To appear in ICPR, Dec. 2008

Filtering Large Fingerprint Database for Latent Matching

Jianjiang Feng and Anil K. Jain
Department of Computer Science and Engineering
Michigan State University

Abstract

Latent fingerprint identification is of critical importance to law enforcement agencies in apprehending criminals. Considering the huge size of fingerprint databases maintained by law enforcement agencies, exhaustive one-to-one matching is impractical and a database filtering technique is necessary to reduce the search space. Due to low image quality and small finger area of latent fingerprints, it is necessary to use several features for an efficient and reliable filtering system. A multi-stage filtering system is proposed, which utilizes pattern type, singular points and orientation field. We have tested our system by searching 258 latent fingerprints in NIST SD27 against a background database containing 10,258 rolled fingerprints (obtained by combining 2,000 in NIST SD4, 8,000 in SD14 and 258 in SD27). Although latent fingerprints contain very limited information, the filtering system not only improved the matching speed by three fold but also improved the rank-1 matching accuracy from 70.9% to 73.3%.

1. Introduction

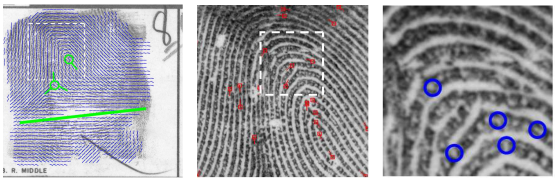
Fingerprint images can be broadly classified into three categories, namely, (i) rolled, (ii) plain and (iii) latent (see Figure 1). Rolled and latent fingerprints are mainly used in forensic applications, whereas plain fingerprints are mainly used in commercial and government applications. Rolled fingerprints are obtained by rolling a finger from one side to the other ("nail-to-nail") in order to capture all the ridge-details of a finger. Plain impressions are those in which the finger is pressed down on a flat surface but not rolled. While plain impressions cover a smaller area than rolled prints, they typically do not have the distortion introduced during rolling. Rolled and plain impressions are obtained either by scanning the inked impression on paper or by using live-scan devices. Since rolled and plain fingerprints are acquired from co-operative subjects, they are typically of good quality and are rich in information content. In contrast, latent fingerprints are lifted from surfaces of objects that are inadvertently touched or handled by a person through a variety of means ranging from simply photographing the print to more complex dusting or chemical processing [6]. It is the matching of a latent fingerprint against a database of rolled prints that is extremely challenging and of utmost importance in forensics to identify and apprehend criminal suspects.

(a) (b) (c)

Figure 1. (a) Rolled, (b) plain and (c) latent fingerprint

Once latent fingerprints are "lifted" from a crime scene, they are first searched against the fingerprint database of local police agency, then the database of state police agency, and finally the database of federal agency (FBI in the United States). The size of the rolled fingerprint database maintained by the police department of a typical large city can be of the order of a million. The Integrated Automated Fingerprint Identification System (IAFIS) maintained by FBI contains the fingerprints for more than 55 million subjects. In practice, a search of such databases involves some keywords (attributes), for example gender and age, to reduce the search space. However, even after keyword-based filtering, the filtered database is still large and, in many cases, such keyword information is not available. To obtain a greater reduction of search space, image-based filtering techniques are necessary.

Almost all fingerprint matchers are based on match-



(a) (b) (c)

Figure 2. (a) Orientation field, singular points and flexion crease, (b) minutiae and (c) pores

ing features extracted from fingerprint images. Features in fingerprints can be classified into three levels according to their visibility at different resolutions (See Figure 2). While level 2 and level 3 features like minutiae, pores, ridge shapes, can be used to individualize fingerprints, level 1 features, like orientation field, singular points, pattern type, flexion crease, and finger direction, are also important for fingerprint matching, since they not only can be used for filtering database but also serve as reference frame for minutiae matching.

We have described a minutiae-based latent fingerprint matching system in [4], which did not use level 1 features for database filtering and coarse alignment. Although the query time of the system in [4] is reasonable for medium-sized database, when the background database becomes large, the query time will become unacceptable. To improve the matching efficiency, filtering should be performed before minutiae matching algorithm (Figure 3) which can exclude a large number of background fingerprints from minutiae matching.

Existing fingerprint database filtering techniques are based on pattern classification [2], relationships among level 1 features [8], orientation information around reference points [5] and relationships among level 2 features [3]. Due to limited information contained in latent fingerprints, a single filtering technique may not be adequate. Therefore, a multi-stage filtering system is proposed, which utilizes pattern type, singular points and orientation field. Minutiae matching algorithm is performed only for the fingerprints passing all the three stages. We have tested our system by searching 258 latent fingerprints in NIST SD27 against a background database containing 10,258 rolled fingerprints (obtained by combining NIST SD4, SD14 and SD27). Our system obtained a penetration rate of 39.0% at an accuracy of 97.3%. Meanwhile, the rank-1 identification accuracy was also improved due to that fact that a large number of background fingerprints were excluded from minutiae matching.

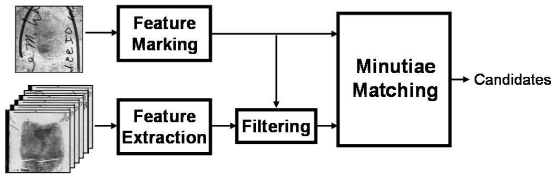


Figure 3. Latent fingerprint matching system

2. Feature marking for latent fingerprint

For all the latent fingerprints in NIST SD27, the minutiae have been already marked by a team of latent examiners. Other level 1 features in these images (pattern type, orientation field, reference points and singular points) were marked by the authors. In NIST SD27, only for about 50% of the latents, the pattern type can be determined exclusively. For other latents, multiple pattern types are specified. Orientation field (defined in blocks of 16*16) is first estimated using gradient-based algorithm and then manually modified. For fingerprints whose pattern area is visible, reference points (for arch) or singular points (for non-arch) are manually marked. The reference point of an arch is defined as the maximum curvature point. Both reference points and singular points have five common attributes: x, y, direction, radius of position uncertainty and radius of direction uncertainty. Singular points have an additional attribute: type, which can be concave, convex or unknown for a core; left, right or unknown for a delta.

3. Feature extraction for rolled fingerprint

For rolled fingerprints, all the features are extracted automatically.

3.1. Orientation field and curvature field

Orientation field is estimated based on extracted ridge lines [4]. Curvature field is then computed based on the orientation field. To compute curvature at a point $s = (x, y)$, two pseudo ridges $\{s, p_1, p_2, ..., p_m\}$ and $\{s, q_1, q_2, ..., q_m\}$ are traced from s. These points are difined as $p_i = p_{i-1} + \sigma \cdot o_{p_{i-1}}$ and $q_i = q_{i-1} - \sigma \cdot o_{p_{i-1}}$, where σ denotes the step length and o_p denotes the orientation vector at point p. In our experiments, $\sigma = 4$ and $m = 8$. Let P and Q denote the unit vectors $\overrightarrow{p_n s}/|\overrightarrow{p_n s}|$ and $\overrightarrow{sq_n}/|\overrightarrow{sq_n}|$. The normal vector at point s is computed as $N_s = P - Q = K_s \cdot U_s$, where K_s and U_s denote the curvature and unit normal vector at point s.

3.2. Singular point

A pyramid of orientation fields is built. The Poincaré index method is used to extract singular points at the high level and their locations are refined at the lower level. The direction of core and the upward direction of delta is computed by comparing with orientation field of standard core and delta [1]. Pseudo ridge tracing is performed to validate singular points as proposed in [7].

3.3. Pattern type

In this paper, fingerprints are classified into five pattern types: plain arch, tented arch, left loop, right loop and whorl. If two cores or deltas are detected, fingerprint pattern is directly set as whorl; otherwise fingerprint is classified by tracing and analyzing pseudo ridges using an improved version of the algorithm in [2]. Plain arch and tented arch are distinguished based on the existence of singular points. We also used a more robust method than [2] to distinguish left loops from right loops. If both core and delta have been detected, loop type is distinguished by checking the position of delta with respect to core direction. If only core is available, loop type is determined by checking if the core points to the left or right. If neither core nor delta has been detected, the maximum curvature point of all loop ridges is detected and loop type is determined by checking if the normal vector at maximum curvature point points to left or right.

3.4. Reference point of plain arch

Reference point of a fingerprint of plain arch type is defined as a maximum curvature point where the symmetry value of the orientation field with respect to normal vector is not less than a predefined threshold. The symmetry value at a point p is defined as follows. A local coordinate system is defined with p as origin and the normal vector at p as y axis. A set of sampling points $\{x, y | x = (\pm\sigma, \pm 2\sigma, \cdots, \pm m\sigma), y = (0, \pm\sigma, \pm 2\sigma, \cdots, \pm m\sigma)\}$ is defined and ridge orientation at these points is computed with respect to the local coordinate system. The symmetry value is computed as the mean symmetry value of orientation of all symmetric points with respect to y axis.

4. Multi-stage filtering

Before minutiae matching is performed, a background (database) fingerprint undergoes multi-stage filtering, which consists of three stages: pattern filtering, singularity filtering, and orientation filtering. Failing to pass any of the three stages will lead to rejection of a background fingerprint. The pattern filtering checks if the patterns of two fingerprints are the same. Singularity filtering checks if the relationships between singular points of two fingerprints are consistent. Orientation filtering checks if orientation fields around singular points of two fingerprints are consistent.

4.1. Singularity filtering

The correspondence between singular points of two fingerprints is determined based on the type of singular points. If the number of corresponding singular points is less than two, the background fingerprint is deemed as passed; otherwise, the similarity between relationships of any two pairs of corresponding singular points is computed and the mean similarity is compared to a predefined threshold to determine if it should pass. The relationships between two singular points include the distance, the angle between their directions, and the angle between the direction of one singular point and the line connecting them. Let $\delta_d, \delta_\alpha, \delta_\beta$ denote the absolute values of difference between distance and two angles of two pairs of singular points. The similarity (s_s) between relationships of two pairs of corresponding singular points is computed by

$$s_s = 0.4f(\delta_d, 40) + 0.3f(\delta_\alpha, 30) + 0.3f(\delta_\beta, 30) \quad (1)$$

where $f(x, a) = max(1 - x/a, 0)$.

4.2. Orientation filtering

The similarity between orientation fields around corresponding singular points is computed and the maximum similarity is compared to a predefined threshold to determine if they should pass. To compare orientation field, they are first aligned according to singular points. Then the difference of orientations at a set of sampling points is computed and the mean value δ_o is obtained. The similarity is computed as $1 - \delta_o/(\pi/2)$.

5. Experiments

The experiments were conducted on NIST SD27 database which contains 258 latent fingerprints and their corresponding rolled prints. This is the only public domain database available containing mated latent and rolled prints. To show the advantage of multi-stage filtering, we expanded the background database by adding fingerprints from NIST SD4 and SD14. There are 2,000 and 27,000 different fingers in SD4 and SD14, respectively, with 2 rolled impressions per finger (one referred to as file print, the other referred to as search print). These fingerprints were also scanned from paper and have similar characteristics to the rolled prints in SD27. The file prints of all 2,000 fingers in SD4 and the first 8,000 fingers in SD14 are taken in addition to the 258 rolled prints in SD27 to form a background database containing 10,258 rolled prints.

We have conducted experiments showing the effect of the proposed multi-stage filtering algorithm on the minutiae-based latent fingerprint matching system in [4]. By setting thresholds on singularity filtering and orientation filtering as 0.25 and 0.6, our filtering algorithm obtains a penetration rate (P) of 39.0% at the accuracy (A) of 97.3%. The penetration rate measures the proportion of rolled prints, which are not filtered, to the whole background database. The accuracy measures the proportion of the corresponding rolled prints of latents, which are not filtered, to all latents. Ideally, accuracy should be 100% and penetration rate should be as small as possible. Since no performance on latent-to-roll filtering algorithms has been reported in the open literature, we mention the performance of one of the state-of-the-art roll-to-roll filtering algorithms [5], which reached P=40% and A=99% on NIST SD4. However, it should be noted that latent-to-roll filtering is much more difficult than roll-to-roll filtering. The Cumulative Match Characteristic (CMC) curves of our matching algorithm with and without filtering are shown in Figure 4. It can be observed that, although filtering makes identification rates at higher ranks drop, the rank-1 rate with filtering is better than that without filtering. As expected, the rank-1 rate (73.3%) for a background database of 10,258 rolled prints is lower than the corresponding number (86.4%) when the background database contains only 258 rolled prints in SD27 [4]. The filtering and matching algorithms have been implemented in C language and tested on a PC with Intel Core2 Duo CPU and Windows XP operating system. The average time for searching a latent against all the 10,258 rolled prints is reduced from 205 sec to 62 sec due to the proposed filtering algorithm.

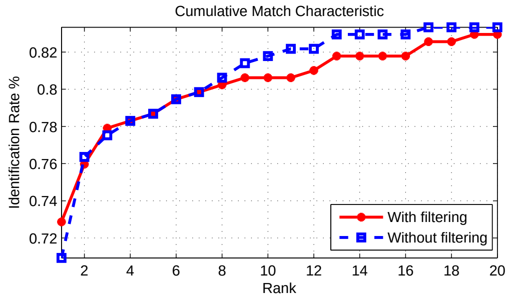


Figure 4. CMC curves of minutiae matching algorithm with and without filtering

6. Conclusions

A multi-stage filtering system, which utilizes information on pattern type, singular points and orientation field, is proposed in order to speed up a latent fingerprint matching system. The minutiae matching algorithm was performed only for the fingerprints passing all three filtering stages. Although latent fingerprints contain very limited information, our system reached a penetration rate of 39.0% at the accuracy of 97.3% in searching 258 latent fingerprints in NIST SD27 against a background database containing 10,258 rolled fingerprints. The identification accuracy is also improved by using the proposed filtering algorithm.

Our future work includes (i) improving singular point detection algorithm, (ii) adding other filtering techniques, like minutiae triplet-based technique, (iii) utilizing virtual singular points in latents, which are invisible in the region of interest but can be reliably predicted by human experts, and (iv) increasing the size of the background database to 200,000 rolled prints we received from law enforcement agencies.

References

[1] A. M. Bazen and S. H. Gerez. Systematic methods for the computation of the directional fields and singular points of fingerprints. *IEEE Trans. PAMI*, 24(7):905–919, 2002.

[2] S. C. Dass and A. K. Jain. Fingerprint classification using orientation field flow curves. In *ICVGIP*, pages 650–655, 2004.

[3] R. S. Germain, A. Califano, and S. Colville. Fingerprint matching using transformation parameter clustering. *IEEE Computational Science and Engineering*, 4(4):42–49, 1997.

[4] A. K. Jain, J. Feng, A. Nagar, and K. Nandakumar. On matching latent fingerprints. In *Workshop on Biometrics, CVPR*, June 2008.

[5] X. Jiang, M. Liu, and A. C. Kot. Fingerprint retrieval for identification. *IEEE Trans. on Information Forensics and Security*, 1(4):532–542, 2006.

[6] H. C. Lee and R. E. Gaensslen, editors. *Advances in Fingerprint Technology*. CRC Press, New York, 2001.

[7] P. Lo. Method and apparatus for adaptive hierarchical processing of print images. US Patent Application Publication No. 2007/0292005A1, 2007.

[8] K. Uchida, T. Kamei, M. Mizoguchi, and T. Temma. Fingerprint card classification with statistical feature integration. In *Proc. of ICPR*, pages 1833–1839, 1998.