Correlation-resistant Fuzzy Vault for Fingerprints

Johannes Merkle[1], Moazzam Butt[2], Ulrike Korte[3], Christoph Busch[4]

Abstract: The fuzzy vault is one of the most popular biometric encryption schemes for protecting fingerprint data. However, its implementation faces two challenges: First, the fingerprints need to be aligned. Some publications have proposed the storage of auxiliary data to assist alignment, but these data may leak information about the biometric features. Secondly, the fuzzy vault is susceptible to attacks that correlate the data from two protected templates, which does not only violate the requirement of unlinkability but also allows the recovery of the biometric data.

In this work, we present a fuzzy vault construction for fingerprint data (minutiae) that addresses both issues. We do so by applying an absolute alignment method to the fingerprints, performing a quantization of the minutiae positions to a grid, and using all grid points unoccupied by minutiae as chaff. This approach results in all vaults containing the same set of points. In order to improve recognition performance, we also use the minutiae's angles and types. We present experimental evaluations and compare the results with the existing works on fuzzy fingerprint vault.

Keywords: Fuzzy Vault, Biometric Template Protection, Fingerprint Recognition

1 Introduction

Biometrics is increasingly used for secure identification throughout the world. However, the storage of biometric reference data raises privacy concerns and, thus, demands rigorous protection. While in password-based authentication, typically, one-way hash functions are used to protect the reference data of the user's passwords, this approach cannot be easily adopted for biometric data due to the noise inherent in its measurement. *Biometric template protection* aims to solve this issue by combining cryptography with error tolerance or error correction techniques: the biometric reference data are stored as protected templates, which preserve the privacy of the biometric information, but still allow biometric verification without the need to maintain secret keys for decryption.

For template protection, the following privacy properties are required [CS09, IS11]. Firstly, the protected templates prevent the recovery of the original biometric data unless a sufficiently similar feature data is provided for comparison (*Irreversibility*). Secondly, many different protected templates can be generated from the same biometric data (*Revocability*) so that it is impossible to link two protected templates generated from the same biometric data (*Unlinkability*).

[1] secunet Security Networks AG, Eschborn, Germany
[2] Fraunhofer-Institute for Computer Graphics Research IGD, Darmstadt, Germany
[3] Bundesamt für Sicherheit in der Informationstechnik, Bonn, Germany
[4] da/sec - Biometrics and Internet Security Research Group, Hochschule Darmstadt, Germany

One of the most popular template protection schemes is the fuzzy vault [JS02], which works on unordered feature sets and is, thus, considered particularly eligible for protecting data representing fingerprint minutiae (endings and bifurcations of the skin ridges). In the protected templates (*vaults*) generated by the fuzzy vault, the elements of the feature set are represented as finite field elements and then evaluated on a random secret polynomial of small degree; the resulting set of points on the polynomial's function curve represent a redundant encoding of the polynomial and ensure the error tolerance in case that some of the minutiae are not detected or if additional minutiae are measured during verification. Finally, these *genuine points* are hidden by adding a large number of random *chaff points* not lying on the polynomial's function curve.

However, there are two challenges for the application of the fuzzy vault to fingerprints: First, fingerprints are typically not aligned to each other, i.e., different captures of the same finger are rotated and shifted with respect to each other. Some publications propose to store auxiliary data (extracted from the fingerprint's orientation field) along with the vault to aid alignment [UJ06, NJP07, NNJ10], but this data may leak biometric information. Secondly, the fuzzy vault is susceptible to correlation attacks: Since the chaff points are chosen at random, the intersection of two vaults of the same fingerprint is likely to contain most of the genuine points, while most chaff points will be filtered out; this does not only allow the correlation of two vaults of the same subject (violating the unlinkability requirement) but can even allow recovery of the biometric data [SB07, KY08].

In this work, we present a fuzzy vault construction for fingerprints, which tackles both issues. Using the directed reference point estimation method from [Ta13a], we represent the minutiae's position and angle relatively to a coordinate system that can be robustly computed from the fingerprint. This approach is equivalent to a pre-alignment of the fingerprints [Ma09] and eliminates the need to store auxiliary alignment data. Furthermore, we quantize the minutiae positions to a grid and use all remaining grid points as chaff points. This implies that each vault contains the same set of points, which thwarts the aforementioned correlation attacks. Besides the minutiae's positions, we also use their angles and types, but since they may reveal too much information [CS09],[4] we use them to obscure the outputs of the secret polynomial. Furthermore, we store remainders of the minutiae angles (resulting from a modulo operation with a quantization parameter) to allow some error correction of the minutiae's angles.

Section 2 gives an overview of previous work. In Section 3, we describe our fuzzy fingerprint vault in detail and discuss its security in Section 4. In Section 5, we present the results of experimental evaluations, and compare the observed error rates and security estimates with that of comparable constructions. Finally, Section 6 gives a conclusion.

[4] Minutiae angles resemble the direction of the fingerprint's orientation field, and chaff points, in order to be indistinguishable from genuine points, would also need to have angles that are in accordance with it.

2 Previous Work

The first fuzzy fingerprint vault constructions in [CKL03] and [UPJ05] used only the locations of the minutiae, and assumed the fingerprints to be pre-aligned. In [UJ06], these ideas were improved by deploying an algorithm for fingerprint pre-alignment based on high curvature points as auxiliary data, and [NJP07] extended this construction by using also minutiae angles. The fuzzy fingerprint vaults of [Li08] and of [YV05] also used minutiae positions and angles, but deployed alternative pre-alignment methods using auxiliary data based on topological structures around the core and on a reliable reference minutiae, respectively. In contrast, the implementation of [Li10] used minutiae descriptors that are alignment-independent. The multi-finger fuzzy vault of [Me11] used minutiae positions and performed relative alignment using a minutiae matching algorithm.

All constructions mentioned so far are susceptible to the correlation attack that was sketched in [SB07] and practically implemented in [KY08]. The first fuzzy fingerprint vault immune to this attack was presented in [Ta13a]; there the fingerprints were pre-aligned by means of a directed reference point estimation (also used by our construction), the minutiae data (positions and angles) were quantized to a hexagonal grid, and all unoccupied grid points were used as chaff. A slight improvement (deploying the improved fuzzy vault scheme of [Do03]), together with a detailed analysis, was presented in [TMM15]. The implementation of [Ta15a] applies this construction to multiple fingerprints using Reed-Solomon list-decoding. In [Ta15b], another implementation of the improved fuzzy fingerprint vault resistant to correlation attacks was proposed that used quantized minutiae angles and positions, pre-aligned using the method of [Ta13a], and three different, alignment-independent minutiae descriptors.

3 Proposed Fuzzy Vault Construction

In this section, an approach is proposed to include minutiae angles and types in the fuzzy vault in a way that it is still hard to distinguish between genuine and chaff points. The use of minutiae angles and types will increase the amount of information (entropy) per minutiae, which results in higher security against brute force [MMT09] or false accept attacks [TMM15] and improves the recognition performance (as shown in Section 5).

One of the pre-requisite conditions for fuzzy vault decoding is the correct alignment of the query fingerprint minutiae template with the enrolment fingerprint minutiae template. In this work, the alignment has been performed on the fingerprint samples using the tented arch reference point estimation based on the orientation map around the core point [Ta13a]. After the alignment of fingerprint samples, the minutiae are extracted from these pre-aligned fingerprint samples. These minutiae templates are used as enrolment and query minutiae templates in the fuzzy vault encoding (enrolment) and decoding (verification) process, as explained in the following sections. The parameters used have been determined empirically, as explained in Section 5.

The processes of enrolment and verification are shown in Figure 1.

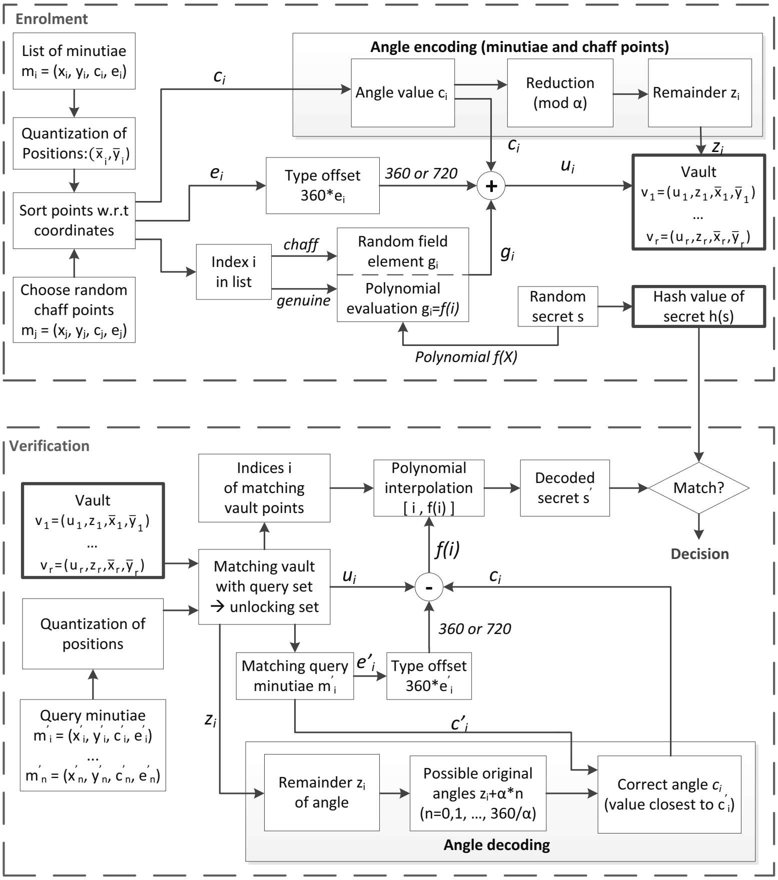
Enrolment
List of minutiae m_i = (x_i, y_i, c_i, e_i)
Quantization of Positions:(x̄_i,ȳ_i)
Sort points w.r.t coordinates
Choose random chaff points m_j = (x_j, y_j, c_j, e_j)
Angle encoding (minutiae and chaff points)
c_i
Angle value c_i
Reduction (mod α)
Remainder z_i
c_i
z_i
e_i
Type offset 360*e_i
360 or 720
+
u_i
Vault v_1=(u_1,z_1,x̄_1,ȳ_1) … v_r=(u_r,z_r,x̄_r,ȳ_r)
Index i in list
chaff
genuine
Random field element g_i
Polynomial evaluation g_i=f(i)
g_i
Random secret s
Hash value of secret h(s)
Polynomial f(X)
Verification
Vault v_1=(u_1,z_1,x̄_1,ȳ_1) … v_r=(u_r,z_r,x̄_r,ȳ_r)
Quantization of positions
Query minutiae m′_i = (x′_i, y′_i, c′_i, e′_i) … m′_n = (x′_n, y′_n, c′_n, e′_n)
Indices i of matching vault points
Polynomial interpolation [i , f(i)]
Decoded secret s′
Match?
Decision
Matching vault with query set → unlocking set
f(i)
u_i
-
c_i
360 or 720
Matching query minutiae m′_i
e′_i
Type offset 360*e′_i
z_i
c′_i
Remainder z_i of angle
Possible original angles z_i+α*n (n=0,1, …, 360/α)
Correct angle c_i (value closest to c′_i)
Angle decoding

Fig. 1: Enrolment and verification process

3.1 Enrolment

Let the minutiae in the enrolment template be represented as $m_i = (x_i, y_i, c_i, e_i)$, where (x_i, y_i) are the Cartesian coordinates of the position, c_i is the orientation (measured by the angle with the horizontal axis) and e_i the type of the minutia. The minutiae type $e_i = 1$ represents a ridge ending and $e_i = 2$ a ridge bifurcation; minutiae of unknown types $(e_i = 0)$ are neglected in our scheme.[5] Furthermore, we discard all minutiae lying outside a region of interest defined by an ellipsis area having semi-minor and semi-major axis lengths (a, b) (see Figure 2).

[5] While further minutiae types can be distinguished (see [Ma09]), commercial minutiae detection algorithms only output these three types as specified by ISO/IEC 19794-2.

The positions (x_i, y_i) of the remaining minutiae points are then quantized according to a rectangular grid with bin spacing q_x and q_y (determined parameters empirically), respectively, as shown in Figure 2, so that the quantized values $(\bar{x}_i, \bar{y}_i)$ are the centers of the corresponding cell in which the minutiae is located. In cases, where two or more minutiae are found in the same cell, the data (quantized position, angle and type) of the minutia having the highest quality value (output by the minutiae extractor) are taken and those of the others are discarded.

From the remaining quantized minutiae, those t having highest minutiae quality value (calculated by the minutiae extractor) are selected for the enrolment process. If less than t quantized minutiae are left over, all of them are chosen.

In order to obscure this set of (at most t) minutiae, all unoccupied (by minutiae) cells of the quantization grid within the ellipsis are added in the minutiae list [Ta13b] as chaff points. Angle and type of these chaff points are randomly generated.

The joined set of r points, comprising the (at most t) quantized minutiae (genuine points) and the chaff points, are sorted with respect to their coordinates. The total number r of points equals the number of grid cells in the region of interest. To protect the vault, a random polynomial f of degree k over a finite field $\mathbf{F}_p$ of prime order $p \geq r$ is chosen; the concatenation of the $k+1$ coefficients represents the binary secret s that must be recovered during verification. For each genuine point, its index i in the sorted list of vault points, represented as finite field element, is evaluated over the polynomial f; the resulting value $f(i)$ is then added to the minutia's angle value c_i and an offset value depending on the minutia's type e_i, resulting in an ordinate value $u_i = f(i) + c_i + 360 \cdot e_i$. For each chaff point, the value u_i is simply chosen as a random finite field element g_i, added to a random angle c_i and an offset $360 \cdot e_i$ with random $e_i \in \{1, 2\}$.

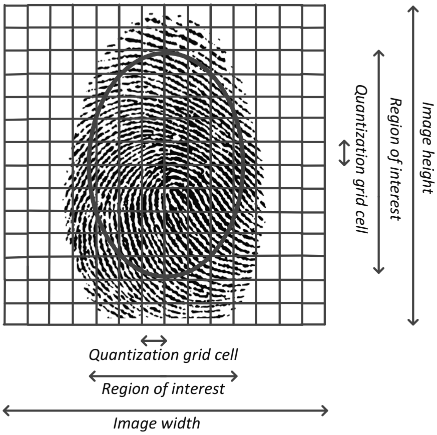

Fig. 2: Fingerprint region of interest and quantization grid

In order to minimize verification errors due to inaccurately measured minutiae angles, we store, for each genuine point, the remainder c_i mod α, where parameter α is a divisor of 360. For chaff points, the remainder $z_i = c_i$ mod α of a randomly chosen angle c_i is stored.

The resulting vault is the list of r points, containing (at most t) genuine points, each of which comprising the values $(u_i, z_i, \bar{x}_i, \bar{y}_i)$. In addition, a hash value $h(s)$ of the secret s is stored in the vault.

3.2 Verification

For verification, a query minutiae template is acquired containing a number of minutiae, each of which represented as $m'_j = (x'_j, y'_j, c'_j, e'_j)$, as described in Section 3.1. As during enrolment, minutiae having unknown type and minutiae located outside the region of interest defined by an ellipsis with semi-minor and semi-major axis lengths (a,b) (see Figure 2) are excluded. The positions of the remaining minutiae are quantized according to a rectangular grid with bin spacing q_x and q_y as shown in Figure 2; the quantized value $\bar{x}'_j, \bar{y}'_j$ is the center of the cell in which the minutia lies. In case two or more minutiae are quantized to the same position, the data (angle and type) of that one having the highest quality value (output by the minutiae extractor) are taken and that of the others are discarded.

Then, for each quantized minutiae position $(\bar{x}'_j, \bar{y}'_j)$ in the query template, the matching point $(u_i, z_i, \bar{x}_i, \bar{y}_i)$ with $(\bar{x}_i, \bar{y}_i) = (\bar{x}'_j, \bar{y}'_j)$ in the vault is identified and added to the list of candidates, referred to as unlocking set. (Since the vault contains all quantized positions in the region of interest, such a point always exists). Then, for each candidate point, the angle c'_j of the mating query minutiae is used to (try to) recover the original angle value c_i as the angle $\bar{c}_i$ that is closest to c'_j and equals z_i up to a multiple of α, i.e., as

$$\bar{c}_i = \text{argmin}_{\beta \in [0,359]} (\Delta(c'_j, \beta) \,|\, \beta = z_i \bmod \alpha),$$

where Δ denotes the distance of two angles along the unit circle. If $\Delta(c_i, c'_j) < \alpha/2$, i.e., if the angle of the query minutiae deviates from the original angle of the mating vault point by less than $\alpha/2$, the original angle c_i is correctly recovered.

For each point in the unlocking set, the recovered angle $\bar{c}_i$ is used to compute an abszissa value $f_i = u_i - \bar{c}_i - e'_i \cdot 360$. If a point is a genuine point, its angle has been correctly recovered, and if its minutia type e_i equals the type e'_j of the mating query minutiae, we obtain $f_i = f(i)$, where i is the index of the point in the vault. However, for chaff points, or if its angle is incorrectly recovered, or if its type deviates from that of the mating query minutia, most likely $f_i \neq f(i)$.[6]

For each combination of $k+1$ points (i, f_i), a supporting polynomial is determined by Lagrange interpolation and checked for correctness by comparing the hash value of its concatenated coefficients with the hash value of the original secret s stored in the vault. If a polynomial is found, where the hash matches, the verification is successful.

[6] Since f_i is the sum of three random variables, there is a small chance that it matches $f(i)$ by incident.

4 Security

A very important aspect of a fuzzy vault implementation is its resistance to recovery attacks, i.e., the effort for an attacker given a vault record to recover the original feature sets or, equivalently, the secret polynomial. When estimating the security of our fuzzy fingerprint vault, we count the expected number of polynomial decoding attempts. This estimate is conservative insofar as we neglect the computational effort for each attempt.

Generally, the fuzzy vault can be attacked by a *brute-force attack*, where the attacker repeatedly samples k from the vault and tries to interpolate the secret polynomial from these. The expected number of attempts until k correct points are chosen can be estimated as $\binom{r}{k}/\binom{t}{k}$ [MMT09]. However, in our implementation, the attacker also has to guess, for all chosen points, the corresponding minutiae angles c_i and types e_i from the given remainders $z_i = c_i \bmod \alpha$, resulting in an additional factor of $(2 \cdot 360/\alpha)^k$.

The above attack can be improved by exploiting the statistical distribution of the feature data. Precisely, the attacker can use estimations for the distributions of minutiae positions and angles, e.g., obtained from public fingerprint databases, to rank the vault points according to the probability of occurrence of the corresponding feature data, and sample the points for polynomial interpolation only from the, say w, top ranked points. Furthermore, instead of guessing random values for the minutiae angles (matching the given remainders z_i) and types, the attacker can choose the most likely values. In order to analyze the success probability of this *statistical attack*, we empirically determined the distribution of the feature data stored by our fuzzy vault, i.e., the distribution of the minutiae positions quantized to the grid and of the remainder (modulo α) of minutiae angles at each grid point, computed from over 130.000 minutiae extracted with the FingerJetFX algorithm from 2500 fingerprints from the MCYT-100 database [OGFAS03]. Using this distribution, we determined, for each grid point, the predominant α-rounded minutiae angles $c - (c \bmod \alpha)$ and minutiae types as the attacker's guess for that position. Then, we estimated, for different $w \geq k$, the probability $P(w)$ that, for a randomly generated vault record, the top ranked (according to the distribution of the minutiae data) w vault points contain at least k genuine points, for which both the minutia's α-rounded angle and type assume the most predominant value. Using these estimates, the expected number of attempts until the attacker is successful can be estimated as $P(w)^{-1}\binom{w}{k}$, minimized over w.

In contrast, the *false-accept attack* exploits the specific distribution of the biometric features simply by simulating repeated (impostor) verification attempts using the features of randomly chosen (real) fingerprints, e.g., chosen from a biometric database [TMM15]. The success probability of the false-accept attack is equal to the False Match Rate (FMR) for the parameters used. In general, the attacker can deviate in her simulation from the parameters used in actual operation to optimize her success rate; however, in our fuzzy vault implementation, the number n of decoding iterations is the only parameter that is not already fixed in the enrolment. It has been proven in [TMM15] that the expected number of decoding attempts of the false-accept attack is minimized for $n = 1$. Hence, we estimate the security against false-accept attacks using this optimal strategy.

Estimating very high security levels assumes sharp estimations of FMRs when they are close to zero. In biometric systems with deterministic verification algorithm, the FMR can only be estimated down to the magnitude of $1/N$, where N is the number of impostor verifications performed in the evaluation. However, the verification of our implementation is probabilistic as soon as the unlocking set contains more than k points. This property allows us to give heuristic estimates of FMRs that are much smaller than $1/N$: For each single impostor verification, we compute the success probability based on the size of the unlocking set and the number of correct points contained, and, finally, we estimate the FMR as the mean over all verifications. For details, we refer to [TMM15].

Another very important security aspect concerns the risk of correlation attacks on two or more vault records of the same subject. The correlation attack of [SB07, KY08] exploits that a correlation of the points from two vault records of the same biometric instance (fingerprint) will mostly result in genuine points. However, since we use all grid points unoccupied by minutiae as chaff points, all vault records contain the same set of quantized minutiae positions, namely, all grid points. This implies that a correlation between vault records on the basis of the quantized minutiae positions will always yield all vault points and, hence, gives no information. On the other hand, the attacker may try to perform correlation on the basis of the remainders (modulo α) of the minutiae angles stored in all vault points. However, for the optimal value $\alpha = 30$ (see 5), these remainders contain a lot of noise and only limited information, i.e., have a poor signal-to-noise-ratio. Nevertheless, we acknowledge that a correlation attack based on correlation of the angle's remainders must be taken into account and leave this aspect, as well as potential countermeasures, e.g. choosing the remainders for chaff points according to a more realistic distribution, to future investigations.

5 Experiments

In order to determine optimal parameters and the corresponding error rates, we performed experiments using the MCYT-100 database [OGFAS03]. Precisely, we used 1200 optical scans of the right index fingers captured with a Digital Persona UareU sensor from 100 subjects; these fingerprints have an image size of 256x400 pixels and a resolution of 500 dpi. Feature extraction was performed using the FingerJetFX minutiae extractor [FJF11]. For the determination of the False Non-Match Rate (FNMR), we enrolled every fingerprint and, for each vault record, performed verification attempts with all other fingerprints of the same finger, resulting in up to $100 \cdot 12 \cdot 11 = 13.200$ genuine comparisons. In order to estimate the FMR, we enrolled 6 fingerprints per subject and conducted 99 impostor verifications (with distinct subjects) per vault record, resulting in up to $100 \cdot 6 \cdot 99 = 59.400$ impostor comparisons.

We found the best trade-off (for variable k) between the FNMR and the security against recovery attacks for parameters $t = 35$, $q_x = 15$, $q_y = 20$ and $\alpha = 30$, which results in a vault size $r = 651$. For these parameters and varying k, Table 1 summarizes the observed error rates and the security (in bits) against the brute-force attack (SEC_{BF}), statistical attack (SEC_{ST}) and false-accept attack (SEC_{FA}), estimated as described in Section 4. The Failureto-enrol rate (FTE) was zero as no failures occurred in feature (minutiae) extraction and our implementation does not impose any quality requirements, e.g. a minimum number of minutiae, on the feature set.

Tab. 1: Error rates and security against relevant attacks of the scheme.

k	FNMR	FMR	SEC_{BF}	SEC_{ST}	SEC_{FA}
4	6.2%	1.56%	35.5	17.5	19.6
5	8.2%	0.16%	44.4	23.9	24.0
6	12.2%	0.02%	53.4	29.5	28.1
7	18.7%	0%	62.5	37.6	32.1

It turned out that the statistical attack is most efficient for security levels up to 24 bits, while, for higher security, the false-accept attack performs better. However, we stress that the efficiency of the statistical attack may be reduced by choosing the (angle's) remainders for chaff points according to a more realistic distribution. The effectiveness of that countermeasure and its impact to the error rates and security against false accept attacks remain to be investigated.

Tab. 2: Error rates and security against relevant attacks of the scheme without using minutiae types.

k	FNMR	FMR	SEC_{BF}	SEC_{ST}	SEC_{FA}
5	6.0%	1.9%	39.4	16.5	19.8
6	7.4%	0.3%	47.4	20.1	23.4
7	9.6%	0.04%	55.5	24.2	27.1
8	13.1%	0%	63.6	27.3	31.0

We also evaluated the error rates and security for the case that minutiae types are discarded (see Table 2). For this reduced implementation (and using accordingly adapted estimates for attacks), we observed a comparable recognition performance in terms of relation between FNMR and FMR and even a slightly better FMR for a given security level against the false-accept attack, which indicates that the recognition of types is too unreliable to improve the error rates. However, the statistical attack becomes more efficient and beats the false-accept attack for all security levels. Therefore, we conclude that, by discarding the minutiae types, the overall security, i.e., security against the most efficient attack, is slightly decreased for reasonably limited FNMR.

Finally, we compare our scheme with the fuzzy fingerprint vault of [Me11], which uses minutiae positions (without quantization), the multi-finger fuzzy vault of [TMM15], which uses minutiae positions quantized to a hexagonal grid, and the fuzzy vault presented in [Ta15b], which combines quantized minutiae positions (in polar coordinates) and angles (but no types) with three different local descriptors of the minutiae. Our scheme as well as those from [TMM15] and [Ta15b] rely on the directed reference point estimation method from [Ta13a] to address the alignment of minutiae positions and angles.

It is important to note that the scheme of [Me11] is vulnerable to the correlation and recovery attack described in [SB07, KY08], while the other schemes are resistant against this attack.

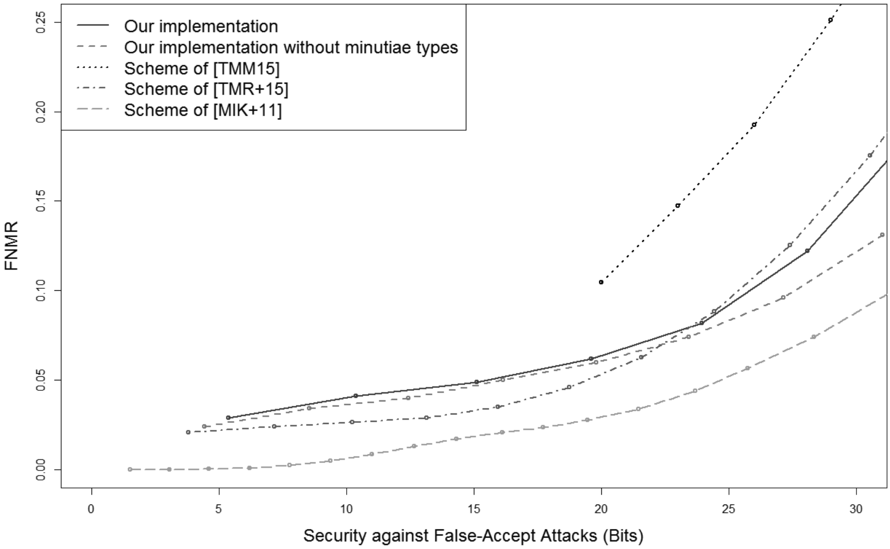

Fig. 3: Comparison of our fuzzy fingerprint vault implementation with those of [Me11], [TMM15] and [Ta15b] with respect to FNMR and security against false-accept attacks.

Figure 3 shows plots of the FNMR versus the security against false-accept attacks (depending on k) of the security and FNMR of our implementation, with and without using minutiae types, and for the schemes of [Me11], [TMM15] (in the single-finger setting) and [Ta15b]. For [Me11] and [Ta15b], we evaluated the error rates using the original implementations; in the latter scheme, we used the parameters reported as optimal in [Ta15b], while, for the former, we determined optimal parameters for an FTE below 5% in the single-finger setting.[7] Since the implementation of [TMM15] was not available to us, we used the results reported therein.

The comparison shows that our scheme achieves a much lower FNMR at the same security level than that of [TMM15]. Since both schemes apply the same pre-alignment and perform a quantization of the minutia positions, we attribute this improvement to the incorporation of minutia angles (and types), which were not used in [TMM15]; in fact, we observed error rates similar to that of [TMM15] when using only the positions of the minutiae.

When comparing our scheme to that of [Ta15b], our scheme has a higher FNMR for security levels under 23 bits, but performs considerably better for higher security levels, in particular, if minutiae types are not used. Both schemes have a significant higher FNMR

[7] In [Me11], two or more fingers were used.

than the implementation of [Me11]; we believe that the higher FNMR is a consequence of the quantization of the feature data, the use of all unoccupied points as chaff,[8] and the absolute pre-alignment. In particular, the reference point estimation of [Ta13a], which is used for absolute alignment in our scheme and in that of Tams et al., fails and gives highly inaccurate reference points for a small fraction of fingerprints, which explains the 'baseline' FNMR of $2\% - 2.5\%$ (even for $k = 1$) in both schemes. On the other hand, quantization and absolute pre-alignment are needed to achieve resistance against correlation attacks and, thus, we conclude that correlation-resistance comes at the price of reduced recognition accuracy.

6 Conclusion

In this paper, a step towards a correlation-resistant fuzzy vault is proposed that is based on the usage of all points of a quantization grid to achieve correlation-resistance. The fuzzy vault construction in this work also employs minutiae angles and types, in addition to minutiae positions. The results show that the use of more minutiae level information (positions, angles and types) have led to lower error rates and also higher security against brute force or false accept attacks. It turns out that minutiae types slightly deteriorate the results. The comparison of the results with previous works show the validity of the concept.

References

[CKL03] Clancy, T. Charles; Kiyavash, Negar; Lin, Dennis J.: Secure Smartcard-Based Fingerprint Authentication. In: Proc. ACM SIGMM workshop on Biometrics methods and applications. ACM, New York, NY, USA, pp. 45–52, 2003.

[CS09] Cavoukian, Ann; Stoianov, Alex: Biometrics: theory, methods, and applications. John Wiley & Sons, Inc., Hoboken, NJ, USA, chapter 26 - Biometric Encryption: The New Breed of Untraceable Biometrics, 2009.

[Do03] Dodis, Yevgeniy; Ostrovsky, Rafail; Reyzin, Leonid; Smith, Adam: Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data. IACR Cryptology ePrint Archive, 2003:235, 2003.

[FJF11] FingerJetFX OSE – Fingerprint Feature Extractor, Open Source Edition. README file, 2011.

[IS11] ISO/IEC 24745:2011: , Information Technology – Security Techniques – Biometric Information Protection. International Organization for Standardization, 2011.

[JS02] Juels, A.; Sudan, M.: A Fuzzy Vault Scheme. In: Proc. Int. Symp. Inf. Theory. p. 408, 2002.

[KY08] Kholmatov, A.; Yanikoglu, B.: Realization of Correlation Attack Against the Fuzzy Vault Scheme. In: Proc. SPIE. volume 6819, 2008.

[8] For the scheme of [Me11], we used 85 chaff points, and their vicinities (where query minutiae are considered to match the point) cover approximately 20% of the relevant fingerprint area.

[Li08] Li, Jianjie; Yang, Xin; Tian, Jie; Shi, Peng; Li, Peng: Topological structure-based alignment for fingerprint Fuzzy Vault. In: Proc. Int. Conf. on Pattern Recognition. pp. 1–4, 2008.

[Li10] Li, P.; Yang, X.; Cao, K.; Tao, X.; Wang, R.; Tian, J.: An alignment-free fingerprint cryptosystem based on fuzzy vault scheme. J. Netw. Comput. Appl., 33:207–220, 2010.

[Ma09] Maltoni, D.; Maio, D.; Jain, A.K.; Prabhakar, S.: Handbook of Fingerprint Recognition. Springer Publishing Company, Incorporated, 2nd edition, 2009.

[Me11] Merkle, J.; Ihmor, H.; Korte, U.; Niesing, M.; Schwaiger, M.: Performance of the Fuzzy Vault for Multiple Fingerprints. In: Proc. BIOSIG 2011. volume 191 of LNI. GI, pp. 57–72, 2011.

[MMT09] Mihăilescu, Preda; Munk, Axel; Tams, Benjamin: The Fuzzy Vault for Fingerprints is Vulnerable to Brute Force Attack. In: Proc. of BIOSIG 2009. volume 155 of LNI. GI, pp. 43–54, 2009.

[NJP07] Nandakumar, K.; Jain, A. K.; Pankanti, S.: Fingerprint-Based Fuzzy Vault: Implementation and Performance. IEEE Transactions on Information Forensics and Security, 2(4):744–757, 2007.

[NNJ10] Nagar, A.; Nandakumar, K.; Jain, A. K.: A hybrid biometric cryptosystem for securing fingerprint minutiae templates. Pattern Recogn. Lett., 31:733–741, June 2010.

[OGFAS03] Ortega-Garcia, J.; Fierrez-Aguilar, J.; Simon *et al.*, D.: MCYT baseline corpus: a bimodal biometric database. IEE Proc. on Vision, Image and Signal Processing, 150(6):395–401, 2003.

[SB07] Scheirer, W. J.; Boult, T. E.: Cracking Fuzzy Vaults and Biometric Encryption. In: Proc. of Biometrics Symp. pp. 1–6, 2007.

[Ta13a] Tams, Benjamin: Absolute Fingerprint Pre-Alignment in Minutiae-Based Cryptosystems. In: Proc. of BIOSIG 2013. volume 212 of LNI. GI, pp. 75–86, 2013.

[Ta13b] Tams, Benjamin: Attacks and Countermeasures in Fingerprint Based Biometric Cryptosystems. CoRR, abs/1304.7386, 2013.

[Ta15a] Tams, B.: Unlinkable Minutiae-Based Fuzzy Vault for Multiple Fingerprints. IET Biometrics, 2015. to appear.

[Ta15b] Tams, Benjamin; Merkle, Johannes; Rathgeb, Christian; Wagner, Johannes; Korte, Ulrike; Busch, Christoph: Improved Fuzzy Vault Scheme for Alignment-Free Fingerprint Features. In: Proc. of BIOSIG 2015. volume 245 of LNI. GI, 2015.

[TMM15] Tams, B.; Mihăilescu, P.; Munk, A.: Security Considerations in Minutiae-based Fuzzy Vaults. IEEE Trans. Inf. Forensics Security, 10(5):985–998, 2015.

[UJ06] Uludag, Umut; Jain, Anil K.: Securing fingerprint template: fuzzy vault with helper data. In: Proc. Workshop on Privacy Research In Vision. pp. 163–169, 2006.

[UPJ05] Uludag, Umut; Pankanti, Sharath; Jain, Anil K.: Fuzzy vault for fingerprints. In: Proc. Int. Conf. on Audio- and Video-Based Biometric Person Authentication. pp. 310–319, 2005.

[YV05] Yang, Shenglin; Verbaudwhede, Ingrid: Automatic secure fingerprint verification system based on fuzzy vault scheme. In: Proc. Int. Conf. on Acoustics, Speech and Signal Processing. pp. 609–612, 2005.