the position (x, y):

$$\Phi_\alpha(x,y) = \{(x,y) + \lfloor m \cdot (\cos\alpha, \sin\alpha) \rfloor : m \in \mathbb{N}\},$$

For each direction α the maximal value of m is computed, such that the section $\Phi_\alpha(x,y)$ stays within the fingerprint line. The section length for that m is computed as $l_\alpha(x,y) = |B_\alpha(x,y) - (x,y)|$ where $B_\alpha(x,y) : \mathbb{N}^2 \to \mathbb{N}^2$ returns the point where the section

Table 1. Tracing process



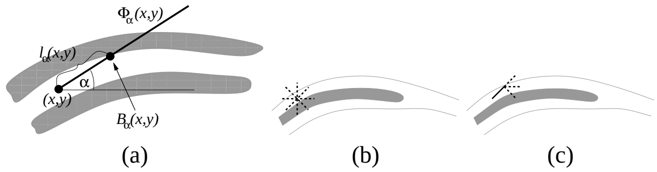


Fig. 2. (a) Sample values of functions for aura computation (b) Aura in a starting point (c) Aura in a following point

$\Phi_\alpha(x,y)$ crosses the line border and $|\cdot|$ denotes the Euclidean distance, see Fig. 2a.

The set of directions Ω for the first point in the traced line is a discrete set of equidistant angles in the circle:

$$\Omega = \left\{ \frac{i}{n} 2\pi : i = 0 \ldots n-1 \right\}$$

Since ridges and valleys do not take abrupt turns but change direction rather smoothly, we can restrict Ω to some subset $\Omega_0 \subset \Omega$ for the following points on the line so that only directions in a cone around the previous direction are taken into account. An example is shown in Fig. 2b and 2c. An extracted fingerprint line between two minutiae or border points will be called a segment.

2.2.2. Valid starting point

A point in a fingerprint image at which a single tracing process is started has to fulfil the following conditions:

- it is located in the foreground,
- it is within a fingerprint line,
- the line has not been traced yet.

Fig. 3. The order of bifurcation branches

In order to check the last two conditions, the full aura is computed for a given point. It has to have significantly larger values for two opposite directions compared with their orthogonal directions, and none of the feelers should meet an already traced line at the distance below a given threshold.

The candidates for starting points are taken from a regular grid. The tests performed have shown that the order in which the points on the grid are examined does not influence the result of the tracing.

2.2.3. Branches of a bifurcation

In order to properly identify the segments connected to a bifurcation, the adjacent segments are numbered according to a certain scheme: the first branch is the one on the opposite side of the smallest angle and all branches are ordered anticlockwise (Fig. 3a).

To find the correct order, the distances a, b, c (Fig. 3b) between the points on the connected segments at a given distance from a bifurcation point are computed. The smallest distance (in this case c) is between the second and the third branch.

2.3. Fingerprint structure

The information obtained from the tracing process is:

- minutiae and border points;
- closed segments (whorls) which are usually not captured by a typical minutiae extraction algorithm;
- ordinary segments with two ends from which one is called *front end* and the other *back end*.

Information stored for a minutia or a border point:

- position
- adjacent segments (3 for a bifurcation and 1 for an ending or a border point) with information whether the minutia is connected at the front or back end of the segment.

The following data are stored for segments:

- type (simple or closed)
- in case of a simple segment: minutiae or border points at both ends of the segment
- line course as a sequence of points.

3. VALIDATION

The white and black areas are naturally dual to each other. Generally, bifurcations in one area correspond bijectively to endings in the other area (Fig. 4a). Natural exceptions can be encountered only at singular points (Fig. 4b); further exceptions are consequences of noise and thus the existence of dual minutiae is an indicator for a reliable minutia point.

During the validation step, for every bifurcation its corresponding ending is searched for in the area between the second and third branch of the bifurcation, as shown in Fig. 4c and 4d. Once the dual ending has been found, the bifurcation position is reset to the ending position. This eliminates the problem of false positions of bifurcations during the tracing step. For each validated minutia, the pointer to the corresponding minutia is stored in the structure, which introduces the first connection between dual areas.

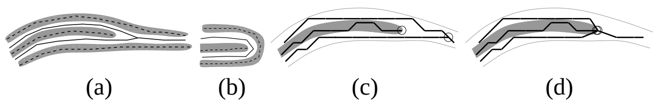
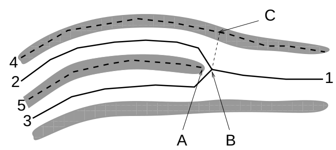


Fig. 4. Duality and its application. a) - a white bifurcation with corresponding black ending. b) - the exception for a singular point. c), d) - validation process.



At point C, line 4:

side:	right
position:	25
before:	2
after:	1
minutia:	B

At point A, line 5:

side:	left
position:	0
before:	none
after:	3
minutia:	none

At point A, line 5:

side:	right
position:	0
before:	none
after:	2
minutia:	none

Fig. 5. Sample change points for segment neighbours, front ends of 4 and 5 are on their right.

4. NEIGHBOUR INFORMATION

In order to provide for the required robustness of the features, additional information contained in the fingerprint structure must be evaluated. The most important information for that purpose is the neighbourhood information described in this section.

4.1. Segment neighbours

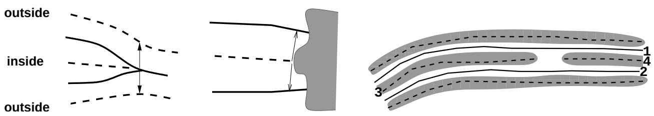
For further postprocessing and applications to other fingerprints tasks (classification, matching), neighbouring dual lines, which are the lines running parallel in the dual colour area, are identified for each traced line or, in case of an outermost segment, the background is marked as the neighbour.

This neighbour information is stored as a list of change points - the points on a line where the neighbours change. The information stored in one list element is:

- side of change - right or left looking from the front towards the back end of the segment;
- position of change - the length of the segment between the front end and the change point;
- neighbouring segment towards the front end (before);
- neighbouring segment towards the back end (after);
- minutia causing the change of the neighbour.

Examples for the stored information are shown in Fig. 5. For sake of simplicity, in the following description we assume that minutiae are adjacent to the segments at the corresponding segments' front ends.

Clearly, in a non-disturbed area the segment neighbour changes only if there is a minutia on the neighbouring segment. Hence, it



(a) (b) (c)

Fig. 6. Finding neighbours for a bifurcation (a) and a border point (b). Utilisation of neighbour information for bridging (c).

suffices to find the neighbours for all minutiae and border points in order to create the neighbour lists. Consequently, the computation time is reduced in comparison to identifying neighbours by following segments and checking at fixed lengths.

The points are processed in the following order:

- validated bifurcations
- border points
- unvalidated minutiae

For a validated bifurcation the information about *inside* neighbours already exists due to the duality principle (compare Fig. 6a). Consequently, the right neighbour of the second branch and the left neighbour of the third branch is the segment connected to the dual ending. Hence the neighbours of the segment connected to the ending are defined as well. In order to find the *outside* neighbours, we follow the directions orthogonal to the bifurcation direction until a segment in the dual area is met. For border points and unvalidated minutiae the strategy is similar to the one for "outside" neighbours at a bifurcation. We look for the neighbours in the directions orthogonal to a border point or an unvalidated minutia direction (Fig. 6b).

4.2. Neighbours at border points and unvalidated endings

In addition to the information about segment neighbours, the neighbour point (if one exists) is stored for each unvalidated ending or border point. For the situation presented in Fig. 7a at the border point B the following data is stored: the left neighbour is the border point C and the right neighbour is the unvalidated ending A.

4.3. Postprocessing using neighbours information

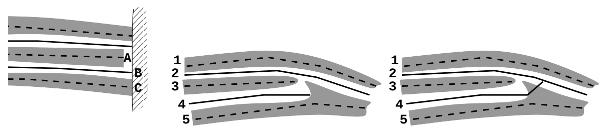
Bridging. For connecting disconnected lines dual information is used. Generally, if a ridge is disconnected, their dual neighbours are not broken as presented in Fig. 6c. We use this information to repair broken ridges.

In order to distinguish between a line break caused only by noise and that contained in a fingerprint (creases, scars), the number of neighbouring disconnected lines is counted. If it is less than or equal 3, we treat it as noise and consequently connect the lines. Otherwise the disconnection is marked as a scar for further processing.

Correcting unvalidated endings As already mentioned, an unvalidated minutia, apart from one for each singular point, is an indicator for an extraction error. The neighbour information can be used to repair such errors.

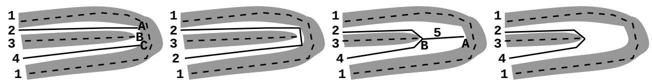
If an unvalidated ending has a border point as its minutia neighbour (Fig. 7a), its type is changed from ending to border point.

Another unvalidated ending occurs when the tracer gets stuck instead of finding a bifurcation as shown in Fig. 7b. In order to find and correct these errors, the neighbourhood information can be used and the segment adjacent to one of the endings is connected to its neighbour segment in such a way that the resulting bifurcation is the dual one for the second unvalidated ending (Fig. 7c).



(a) (b) (c)

Fig. 7. An unvalidated ending next to the background.The correction of an unvalidated ending



(a) before (b) after (c) before (d) after

Fig. 8. Correcting unvalidated endings located in a core area; before and after postprocessing.

Yet other rules apply for an unvalidated ending in a core area. First, the core has to be identified, which is an easy task with the neighbour information. The possible situations are presented in Fig. 8a and 8c. Points A and C have the same segment neighbour on opposite sides in the first case, and in the other case A has the same neighbours on both of its sides and the length of segment 5 is below the threshold. The corrections are presented in Fig. 8b and 8d.

5. FEATURES

The above presented fingerprint structure containing minutiae, segments and connectivity information is an important basis for extracting the information contained in a fingerprint. The general features are presented below. For other applications, not described in this work (e.g. classification), additional features such as flow lines can be extracted from the structure.

5.1. Standard features

In addition to the new features described below, the standard features such as orientation field or minutiae are also contained in the extracted structure. For example, the orientation field can be straightly computed from the angle between a linear piece of a traced line and the horizontal line.

5.2. Line distance

To compute the distance between two points p_1, p_2 lying on the same segment, their Euclidian distance can be used. However, in order to capture the shape of the line, we use the notion of **line distance** defined as the length of the segment's interval between p_1 and p_2.

5.3. Segment line shape

In order to describe the flow and the shape of a segment, the following descriptors can be introduced:

- length of the segment - line distance between the end points,
- cumulative direction change - the sum of direction changes along the segment line,

- winding number,
- curvature.

In order to compute the curvature each segment is approximated by a B-spline curve [10], which is an additional segment descriptor.

5.4. Scars and creases

As has already been mentioned, in contrast to simple line breaks, scars or creases are not being connected. They are marked as level 3 features to be used in the further matching process.

Structures which at first sight look like a scar or crease may turn out to be only noise, and consequently scars and creases should be validated by means of multiple acquisition. Furthermore, some scars can be temporary and disappear over time (e.g. a wound that heals). To distinguish between a permanent and a temporal scar some correction over time might have to be applied. If a fingerprint with scar was stored as a template and the scar has not been detected in a series of future acquisitions, the scar should be deleted from the template.

5.5. Fingerprint structure as a graph

We can regard the obtained fingerprint structure (extracted minutiae, border points, and segments) as a graph in which minutiae and border points are vertices and segments are edges between them. Since it may happen that two edges join the same bifurcations (e.g. for an eye) it it is in general a *multigraph* (a graph in which multiple edges and loops are allowed) [11]. In order to include duality information we add special edges that connect corresponding validated minutiae. The graph is defined as follows:

$$G_M = (V_M, E_M, E_D),$$

where V_M is the set comprising minutiae and border points. The sets of edges $E_M, E_D \subseteq V_M \times V_M$ fulfil the following conditions:

- $\{v_1, v_2\} \in E_M \iff v_1$ and v_2 are connected via a segment,
- $\{v_1, v_2\} \in E_D \iff$ one of v_1, v_2 is a bifurcation and the other one is an ending, and they are dual to each other.

Hence the vertex degree $d(v) \in \{1, 2, 3, 4\}$: border points have degree 1, endings have 1 (unvalidated) or 2 (validated), and bifurcations have 3 (unvalidated) or 4 (validated).

5.6. Virtual minutiae

A challenge in fingerprint matching process is the switching of the minutia type between an ending and a bifurcation. Thus in the most automatic fingerprint matching methods the type of a minutia is generally not taken into consideration [4]. An ending can change to a bifurcation[1] if the line adjacent to the ending gets connected to a neighbouring line. It can happen either during the enhancement of the image (e.g. by directional smoothing), or already during the acquisition due to skin elasticity. While the disruptive influence of the first one can be decreased by improving the enhancement method, the second problem cannot be avoided when contact-based sensors are used for acquisition as is commonly done.

Minutia A from Fig. 9a changes its type from an ending to a bifurcation (Fig. 9b). It bears more consequences: when minutia A is an ending, the points 1 and 2 lie in the same connected component but after the minutia type has changed, the points may be even not in the same dual connected component[2].

[1] Note that the corresponding dual bifurcation changes to an ending.

[2] For the definition of the dual connectivity see next subsection.

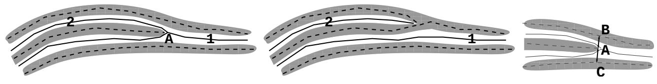


Fig. 9. An example of minutia type change and a virtual minutia

To deal with this problem we propose *virtual bifurcations*, that is we introduce additional points on the neighbour segments and add links between them and the ending (Fig. 9c), which results in an extension of graph G_M:

$$G_V = (V_M \cup V_V, E_M, E_D, E_V),$$

where V_V is the set of the additional points on the segments and $\langle m, v \rangle \in E_V \subseteq V_M \times V_V$ is a link between a virtual minutia and an ending or an end of the segment on which the virtual minutia lies.

5.7. Connected components

Three commonly applied connectivity systems in the graph G_M are:

- **simple** – two vertices are connected if there exists a path between them containing only edges from the set E_M.
- **dual** – two vertices are connected if there exists a path which may contain edges from both sets E_M and E_D.
- **extended** – a connectivity system can be extended with virtual minutiae. If there exists an edge in E_V connecting two different connected components, they are combined into one.

The above listed descriptors of a segment can naturally be extended (with respect to a defined connectivity system) to measures between two points lying on different segments. E.g. for computing the distance between two points not lying on the same segment the following procedure applies: if two points are in the same connected component, the line distance between them is the length of the shortest path. Otherwise, the distance is set to infinity.

6. QUALITY OF EXTRACTED FEATURES

We have aimed at checking the influence of the described features on the matching rate. As we are still developing a matcher that fully utilises all the extracted features, we have used a standard minutiae-based matching algorithm to check exclusively the quality of extracted minutiae. The minutiae-based matcher has been then extended to include some of the extended features.

6.1. Minutiae-based matching

In order to check the quality of the extracted minutiae, we have taken a commercial fingerprint recogntition software belonging to the top ten in the FVC 2006 [12] ranking list. We have chosen two databases from the FVC databases, an easy one (db2 from FVC2002) and a more difficult one (db3 from FVC2000). We have computed equal error rates by matching the fingerprint impressions according to the FVC protocol, which results in 2800 genuine and 4950 impostor pairs for each database and algorithm. We have extracted minutiae with both the entracer and with the comercial extractor. Then, we have used the commercial matcher to match the extracted minutia sets[3]. The results are presented in Table 2. These tests show that the

[3] For the tests we have not used any additional extracted information but only minutia position and direction.

Table 2. Equal error rates for minutiae matching

	commercial extractor	Entracer minutiae-based	Entracer consistency-based
FVC2002 db2	1.056%	0.953%	0.784%
FVC2000 db3	7.692%	7.253%	6.727%

error rates are reduced even when no additional extracted features are used. In the next step, we extended the matching algorithm so that it exploits the advantages of some of the extended features.

6.2. Consistency-based matching

First, the extracted minutiae has been matched with a minutiae-based method. Next, the following consistency test is applied.

Let $G_M = (U, E_M, E_D)$ and $H_M = (V, F_M, F_D)$ be the extracted fingerprint graphs for the fingerprints to be matched.

Let M be the set of all matched minutiae pairs

$$M := \{(u_i, v_i)\}_{i=1}^{n} \subseteq U \times V$$

M is a relation between the elements of the set U and the set V. Let $U_m \subseteq U$ and $V_m \subseteq V$ be its domain and its range, respectively.

For the consistency test we take two minutiae $u_1, u_2 \in U_m$ lying in the same extended dual connected component of the graph G_M. For the corresponding matched minutiae in the second fingerprint

$$v_1, v_2 \in V_m : (u_1, v_1) \in M \wedge (u_2, v_2) \in M$$

we check the following conditions in a top-down manner, that is, if any of the earlier condition is not fulfilled, the following conditions will not be checked:

- v_1 and v_2 lie in the same extended dual connected component of the graph H_M,
- the difference of line distances is below a given threshold:

$$|d_l^{ed}(u_1, u_2) - d_l^{ed}(v_1, v_2)| < t_d,$$

- the difference in the cumulative direction change on the shortest path between the vertices is below a certain threshold

$$|\phi^{ed}(u_1, u_2) - \phi^{ed}(v_1, v_2)| < t_\phi.$$

Consequently, the scores for pairs (u_1, v_1) and (u_2, v_2) are updated according to the result of the consistency test.

This method does not exhaust the full power of the extended features as it only decreases scores for falsely matched minutiae, whereas the main reason for using extended features is to decrease the false rejection rate in case level 2 features (usually minutiae) do not provide enough information to declare a pair as genuine [3]. However, even with the described simple extension we were able to decrease the error rates, as shown in the third column of Table 2.

7. CONCLUSION AND FUTURE WORK

Traditionally, minutiae have been used as primary features for fingerprint comparison and global features – for classification. The goal of this work has been to investigate the use of alternative features that can be extracted from fingerprint images, which includes a proposal for introducing a hierarchically linked feature set and a new entropy-sensitive extraction approach. The universal class of features defined in this work can be applied in various stages of fingerprint processing since the features combine the global and local scales which are ordinarily used separately. Consequently, this approach makes the automatic fingerprint recognition procedure more similar to the way human experts work. The performed tests for matching have shown a performance increase as the result.

We are still developing methods for fingerprint matching and classification which fully utilise the new features.

The applications of the new features for other tasks in fingerprint processing are also worth consideration. The extracted features are robust, which is crucial for securing a template. Finally, since the features are extracted from the underlying fingerprint structure exclusively, storing of the structure instead of the whole fingerprint requires less memory, and can save resources. Hence, it would be desirable to investigate the problem of reconstruction of fingerprint image from its fingerprint structure.

8. REFERENCES

[1] *CDEFFS: the ANSI/NIST Committee to Define an Extended Fingerprint Feature Set*. [Online]. Available: http://fingerprint.nist.gov/standard/cdeffs

[2] A. Hicklin, *CDEFFS – Extended Feature Sets*, Latent Testing Workshop, National Institute of Standards and Technology, 2006. [Online]. Available: http://www.itl.nist.gov/iad/894.03/latent/presentations.htm

[3] A. K. Jain, Y. Chen, and D. Meltem, “Pores and ridges: high-resolution fingerprint matching using level 3 features,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 15–27, 2007.

[4] S. Pankanti, S. Prabhakar, and A. K. Jain, “On the individuality of fingerprints,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, pp. 1010–1025, Aug. 2002.

[5] D. Maio and D. Maltion, “Direct gray-scale minutiae detection in fingerprints,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 1, pp. 27–40, 1997.

[6] X. Jiang, W.-Y. Yau, and W. Ser, “Detecting the fingerprint minutiae by adaptive tracing the gray-level ridge,” *Pattern Recognition*, vol. 34, pp. 999–1013, 2001.

[7] H. Liu and Chang, “Direct minutiae extraction from gray-level fingerprint image by relationship examination,” in *Proc. International Conference on Image Processing*, vol. 2, 2000, pp. 427–430.

[8] L. Hong, Y. Wan, and A. K. Jain, “Fingerprint image enhancement: Algorithm and performance evaluation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 8, pp. 777–789, Aug. 1998.

[9] A. M. Bazen and S. H. Gerez, “Segmentation of fingerprint images,” in *Proc. ProRISC 2001 Workshop on Circuits, Systems and Signal Processing*, 2001, pp. 276–280.

[10] M. M. S. Chong, T. H. Ngee, L. Jun, and R. K. L. Gay, “Geometric framework for fingerprint image classification,” *Pattern Recognition*, vol. 30, no. 9, pp. 1475–1488, 1997.

[11] B. Bollobas, *Graph Theory*. New York: Springer, 1979.

[12] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*. New York: Springer, 2003.