Geometric-Aligned Cancelable Fingerprint Templates

Bian Yang[1], Christoph Busch[1], Mohammad Derawi[2], Patrick Bours[1], and Davrondzhon Gafurov[1]

[1] Norwegian Information Security Laboratory at Gjøvik University College, Teknologivegen 22, N-2815, Gjøvik, Norway
[2] Informatics and Mathematical Modelling at Technical University of Denmark, DK-2800 Kongens Lyngby, Denmark
{bian.yang,christoph.busch,patrick.bours, davrondzhon.gafurov}@hig.no

Abstract. A minutiae encryption algorithm based on geometric transformation of minutiae positions is proposed to generate cancelable fingerprint templates. A geometric transformation is used for alignment. A parameter-controlled minutiae encryption is performed within a local area to generate a cancelable minutiae template, and then all local encryption results are superimposed to form a protected template. Parameters to control the minutiae encryption are generated independent of the geometric-aligned minutiae, which ensures solid non-invertibility compared to those cancelable template generating algorithms with to-be-encrypted minutiae information as parameters.

Keywords: template protection, cancelable fingerprint template, fingerprint alignment, coordinate encryption.

1 Introduction

Biometric recognition is increasingly used as a strong security measure for authentication with the assumption that biometric characteristics are unique to a subject and cannot be forwarded to another individual. Thus a bypass of an existing security policy – that frequently happens with token or knowledge based authentication systems – can be avoided. However, from the view of data security and privacy, biometric templates are under potential compromise and therefore need careful protection, because biometric characteristics usually cannot be updated like normal passwords or PIN codes. Direct encryption of a biometric template by standard encryption algorithms (DES, AES, etc) is infeasible because the encrypted template needs decryption to invert to its plain-text for comparison. This happens during every verification process and is insecure, as full access to samples or unprotected biometric features are given to the potentially untrusted entity that conducts the comparison. It is better to run the comparison process in an encrypted domain to avoid decryption. However, standard encryption algorithms tolerate no fuzzy distortions inherent with biometric probes. Therefore, biometric template protection [1-19] was proposed, among which fingerprint template protection algorithms are intensively investigated [2,4-6,8-9, 13-15]. In general, there are three approaches to fingerprint template protection: the

P. Foggia, C. Sansone, and M. Vento (Eds.): ICIAP 2009, LNCS 5716, pp. 490–499, 2009.

first approach directly extracts lumination features from fingerprint raster images by image processing techniques [4-5]; the second approach complements the minutiae with additional biometric features (such as ridge context) [6,16] to enhance biometric performance or security; and the third approach [2,9,13-15] aims at protecting plain-text minutiae templates that are already generated conforming to ANSI or ISO standards. A serious security leakage [12] was found in the fuzzy vault [2]. Although fuzzy extractor [13] and secure sketch [14] can alleviate this problem, they sacrifice comparison performance in some degree. Biotokens proposed in [17] exhibits good performance by exploiting enlarged feature space with large template size.

Cancelable fingerprint templates [9] was proposed to distort minutiae data in a non-invertible way which generates diversified protected templates via setting transformation parameters. As the protection mechanism is non-invertible, there is no way to launch the key-inversion attack [12] on the protected template. Furthermore, the generated cancelable fingerprint templates are compliant in format to the original minutiae template, making the state-of-art minutiae comparators applicable.

However, cancelable fingerprint templates [9] assume that all minutiae data (position and angle) are pre-aligned. Automatic pre-alignment such as core detection works well for a majority of samples, however a failure-to-align rate of approximately 10% is not unlikely to occur [20]. This will subject the comparison performance to the pre-alignment accuracy. Regarding non-invertiblity, the transformation parameters used in [9] depend on the to-be-transformed minutiae data themselves, which will decrease the number of unknown factors, and thus brute-force searching to generate the same cancelable template will be more likely to succeed.

In this paper, we analyze the non-invertible transformation parameter setting issues in section 2. To tackle the alignment and parameter setting problems of the algorithm in [9], we propose in section 3 our solution with geometric alignment and strong non-invertibility even under the assumption that the transformation parameter set is public.

2 Parameter Setting for Non-invertible Transformation

In [9], a protected template can be easily canceled and renewed by setting different parameters. Cancelable templates by non-invertible transformation can provide solid computational complexity against template inversion by keeping the transformation parameters secret and coordinates' perturbation large enough. But the surface functions used in [9] to scramble the coordinates and angles significantly depend on those to-be-transformed coordinates themselves. This has two limitations:

(1) Comparison performance degradation - because the to-be-transformed coordinates and angles themselves have inherent minor distortion, taking these fuzzy-distorted minutiae data as transformation parameters will cause potential comparison performance degradation. In this sense, it is better to use transformation parameters independent of the minutiae data themselves to avoid distortion amplification;

(2) Security against template inversion - although surface folding in [9] can increase overlapped positions by increasing the coordinates' perturbation or times of transformation, it is easy to find an original point as single solution in a coordinate space for some transformed points (x,y), i.e., overlapped positions cannot cover the

full original minutiae space even after multiple non-invertible transformation. If the parameters are compromised or required to be public, the transformed points with single solution can be identified. If the number of such single-solution points exceeds a threshold, the original minutiae template is compromised. Two methods can be used to strengthen the non-invertibility: a.) superimposition of minutiae points in new coordinate systems (as in Section 3.3), which ensures the solution number roughly equal to superimposition layers; b.) employment of independent transformation parameters, which increases unknown factors other than to-be-encrypted coordinates themselves to increase the solution number (as in Section 3.2). These two methods will roughly ensure that each transformed point will have multiple original points as solutions. This is a stronger non-invertibility compared to the surface folding function in [9]. Although the transformation parameters should be independent of the to-be-transformed minutiae, they are not necessarily to be secret keys stored independently.

3 Proposed Algorithm to Generate Geometric-Aligned Cancelable Fingerprint Template

We propose a geometric-aligned minutiae template protection algorithm as shown in Fig.1, where *T* and *PT* are the original unprotected minutiae template and the protected template respectively. *S* is the randomly bit stream generated from a pseudo-random number generator (PRNG), which is used to form the random quantization table *QT* containing (*N*+1) quantization bins divided by *N* scales in a distance range [0, D] (assuming D > largest distance between a minutia point position and the core[1] point in the original template). $\{m_a\}$ are geometric-aligned minutiae points set within the R-radial local disk area centering each original minutia point. Decided by the distance d_c between each original minutia point and the core point, a quantized index d_k and the corresponding coordinate offset (dx_k, dy_k) are selected to modify all

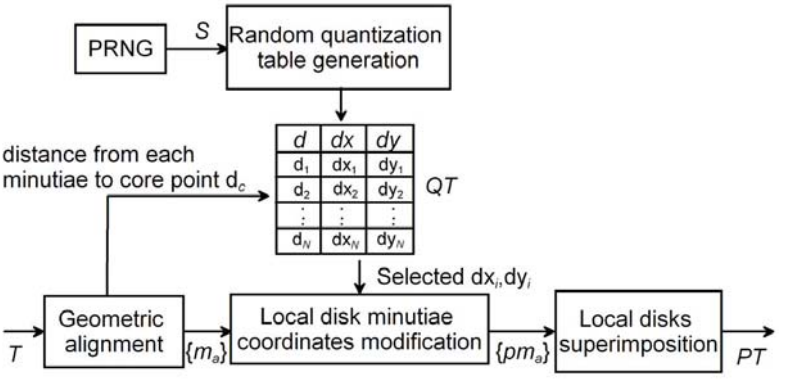
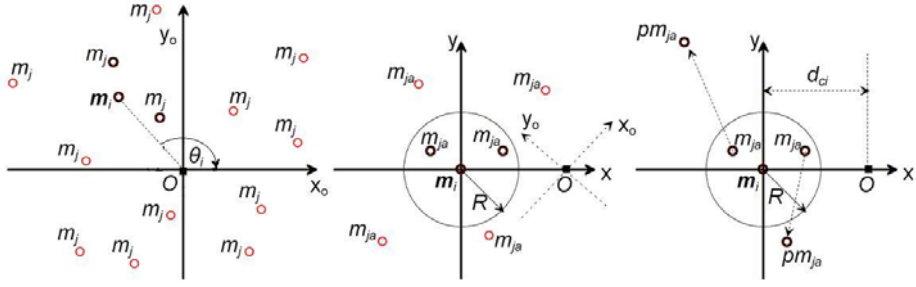


Fig. 1. Diagram of proposed minutiae template protection algorithm

[1] Core – singular point in the fingerprint, where the curvature of ridges reaches a maximum (according to ISO 19794-8).



(a) Original minutiae template (b) Geometric alignment to each minutia point m_i (c) Coordinates modification from m_{ja} to pm_{ja} as function of d_{ci}

Fig. 2. Geometric alignment and inside-disk-area minutiae coordinates' modification

minutiae coordinates within the local disk area. $\{pm_a\}$ are modified minutiae points set. After superimposition of all protected local disks, PT is obtained as the final protected template. We detail the whole process in Fig.1.

3.1 Geometric Alignment

Alignment is usually required for fingerprint template for both raster image [4] and minutiae coordinates [9]. Position and orientation of a core point can be detected as a reference point for alignment. However, the accuracy of core point detection remains to be a challenge, especially for the cases where multiple cores are found (e.g. twin loop fingerprint patterns). In such cases, we always choose the uppermost core as the reference point in our proposed algorithm. Another unstable factor is the core point's orientation, whose precision is differently defined by various standards and applications, such as $360^o/256$ in ISO [21] and $360^o/32$ in ANSI/NIST compatible software [22]. Unlike directly using core orientation in [4,9], we take only the position of a core as the reference for translation alignment of minutiae points but omit the detected orientation. Assuming there are totally M minutiae in the original template, each minutia $m_i(i=1,2,...,M)$ is set to be the new origin, and the line leading to the old origin O from m_i is set to be the new x-axis, and all other minutiae $m_j(i=1,2,...,M, j\neq i)$ can be correspondingly translation-and-rotation aligned to the new origin. Denote these $(M\text{-}1)$ geometric-aligned minutiae points as $\{m_{ai}\}(i=1,2,..,M\text{-}1)$, shown in Fig.2(a), where x_0-O-y_0 represents the old coordinate system. The alignment of $m_j(i=1,2,...,M, j\neq i)$ can be

$$\begin{bmatrix} m_{ja}(x) \\ m_{ja}(y) \end{bmatrix} = \begin{bmatrix} \cos(\theta_i - \pi) & \sin(\theta_i - \pi) \\ -\sin(\theta_i - \pi) & \cos(\theta_i - \pi) \end{bmatrix} \cdot \begin{bmatrix} m_j(x) - m_i(x) \\ m_j(y) - m_i(y) \end{bmatrix} \tag{1}$$

where $j=1,2,...,M$ but $j\neq i$, and $\angle m_i O x_0$ representing the angle decided by m_i's position in the old coordinate system x_0-O-y_0.

3.2 Minutiae Coordinates Random Modification in the Local Disk Area

After the minutiae points $m_j(j=1,2,...,M, j\neq i)$ geometric alignment to the minutia point m_i, some scrambling effect has already been achieved to hide the original positions of m_j in the original template. Due to this alignment operation to every minutia point $m_i(i=1,2,...,M)$ we can obtain M sets of aligned minutiae points $\{m_a\}_i\,(i=1,2,...,M)$. An intuitive idea can be to superimpose all the M sets $\{m_a\}_i\,(i=1,2,...,M)$ to form the final protected template, which is obviously a globally aligned version of the original template. However, this will dramatically increase the number of minutiae from M points in the original template to $M(M\text{-}1)$ points in the protected template. Assuming averagely 30~50 minutiae points in one original template, the number reaches 870~2450 in the protected template. If we assume at least Q minutiae points out of the total P minutiae points in the $H\times W$ sized template need to match their mates for a successful comparison, the probability of success for a brute-force search attack is $\binom{P}{Q}\Big/\binom{H\cdot W}{Q}$. Therefore a high density of resulting points (large value Q) definitely impacts a high probability for a false-match incident. In addition, a high number of resulting points will decrease the efficiency for compact template storage and also weakens the computational efficiency as the number of operational steps in minutiae template comparison increases. In order to reduce the total point number in the protected template, a local disk area with radius R can be defined to mask out those remote minutiae points (all m_{ja} with distance from m_i larger than R as shown in Fig.2.) and keep in the final protected template only those minutiae point inside the R-disk centering m_i.

Obviously, the kept minutiae points m_{ja} convey local topological information of the original template while the global topological information (local disks' orientation and distance to the old origin O) has been removed and thus secured by the previous geometric alignment step. To further secure the local topological relationship among the kept m_{ja} inside each local disk, the coordinates of all m_{ja} should be encrypted. For better comparison performance and stronger non-invertibility (as discussed in Section 2), we perturb m_{ja}'s coordinates by adding a random offset distance $\mathrm{d}x_k$ and $\mathrm{d}y_k$ $(k=1,2,...,N)$ to x- and y- coordinates of m_{ja} respectively, where $\mathrm{d}x_k$ and $\mathrm{d}y_k$ are decided by quantizing the distance d_{ci} (between m_i, the center of ith local disk, and O, the old origin, shown in Fig.2) with a pre-generated random quantization table QT consisting of N distance values: $\mathrm{d}_1,\mathrm{d}_1,...,\mathrm{d}_N$, where N is a pre-set parameter. For each m_{ja} inside the ith local disk, the coordinates' perturbation can be formulated as

$$\begin{cases} pm_{ja}(x) = m_{ja}(x) + \mathrm{d}x_K \\ pm_{ja}(y) = m_{ja}(y) + \mathrm{d}y_K \end{cases} \tag{2}$$

where $\mathrm{d}x_K$ and $\mathrm{d}y_K$ are randomly generated offset values stored as the indexed content by the Kth item d_K in QT (shown in Fig.1) and K is decided by

$$K = \arg\min_k |\,\mathrm{d}_k - QT(d_{ci})\,| \quad (k=1,2,...,N) \tag{3}$$

Unlike directly using minutiae points coordinates as function parameters to encrypt minutiae points themselves in [9], we use a translation and rotation invariant - the distance value d_{ci} as the parameter to control random offset values' selection for encryption of the kept points' coordinates inside the ith local disk. The dependency

between parameters and transformation input in [9] will definitely narrow the brute-force searching space and help find those security-concerned single solutions quickly. While in our algorithm, because d_{ci} is the removed global topological information of the original template and therefore independent of the local topological relationship among the kept points in the local disk, the single-solution security problem discussed in Session 2 can be effectively alleviated.

3.3 Local Disks Superimposition

In the above steps, all the minutiae points inside one disk are geometrically aligned to remove the global information of the disk's position in the original template, and then perturbed in coordinates to encrypt the local information. Now we superimpose all the M local disks with their perturbed points to obtain the final protected template. We consider only encryption of minutiae points' coordinates but not their orientation values because of their instability as discussed in Session 3.1.

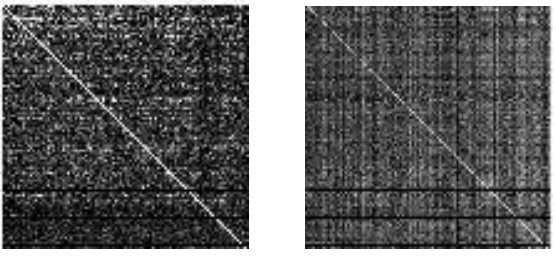
4 Experimental Results

The proposed algorithm was tested with the public fingerprint database FVC2002 DB2_A. This database contains totally 800 gray-level fingerprint images sized 560×296 collected from 100 fingers with 8 samples for each finger. VeriFinger 6.0 from Neurotechnology [23] was used to detect the uppermost core and all minutiae from fingerprint images. We manually adjusted a portion of detected cores with obvious large error in position, but these cases account for less than 10%. Minutiae points extracted from all the 800 images were processed by the proposed algorithm and then 800 protected templates were obtained. To test the comparison performance of the proposed algorithm, the former 7 out of the 8 protected templates for each finger were evaluated to choose the most reliable one as the final reference template, and the 8^{th} sample of each finger was used as a probe. For selecting the most reliable template, firstly we calculated the sum of distances (Hausdorff distance used here) between the minutiae set in one template and the 6 minutiae sets in the other 6 templates, and secondly we found the two templates with minimum distance sum values, and finally we selected the one with more minutiae points out of the two as the final reliable template. In this way 100 protected reference templates and 100 protected probes were obtained. To compare with the unprotected case, we tested the Neurotechnology comparator (called VeriFinger 6.0 Matcher) with the same database in which the first sample of all the fingers was used as the reference template and the 8^{th} samples as a probe. This generated performance results without template protection.

False-match-rate (FMR) and false-none-match-rate (FNMR) were employed to evaluate the biometric performance defined as follows

$$\mathrm{FMR} = \frac{number\ of\ accepted\ imposter\ attempts}{total\ number\ of\ imposter\ attempts} \tag{4}$$

$$\mathrm{FNMR} = \frac{number\ of\ \ rejected\ genuine\ attempts}{total\ number\ of\ genuine\ attempts} \tag{5}$$



(a)VeriFinger Matcher (b) Proposed algorithm

Fig. 3. Comparison scores matrix where brighter pixels indicate higher comparison scores. Vertical indices: probes and horizontal indices: reference templates.

where "accepted" and "rejected" are decided by thresholds calculated as the percentage of matched points in all the points in the protected template. A lower limit for number of matched points was also set to be 10 for a successful comparison.

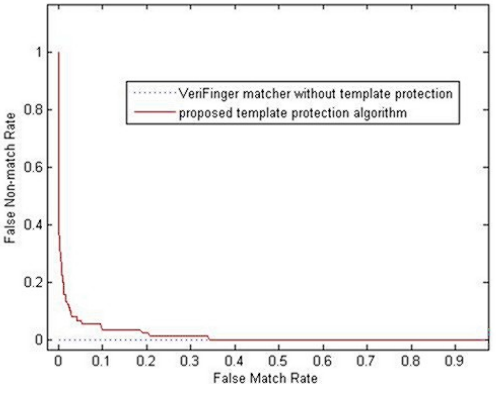


Fig. 4. Biometric performance (FNMR-FMR rate curve): VeriFinger Matcher without template protection (dotted-line) and the proposed template protection algorithm (solid-line)

In the experiments, we compare each of the 100 probes to all the 100 reference templates. The resulting comparison scores form a 256 gray-level (255 as highest and 0 as lowest) matrix in Fig.3, where (a) is the result from VeriFinger 6.0's matcher without template protection, and (b) our proposed algorithm with template protection. The vertical indices are the 100 probes ordered from top to bottom; the horizontal indices are the corresponding 100 templates ordered from left to right. The three dark horizontal lines in both Fig.3(a) and (b) are caused by the low number of detected minutiae in the probes (only 1,2, and 3 minutiae points detected in the 75^{th}, 86^{th} and 97^{th} probes) which were regarded as non-match with comparison score zero.

The parameters we used for experiments are: local disk radius $R = 30$; total number of QT indices $N = 30$; all the $N = 30$ coordinate offset pairs (dx_k, dy_k) $(k=1,2,...,N)$ were

randomly generated within the range [-100,100]. We use the simple Euclidean coordinate distance measure and assume that two points match with their distance < 8.

From the experimental results, out of the 100 fingers, 5 fingers had no core detected. We classified these 5 fingers as the case of failure to extract template, and excluded them in calculation of the FMR and FNMR. Another 5 probes had no core detected but their corresponding former 7 samples successfully contributed to a reference template. We excluded these 5 fail-to-extract-template probes from calculation of the FNMR, but included them for calculation of the FMR because they definitly contribute to the number of total imposter attempts. The biometric performance for both experiments are presented in Fig.4 by setting 500 thresholds in the normalized comparison scores range [0,1], where the dotted-curve is for the case by VeriFinger 6.0's Matcher without template protection and the solid-curve is for the case by the proposed template protection algorithm. From Fig.4 we can see that the result from our algorithm cannot reach the VeriFinger 6.0's result, which is probably due to VeriFinger Matcher's capability to exploit minutiae's orientation information, and its industry-level optimized comparator. We use Euclidean distance during verification and check the percentage of points' match in a protected probe template, and use this percentage value as the final comparison score. However, the biometric performance for protected minutiae templates indicates encouraging result and achieves highest Equal-Error-Rate (EER) of 0.0552 in our experiments. We suppose if a better comparator is employed, the comparison performance could be improved.

In our experiment, we generated 7 times of 100 potected templates and in each time different quantization tables were randomly generated. The EER values were slightly varied between 0.0552 and 0.0701.

5 Security Analysis

In our experiments, each transformed point's position could be originated from M possible local disks and each local disk could center on roughly 560×296 positions in an image, this equals to $M \cdot \binom{560\times 296}{M}$ possibilities (more than 560×296 in the full search case) to guess one true original minutiae point. So it provides roughly $(560\times 296)^Q = (560\times 296)^{10} \approx 2^{173}$ possibilities to guessing Q=10 true minutiae points' positions from the original template. This is a strong non-invertiblity against the key-inversion attack [12] which undermines fuzzy vault and fuzzy commitment. To achieve non-invertibility from the encrypted template $\{pm_{ja}\}$ to the transformed template $\{m_{ja}\}$ against cross-match attack on the transformed template level, it further provides roughly $N^Q=30^{10}\approx 2^{49}$ possibilities to guess Q=10 true transformed minutiae m_{ja}' positions even with the transformation parameters (quantization table QT) public. Keeping the transformation parameters (QT) secret can effectively thwart the template reconstruction attack in which probes can be forged from the protected template by exploiting a public QT without need of inversion to the genuine original template. However, we still notice there is possibility to exploit correlations in topological relationship between encrypted points to gain some linkability between protected templates, which is caused by the simple translation operations via Eq.(2). This could be solvable with some non-linear transformations to replace Eq.(2) to destroy the local topological relationships.

6 Conclusion and Future Work

We proposed in this paper a geometric-alignment based local minutiae encryption algorithm to protect minutiae-based finger template. The proposed algorithm can preserve good comparison performance while providing strong non-invertibility for security enhancement compared to non-invertible transformation based cancelable biometrics proposed in [9]. The strong non-invertibility is assured by the independent information collected from minutiae templates to control encryption parameters. Since the proposed algorithm does not need any other live-captured features other than the minutiae data for security enhancement, it is compatible to both standard minutiae extractors and comparators and thus can work on minutiae templates which are already generated in the existing databases. Our future work will focus on enhancing the unlinkability between protected templates by replacing the employed simple translation operation on the minutiae coordinates with non-linear transformations which we expect to destroy the topological relationship among all the encrypted points in the local disk.

Acknowledgement

This work is supported by funding under the Seventh Research Framework Programme of the European Union, Project TURBINE (ICT-2007-216339). This document has been created in the context of the TURBINE project. All information is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability. The European Commission has no liability in respect of this document, which is merely representing the authors' view.

Thanks to Julien Bringer from Sagem Sécurité and Koen Simoens from Katholieke Universiteit Leuven for their security comments.

References

1. Juels, A., Wattenberg, M.: A Fuzzy Commitment Scheme. In: Sixth ACM Conference on Computer and Communications Security, Singapore, pp. 28–36 (1999)
2. Juels, A., Sudan, M.: A Fuzzy Vault Scheme. In: IEEE Inter. Symp. on Information Theory, Lausanne, Switzerland (2002)
3. Savvides, M., Kumar, B.V.K.V.: Cancellable Biometric Filters for Face Recognition. In: IEEE Inter. Conf. on Pattern Recognition, Cambridge, UK, vol. 3, pp. 922–925 (2004)
4. Tuyls, P., Akkermans, A.H.M., Kevenaar, T.A.M., Schrijen, G.-J., Bazen, A.M., Veldhuis, R.N.J.: Practical Biometric Authentication with Template Protection. In: Inter. Conf. on Audio- and Video-Based Biometric Person Authentication, USA, pp. 436–446 (2005)
5. Sutcu, Y., Sencar, H.T., Memon, N.: A Secure Biometric Authentication Scheme Based on Robust Hashing. In: ACM Multimedia and Security Workshop, USA, pp. 111–116 (2005)
6. Nagar, A., Nandakumar, K., Jain, A.K.: Securing Fingerprint Template: Fuzzy Vault with Minutiae Descriptors. In: Inter. Conf. on Pattern Recognition, Tampa, Florida, USA (2008)

7. Teoh, A.B.J., Goh, A., Ngo, D.C.L.: Random Multispace Quantization as an Analytic Mechanism for BioHashing of Biometric and Random Identity Inputs. IEEE Trans. on Pattern Analysis and Machine Intelligence 28(12), 1892–1901 (2006)
8. GenKey: System, Portable Device and Method for Digital Authenticating, Crypting and Signing by Generating Short-Lived Cryptokeys. US Patent 2006/0198514A1 (2006)
9. Ratha, N.K., Chikkerur, S., Connell, J.H., Bolle, R.M.: Generating Cancelable Fingerprint Templates. IEEE Trans. on Pattern Analysis and Machine Intelligence 29(4), 561–572 (2007)
10. Kelkboom, E.J.C., Gkberk, B., Kevenaar, T.A.M., Akkermans, A.H.M., Van der Veen, M.: "3D Face": Biometric Template Protection for 3D Face Recognition. In: 2nd Inter. Conf. on Biometrics, Seoul, South Korea (2007)
11. Lee, Y.J., Bae, K., Lee, S.J., Park, K.R., Kim, J.: Biometric Key Binding: Fuzzy Vault Based on Iris Images. In: 2nd Inter. Conf. on Biometrics, Seoul, South Korea, pp. 800–808 (2007)
12. Scheirer, W.J., Boult, T.E.: Cracking Fuzzy Vaults and Biometric Encryption. In: Biometrics Symposium (2007)
13. Arakala, A., Jeffers, J., Horadam, K.J.: Fuzzy Extractors for Minutiae-Based Fingerprint Authentication. In: 2nd Inter. Conf. on Biometrics, Seoul, South Korea (2007)
14. Chang, E.C., Roy, S.: Robust Extraction of Secret Bits From Minutiae. In: 2nd Inter. Conf. on Biometrics, Seoul, South Korea (2007)
15. Yang, B., Busch, C., Bours, P., Gafurov, D.: Non-Invertible Geometrical Transformation for Fingerprint Minutiae Template Protection. In: 1st Inter. Workshop on Security and Communication Networks, Trondheim, Norway (2009)
16. Lee, C., Choi, J.Y., Toh, K.A., Lee, S., Kim, J.: Alignment-Free Cancelable Fingerprint Templates Based on Local Minutiae Information. IEEE Trans. on Systems, Man, and Cybernetics – Part B: Cybernetics 37(4), 980–992 (2007)
17. Boult, T.E., Scheirer, W.J., Woodworth, R.: Revocable Fingerprint Biotokens: Accuracy and Security Analysis. In: IEEE Inter. Conf. on Comput. Vis. & Patt. Recog., USA (2007)
18. Breebaart, J., Busch, C., Grave, J., Kindt, E.: A Reference Architecture for Biometric Template Protection Based on Pseudo Identities. In: BIOSIG 2008, GI-LNI (2008)
19. Delvaux, N., Chabanne, H., Bringer, J., Kindarji, B., Lindeberg, P., Mdgren, J., Breebaart, J., Akkermans, T., Van der Veen, M., Vedhuis, R., Kindt, E., Simoens, K., Busch, C., Bours, P., Gafurov, D., Yang, B., Stern, J., Rust, C., Cucinelli, B., Skepastianos, D.: Pseudo Identities Based on Fingerprint Characteristics. In: IEEE Inter. Conf. on Intelligent Information Hiding and Multimedia Signal Processing, pp. 1063–1068 (2008)
20. Bazen, A.M., Veldhuis, R.N.J.: Likelihood-Ratio-Based Biometric Verification. IEEE Trans. on Circuits and Systems for Video Technology 14(1), 86–94 (2004)
21. ISO Standard. Information Technology - Biometric Data Interchange Formats - Part 8: Finger Pattern Skeletal Data. ISO/IEC 19794-8 (2006)
22. NIST Software Document, `http://fingerprint.nist.gov/NBIS/nbis_non_export_control.pdf`
23. VeriFinger Software, `http://www.neurotechnology.com`