Security Capacity of the Fuzzy Fingerprint Vault

Johannes Merkle, Matthias Niesing, Michael Schwaiger
secunet Security Networks AG
D-45128 Essen, Germany
johannes.merkle@secunet.com,
matthias.niesing@secunet.com
michael.schwaiger@secunet.com

Heinrich Ihmor, Ulrike Korte
Bundesamt für Sicherheit in der Informationstechnik
D-53175 Bonn, Germany
heinrich.ihmor@bsi.bund.de
ulrike.korte@bsi.bund.de

***Abstract*—We investigate the security of a privacy enhancing technique for fingerprint authentication known as *fuzzy fingerprint vault*. This technique is based on the *fuzzy vault* of Jules and Sudan, a scheme that allows error tolerant authentication, while preserving the privacy of the reference data. We explore if and under what circumstances a secure fuzzy fingerprint vault can be implemented. We derive both upper and lower security bounds for any attacks that attempt to recover the template from the stored reference data, and, at the same time, significantly improve the best known attack. Furthermore, we show how to select optimal parameters and evaluate both minimum minutiae match rates and minimum number of minutiae needed to obtain an appropriate security level. Our results quantify the security capacity of the fuzzy fingerprint vault and provide important tools for selection of suitable parameters.**

***Keywords*-biometric template protection; fingerprint; fuzzy vault; polynomial reconstruction**

I. Introduction

Without any doubt, fingerprints are the biometric traits most widely deployed for authentication. However, the storage of biometric reference data introduces considerable risks for biometric authentication systems and raises serious concerns regarding privacy and data protection. One of the most prominent solutions to solve this issue is the fuzzy fingerprint vault, which allows error tolerant fingerprint authentication while preserving the privacy of the biometric features [1]. It belongs to the class of *biometric template protection* techniques [2], and is based on the fuzzy vault scheme [3] of Juels and Sudan, which applies Reed-Solomon decoding to redundantly bind the biometric template to a randomly selected secret polynomial.

Fingerprint authentication is typically based on minutiae, which are specific features of the fingerprint pattern. The variety and extent of errors in minutiae measurements, particularly, frequent insertions, omissions and re-ordering of the measured minutiae, pose a considerable challenge to template protection schemes [4]. The fuzzy vault is able to tolerate such errors and, hence, is particularly interesting for minutiae-based authentication.

Several publications [5][6][7][8][9][10] report successful implementation of the fuzzy vault scheme based on minutiae. However, the subsequent publication of efficient attacks [11][12] demonstrates that the parameters proposed do not provide adequate security.

For the fuzzy vault, theoretical results are known, from which rigid security estimates could be deduced. In particular, Dodis, et al. [13] proved upper bounds for the information leakage by the stored data, which determines the maximum success probability of an attack trying to guess the template or the key from the stored reference data, see Section IV-B for details. In addition, an attacker's success probability depends on the original entropy of the biometric feature vector - or, equivalently, its redundancy. Therefore, a realistic estimation of the entropy of the biometric feature vector is a key aspect for a sound security analysis.

On the other hand, these provable lower security bounds are not sharp. Firstly, these bounds only estimate the success probability of attacks and do not consider the effort required for each trial. Secondly, the proof techniques used in [13] overestimate the information leakage. Achieving provable security may be a very appealing objective, but it is also interesting to determine how secure the scheme is in practice.

In this publication, we explore if and under what circumstances a fuzzy fingerprint vault can be secure with respect to both provable security and real attacks. In particular, we generalize the bounds of [13] to the case where the minutiae and chaff points are chosen with a minimum distance to reduce false matchings, and also give an exact estimate for the entropy of a feature vector consisting of minutiae location data. On the other hand, we estimate the effort required for practical attack methods and present an improvement of the best known attack. Then, we show, how the parameters can be optimized and determine minimum minutiae match rates with respect to both provable security and practical security.

This article is structured as follows. In Section III, we give a description of the scheme. In Section IV, we conduct a theoretical analysis of its security and error robustness both with respect to information theoretical results and practical attacks. Section V presents methods for parameter optimization with respect to the deduced security bounds, and Section VI provides results using empirical data. A conclusion is given in Section VII.

II. Background

The fuzzy fingerprint vault is one of many template protection techniques that have been proposed in the literature, for instance, the *Biometric Encryption* scheme by Soutar et. al. [14], *Cancelable Biometrics* by Ratha et. al. [15], robust bit extraction schemes based on quantization, e.g. of Linnartz and Tuyls [16], of Chang et. al. [17], and of Chen et. al. [18], and applications of the fuzzy commitment scheme of Juels and Wattenberg [19] to biometric templates, e.g., the constructions of Martini and Beinlich [20] for fingerprints, of Kevenaar et. al. [21] for face recognition, of Hao et. al. [22] for iris, and of Korte et. al. [23] for DNA fingerprints. The fuzzy vault has also been applied to iris recognition, e.g., in [24].

A. The general fuzzy vault scheme

The fuzzy vault has been proposed by Juels and Sudan in [3] and [25]. It is an error tolerant authentication scheme based on the set of private attributes $m_1, \ldots, m_t$, e.g., biometric feature data. While the reference data stored (the *vault*) allows performing the authentication check, it does not reveal these attributes. The scheme deploys a variant of Reed-Solomon decoding and hides the private user data among a large number of random *chaff points*.

During enrollment of a user, her (pairwise distinct) private attributes are encoded as elements $x_1, \ldots, x_t$ of a finite field $\mathbf{F}_q$. Then a random secret polynomial $P(z)$ over $\mathbf{F}_q$ with degree smaller than k is chosen. Each of the encoded attributes x_i is evaluated over the polynomial, resulting in a list of pairs $(x_i, y_i) \in \mathbf{F}_q^2$ with $y_i = P(x_i)$. In order to hide the private attributes, $r - t$ *chaff points* $x_{t+1}, \ldots, x_r \in \mathbf{F}_q$ are randomly selected so that $x_i \neq x_j$ for all $1 \leq i < j \leq r$. For each chaff point x_i, a random $y_i \in \mathbf{F}_q$ with $y_i \neq P(x_i)$ is chosen. The list of all pairs $(x_1, y_1), \ldots, (x_r, y_r)$, sorted in a predetermined order to conceal, which points are genuine and which are the chaff points, is stored as the *vault*.

The redundant encoding of the polynomial using the genuine points and its hiding among the chaff points is illustrated in Figure 1.

For authentication and recovery of the secret polynomial, another set of attributes (the *query set*) has to be presented. This set is compared with the stored

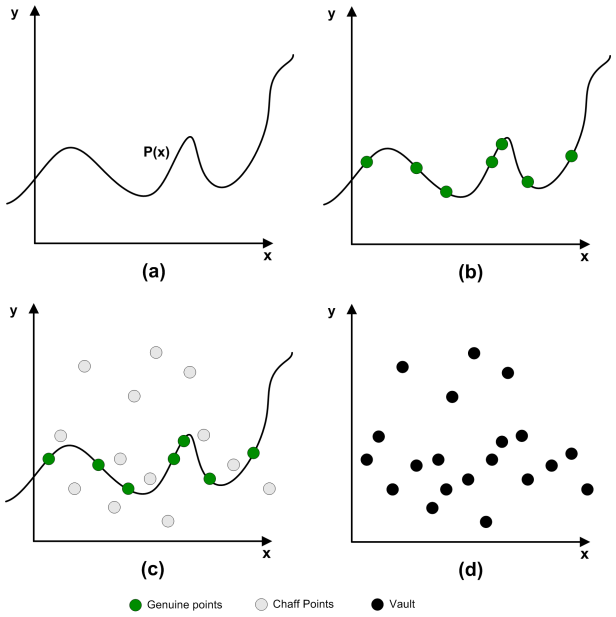

Figure 1. Illustration of the redundant encoding of the polynomial using the genuine points ((a) and (b)) and its hiding among the chaff points ((c) and (d)).

fuzzy vault $(x_1, y_1), \ldots, (x_r, y_r)$, and those pairs (x_i, y_i) are selected, for which x_i corresponds to an attribute in the query set. The selected points are then used to try to recover the secret polynomial using Reed-Solomon decoding.

If the number of genuine points among the identified correspondences (*correct matches*) is at least k, the secret polynomial can be recovered, either by Reed-Solomon decoding or by polynomial interpolation. However, if the set of correspondences also comprises chaff points (*false matches*), the number of correct matches must be greater than k, or the decoding must operate on subsets of the matches resulting in many trials. Details are given in Section III-B3.

B. Previous results

In [3], Juels and Sudan already provided an information theoretical security analysis for the general fuzzy vault scheme by giving estimates for the number of candidate polynomials that would fit with a given vault. A comprehensive information theoretic treatment of the fuzzy vault was given by Dodis et. al. [26][13], who proved upper bounds for the loss of entropy (information leakage) by the stored data for the fuzzy vault, the fuzzy commitment, and other schemes. In [1], we applied these general results to the fuzzy fingerprint vault and deduced lower bounds for the number of required minutiae and minutiae match rates, i.e., the fraction of minutiae in the vault matching with the minutiae of the query fingerprint, were deduced.

Implementations of the fuzzy vault for fingerprints were reported in many publications, the most notably of which are described below.

Clancy et. al. [5] were the first to propose a fuzzy fingerprint vault. Their construction uses only the location information of the minutiae, i.e., their orientations are neglected, and uses several measurements of the minutiae during enrollment to filter out spurious or unreliable minutiae. A drawback of their implementation is that it assumes that the fingerprints are already prealigned. The security against brute force attacks that try to *unlock the vault*, i.e. to determine the minutiae from the vault, was analyzed based on theoretical analysis and empirical data, and reasonable parameters were deduced. However, no actual authentication system was implemented and, consequently, the False Acceptance Rate (FAR) and Genuine Accept Rate (GAR) were not determined.

Uludag et. al. [6] also used minutiae location data only, and encoded a Cyclic Redundancy Check (CRC) into the secret polynomial in order to allow verification of correctness. A drawback of their construction is that it relied on human expert for the detection of minutiae in the fingerprint image and the identification of the minutiae correspondences between fingerprints. Based on experiments, eligible parameters were determined, the FAR and GAR as well as the security against brute force attacks were determined.

In [7], Uludag and Jain refined the construction of [6] by an automatic fingerprint alignment algorithm using the locations of highest curvature of the friction ridge as additional *helper data* stored in addition to the vault. In experiments, the authors determined FAR and GAR values for a single set of parameters.

Nandakumar et. al. [8] extended the ideas of the previous constructions. Their implementation of the fuzzy fingerprint vault used both minutiae locations and orientations. Spurious or unreliable minutiae were filtered by quality indices computed from local properties of the fingerprint image, and the fingerprint alignment and minutiae matching method based on points of highest curvature of [7] was improved. Experiments were conducted on two different databases and with several sets of parameters, FAR and GAR values were reported, and the complexity of brute force attacks was estimated.

In Li et. al. [9], an alternative fingerprint alignment method for the fuzzy vault was proposed, based on the topological structures around the *core* of the fingerprint. Their implementation used both minutiae locations and orientations. Experiments were conducted, and FAR and GAR values were reported and compared to those from [8].

In [10], the authors of the present article present an implementation of the fuzzy fingerprint vault using minutiae locations of several fingers per person. Several optimizations were applied, for instance, filtering of spurious or unreliable minutiae was performed both during enrollment and during authentication by several measurements and by quality values of the feature extraction algorithm, respectively. Fingerprint alignment was performed without additional helper data but by a minutiae matcher algorithm that optimized the number of minutiae correspondences between the fingerprints by means of relative rotation and translation. A comprehensive treatment of parameter selection criteria was given with respect to security against brute force attacks, and eligible parameters deduced by combining empirical data with analytical and heuristic arguments.

Other constructions [27][28] did not use the absolute location of the minutiae at all, but features deduced from the relative topological structures around the minutiae. These features are stable with respect to orientations, and in the case of [28], even of translations. The FAR and GAR values reported in [28] are better than those from [8] at the cost of a larger template size.

In [29], Nagar et. al. propose a combination of the fuzzy fingerprint vault and the fuzzy commitment scheme. The fuzzy commitment scheme is used to individually protect the ordinate values in the vault corresponding to the minutiae, i.e. the corresponding function value of the polynomial, using *minutiae descriptors*, topological properties of the minutiae's neighborhood. Thus, an attacker has to determine both, the minutiae descriptors and their locations. The FAR and GAR values reported are much better than that of [8].

In [12], Mihailescu et. al. presented an improved brute force attack and showed that the parameters suggested by Clancy et. al. in [5], and by Uludag and Jain in [7] do not provide the claimed security.

In [30], Scheirer and Boult proposed three new attack methods beyond the scenario of reconstruction of the biometric template from a single vault. The most serious one is a correlation attack, where an attacker can retrieve the private data from combining two independently generated vaults of the same user. This attack was implemented and proved to be very efficient for relevant parameters by Kholmatov and Yanikoglu [31]. A potential countermeasure against the correlation attack was proposed by Nandakumar et. al. [32].

A complete different type of attack was proposed by Chang et. al. [11], which tried to distinguish genuine minutiae in the vault from chaff points by the number of pixels in their proximity with sufficient distance to other points in the vault. This attack seems particularly promising of the number of chaff points used is close to the maximum possible so that the minimum distance enforced between the points constitutes a dense sphere packing, as discussed in Section IV-C2.

III. The fuzzy vault for fingerprints

In this section, we define the fuzzy fingerprint vault scheme, on which our security analysis is based. This scheme basically matches the implementations presented in [5], [6], [7], [9] and [10].

A. Adaptation to fingerprints

In order to implement the fuzzy vault for fingerprints, several adaptations are necessary.

1) Selection of biometric feature: In the *fuzzy fingerprint vault*, minutiae information is used as private attributes. Minutiae are bifurcations and endings of the ridges in a fingerprint and these features are commonly used for fingerprint authentication. The error correcting capacity of the fuzzy vault scheme fits well with typical measurement errors of minutiae data, in particular with insertions, deletions, and permutations of minutiae. Whereas the constructions of the fuzzy fingerprint vault in [5], [6], [7] and [10] use minutiae locations only, [8] uses both location and orientation of the minutiae.

A somewhat surprising finding is that using minutiae orientations in addition to their locations does not add significant benefit for the privacy protection of the fuzzy fingerprint vault. According to [8] "Using minutia orientation in addition to the location attribute has two advantages. During vault encoding, it increases the number of possible chaff points that can be added because we can now add a chaff point whose location is close to a genuine template minutia but with a different direction. During vault decoding, it makes it easier to filter out the chaff points from the vault because it is less probable that a chaff point will match with the query minutia in both location and direction." While we generally agree with this statement, we stress two points. First, the number of possible chaff points, as the number of potential locations for genuine minutiae, is irrelevant for the protection of the biometric data, as an attacker only needs to determine the genuine minutiae in the vault and, thus, can neglect potential chaff points that are not stored therein. Second, the strong dependencies of a minutia orientation with the corresponding location and with the orientation of other minutiae [33][34] can facilitate distinction of genuine minutiae from chaff points in the vault. For instance in [8], an example of a vault is depicted, where many points can be visually identified as probable chaff points due to their predominantly radial directions, and many pairs of spatially close points most likely contain at least one chaff point as the orientations differ too much to be in accordance with a plausible orientation field of a fingerprint.

For these reasons we restrict our consideration to the fuzzy fingerprint vault using minutiae location information only. Nevertheless, we stress that the method proposed in [29] to utilize additional data of minutiae, e.g. their orientations, in the fuzzy vault by means of the fuzzy commitment scheme, can significantly increase security. However, analysis of this approach is beyond the scope of this paper.

2) Tolerance of minutiae mapping: In the original fuzzy vault scheme, correspondence between points in the query set and the fuzzy vault means equality. For the application of the fuzzy vault to fingerprints, the definition of minutiae correspondence is typically widened to proximity with respect to the Euclidean distance to provide tolerance with respect to small deviations in the measured minutiae locations, which are introduced by elastic skin distortions and the limitation of optical and algorithmic accuracy of the measurement. We will, therefore, assume that a minutia in a query fingerprint matches with a minutia or chaff point in the vault if both locations have an Euclidean distance of at most δ and if there is no other point in the vault closer to the minutia. In the case that several fingers are used per person, matching minutiae or chaff point must be from the same finger.

3) Fingerprint alignment: The position of a fingerprint varies between different measurements, inducing relative translations and rotations of the corresponding minutiae sets. In order to identify correspondences between the minutiae in the query fingerprints and the minutiae stored in the vault, these minutiae sets must be, at least roughly, aligned with respect to each other. The fingerprint alignment method is crucial for the fuzzy fingerprint vault, as an incorrect alignment results in a relative rotation of the query minutiae set to the points in the vault and, with very high probability, in an insufficient number of identified support points of the polynomial, which results in a failure to authenticate. Several techniques have been proposed to ensure a sufficiently correct alignment. Some implementations use topological information of the fingerprint ridge patterns [7][9], while others apply a minutiae matcher algorithm [10][8]. Early proposals [5][6] even relied on minutiae matching by human experts, which is clearly not practical. We do not impose any assumption on the method used for alignment or the goodness of the relative alignment; instead, we only consider the rate, at which the minutiae in the vault are identified in the query fingerprints (the *minutiae match rate*), which in turn depends on the alignment of these minutiae sets.

4) Combining several fingers: According to [35], the minutiae of a single finger do not provide sufficient entropy to extract a secure cryptographic key. Therefore, we allow to use minutiae from more than one finger. The minutiae of the different fingers can be easily fused on a feature level by storing with each minutia or chaff point an index of the corresponding finger. A general

discussion on approaches for multi-instance fusion in template protection schemes and implications to security can be found in [36]. Since we allow the biometric templates to be taken from several fingers of a person, the minutiae, and likewise, the chaff points, are not only represented by their location but also by the finger code. Subsequently, let both, minutiae and chaff points, be represented as points $\mathbf{m} = (\mathbf{a}, l) \in \mathbf{Z}^2 \times \{1, \ldots, f\}$, where $\mathbf{a}$ is a location in the fingerprint image represented with respect an arbitrary coordinate system, and l is an index of the finger. We will define the *distance* $||\mathbf{m}_i - \mathbf{m}_j||$ of minutiae or chaff points as the Euclidean distance of their locations $\mathbf{a}_i$ and $\mathbf{a}_j$, if both points are from the same finger, i.e., if $l_i = l_j$, and as infinite otherwise.

5) Embedding to finite field: In order to evaluate the polynomial, the minutiae data have to be embedded into the finite field. In previous implementations, the minutia location was represented as field element using a suitable encoding function. For our analysis we use a slight optimization and evaluate the polynomial not on the minutiae data but on the indices of the minutiae in the vault. This modification minimizes the size of the function values stored as part of the vault and thus the loss of entropy. See Section IV-B for details. The finite field will be chosen larger than the number of points in the vault and, consequently, injective encoding of the indices to the finite field is possible. For the ease of reading, we will omit the encoding and treat the indices as if they were field elements.

6) Storage of a hash value of the polynomial: In contrast to the implementations in [6], [7], and [8] that incorporate a CRC check sum into the polynomial's coefficients to allow verification of the recovered polynomial, we store a hash value of the coefficients, as it is done (analogously) in the fuzzy commitment scheme [19]. This approach has the advantage that it does not reduce the search space for attackers due to the internal structure of the secret. We assume that the hash value does not leak any information; this assumption is frequently used in cryptographic protocol analysis [37].

7) Selecting the feature space: Subsequently, let $\mathcal{M}$ be the set of all possible minutiae and $n = |\mathcal{M}|$ the number of possible values for a minutia or chaff point. If the fingerprint images have a resolution of $N \cdot M$ pixels (height and width), we have $\mathcal{M} \subseteq [1, N] \times [1, M] \times \{1, \ldots, f\}$, when expressing locations in Cartesian coordinates. As we will see in Section VI-A, it may be useful to restrict the set $\mathcal{M}$ to a subset with high frequency of minutiae occurrence.

B. Description of the scheme

As any biometric authentication system, the fuzzy fingerprint scheme comprises an enrollment and a verification step. In the context of the fuzzy vault, these steps are also referred to as *vault locking* and *vault unlocking*.

1) Enrollment (Locking the Vault): Let q be prime power and $k < t < r \leq q$. For each user, a random polynomial P of degree less than k over the finite field $\mathbf{F}_q$ is selected. The coefficients of this polynomial represent the secret key of the scheme. Then, a set T of t minutiae of the user is determined. This set of minutiae is amended by random chaff points, resulting in a set of r points, containing t genuine minutiae and $r - t$ chaff points. A minimum distance of d is enforced among minutiae and chaff points to reduce errors during verification by wrong mapping of close points. Furthermore, in order to ensure that minutiae and chaff points within the vault are not distinguishable by their index, they are lexicographically ordered.

For all genuine minutiae $\mathbf{m}_j$, where j is its index after applying the lexicographic order, $y_j = P(j)$ is computed. For each chaff point $\mathbf{m}_j$, where j is its index in the lexicographic order, a random value $y_j \neq P(j)$ is chosen. The *vault* consists of the lexicographically ordered list of minutiae and chaff points, paired with the corresponding y_j values. The vault and a cryptographic hash value of the concatenated coefficients of P are stored in the database.

To facilitate security analysis, we assume that the chaff points are chosen uniformly from the set $\mathcal{M}$ of potential minutiae with the restriction that the minimum distance is respected. However, since the locations of genuine minutiae are not uniformly distributed in the image area, see Section VI-A, selecting chaff points with a more natural distribution that resembles that of genuine minutiae would make them less distinguishable from the genuine minutiae in the vault. Nevertheless, since the chaff points are chosen after the genuine minutiae in the vault have been determined, those points in the vault that correspond to image locations, where minutiae occur with particularly high frequency, are more likely to be genuine minutiae anyway.

2) Verification (Unlocking the Vault): We only consider an authentication in the verification scenario, where the identity of the user is known a priori.

In order to verify the identity of a user, the minutiae are measured from a query fingerprint. Then the matches between these minutiae and the minutiae and chaff points contained in the vault are identified. Precisely, for each minutiae in the query fingerprint, the closest point in the vault with Euclidean distance smaller than a threshold δ is identified, where δ is a tolerance parameter. The matching of the minutiae in the query fingerprint with those in the vault requires a (nearly) correct alignment of the query fingerprint with respect to the minutiae in the vault. To accomplish this, either vertical alignment of the fingerprints prior to minutiae extraction , e.g., using singular point detection [38],

can be used, or a minutiae matching algorithm can be deployed that tries to find the alignment, by which the number of matches are maximized [10].

The indices of the matching minutiae and chaff points in the vault, along with the corresponding y_i values, are used to recover the secret polynomial P, see Section III-B3 for details. If the number of genuine minutiae among the matches is sufficiently high the polynomial can be recovered. See Section III-B3 for a discussion.

The correctness of the recovered polynomial is checked using the stored hash value.

3) Recovery of the polynomial: The unlocking of the vault during authentication requires the recovery of the secret polynomial from a set of points (j_i, y_{j_i}), some of which (those resulting from matches with minutiae) lying on the polynomial P, while others (those resulting from matches with chaff points) do not. For this task, an Reed-Solomon decoder is needed that on input $(j_1, y_{j_1}), \ldots, (j_\ell, y_{j_\ell}) \in \mathbf{F}_q^2$ with $\ell \geq k$, outputs $e_0, \ldots, e_{k-1} \in \{0, \ldots, q-1\}$, so that $y_{j_i} = P(j_i)$ holds for at least k of the points (j_i, y_{j_i}) with $P(z) = \sum_{i=0}^{k-1} e_i z^i$, if such a polynomial exists. We assume that the Peterson-Berlekamp-Massey-decoder is used as suggested in [3]. This technique is successful if $(\ell + k)/2$ of the x points handed over to the decoder are correct. Although there are Reed-Solomon decoders that can decode with only $\sqrt{\ell k}$ correct points, they do not offer significant advantage for the fuzzy vault, because $\sqrt{\ell k}$ is quite close to $(\ell + k)/2$ for typical parameters, and they are computationally much less efficient [3].

IV. Security analysis

It is understood that there are different threats for the fuzzy fingerprint vault and that the exposure of the original template is just one of them. Three other types of such attacks against the fuzzy vault are described in [30], among which the correlation of two vaults from independent enrollments ("record multiplicity" attack) represents a serious threat to the fuzzy vault, which is still not completely satisfactorily solved. However, a comprehensive analysis of all potential attacks against the fuzzy vault would go beyond the scope of this paper. In this contribution we focus on the security of the fuzzy fingerprint vault with respect to attacks that try to recover the template or the secret polynomial from the vault. In this context, we will investigate both lower bounds (given by information theoretical results) and upper bounds (given by known attacks) of the security.

Throughout this article, let all logarithms be to the base 2.

A. Provable Security

In this section, we provide lower bounds of security with respect to attacks that aim to recover the template or, equivalently, the secret polynomial from the vault. Precisely, these results upper bound the probability that an attacker, whatever strategy and computational resources he deploys, determines the correct polynomial or template from a given vault. The only way of the attacker to increase his success probability is to check the correctness of his output, e.g., using the hash value stored in addition to the vault, and to repeat his guessing. This "provable security" is achieved by a randomization process during enrollment, which ensures that for each given vault there are many "fitting" templates and polynomials that could have been used to generate it, and the conditional probability of any assumed template or polynomial is small.

The lower bounds on the security are given by security proofs, which are deducted from information theoretical results. We admit that the term *proof* is not completely exact here. Firstly, since the security of a biometric scheme always depends on the distribution of the biometric features within the considered population, estimations based on empirical data are necessary. Secondly, the minimum distance enforced during the enrollment constitutes a sphere packing problem that requires heuristic arguments. In the course of evaluating optimal parameters with respect to the achieved security bounds, we will use further approximations, e.g., to allow a treatment of binomial coefficients with calculus.

Following [26], we use the min-entropy $\mathbf{H}_\infty$ to quantify the security of the scheme. This measure has the advantage that it expresses the (negative logarithm of the) maximum probability of guessing, and thus, can be used to deduct lower bounds on attacks (see Theorem 1). In contrast, some publication, e.g., [39] and [40], use the Shannon entropy $\mathbf{H}$ to assess the security of biometric template protection. The use of the Shannon entropy might be appealing due to the rich underlying mathematical theory, which allows to deduct quite impressive results, e.g., see [41]. However, as shown in [42], the Shannon entropy can, for certain probability distributions, be very insignificant for assessing the minimum attack complexities. In general, the inequality $\mathbf{H}_\infty(A) \leq \mathbf{H}(A)$ holds for any random variable A, but, in the opposite direction, the Shannon entropy can exceed the min-entropy (and thus the logarithm of attack complexities) by any factor. Consequently, the Shannon entropy is not the eligible measure to determine the capacity of the fuzzy fingerprint vault with respect to provable security, i.e., to lower bounds for attack complexities.

Subsequently, let P(X) denote the probability of an event X and let $\mathbf{E}_{a \leftarrow A}[f(a)]$ be the expectation of the function value of a random variable A. The *min-entropy*

of a random variable A is given by

$$\mathbf{H}_\infty(A) := -\log(\max_a(\mathrm{P}(A=a))),$$

and the *average min-entropy* of A given B is defined as

$$\widetilde{\mathbf{H}}_\infty(A \mid B) := -\log\left(\mathbf{E}_{b\leftarrow B}\left[2^{-\mathbf{H}_\infty(A \mid B=b)}\right]\right)$$

For a biometric encryption scheme with feature vector T and vault Y, we call $\mathbf{H}_\infty(T) - \widetilde{\mathbf{H}}_\infty(T \mid Y)$ the *loss of entropy.*

B. Minimum attack complexity

The following result shows that the security of the fuzzy vault for fingerprints can be lower bounded by the average min-entropy of the biometric feature vector given the vault. The result is a trivial adaptation of Theorem 1 and Lemma 2 from [23], and follows immediately from the definition of the min-entropy. It holds (with according notations) for any biometric encryption scheme, in which the secret key and the vault uniquely determine the biometric feature vector.

Theorem 1. *Any algorithm that takes as input the vault Y and tries to output the secret polynomial $P(x) = \sum_i e_i x^i$ or the set of minutiae T has an average success probability of at most $2^{-\widetilde{\mathbf{H}}_\infty(T \mid Y)}$.*

An attacker who has determined the original template T of a user can recover the secret polynomial P by simulating a verification using T and the vault Y; the stored hash value allows checking, if the resulting polynomial is correct. On the other hand, if an attacker has (somehow) learned P, he can easily recover T from the vault Y, simply by determining all $\mathbf{m}_j$ in Y with $y_j = P(j)$. Therefore, it is equally difficult to recover the template T as to determine the secret polynomial P. In terms of information theory, we obtain the following result:

Theorem 2. *Given the stored reference data (vault and hash value), recovering to biometric template T is computationally equivalent to determination the secret polynomial P. Moreover, $\widetilde{\mathbf{H}}_\infty(T \mid Y) = \widetilde{\mathbf{H}}_\infty(P \mid Y)$.*

On the other hand, the success probability of an *fingerprint dictionary attack* (see Section IV-E) trying to recover the polynomial by choosing random templates equals, by definition, the False Accept Rate (FAR), while the min-entropy upper bounds the probability of any attack. Therefore, we can state the following result, which was presented already - for a larger class of schemes and using different mathematical notations - in [43].

Theorem 3. $\widetilde{\mathbf{H}}_\infty(P \mid Y) \leq -\log(\mathrm{FAR})$.

Theorem 3 implies that the information content of a cryptographic key extracted from P cannot exceed $-\log(\mathrm{FAR})$. In [35], this conclusion was drawn for arbitrary schemes, in which biometric data is used to extract a secret key. Since it is indeed possible to recover P from Y in 1/FAR steps on average by the fingerprint dictionary attack (see Section IV-E), the length of any cryptographic key secured by a fuzzy fingerprint vault should not exceed $-\log(\mathrm{FAR})$ bits. In order to extract this number of bits from P while preserving all its entropy, it can be used as a seed of a pseudo-random number generator.

C. Loss of entropy

By definition, the average min-entropy of the biometric feature vector given the vault is the difference between the entropy of the feature vector and the loss of entropy. We now turn to the estimation of the latter quantity. We first consider the case, where no minimum distance is enforced among the minutiae and chaff points, i.e., the case $d = 1$, and then generalize these results to the case $d > 1$.

1) The case of trivial minimum distance: In [13], Lemma D.1, a lower bound for the loss of entropy in the original fuzzy vault scheme has been given. In the case $d = 1$, i.e., if the minimum distance is trivial and the minutiae and chaff points only need to be distinct, the result can be directly applied to our implementation. The proof is a simple adaptation of the proof of Lemma D.1 in [13].

Theorem 4. *If $d = 1$, the loss of entropy is at most $(t-k)\log q - \log\binom{r}{t} + \log\binom{n}{t} + 2$, i.e.,*

$$\widetilde{\mathbf{H}}_\infty(T \mid Y) \geq \mathbf{H}_\infty(T) - (t-k)\log q + \log\binom{r}{t} - \log\binom{n}{t} - 2. \quad (1)$$

Proof: By Lemma 3.1 in [13]

$$\widetilde{\mathbf{H}}_\infty(T \mid Y) \geq \mathbf{H}_\infty(T, Y) - \lambda,$$

where 2^λ is the number of possible values that Y can take.

We first estimate $\mathbf{H}_\infty(T, Y)$. The information contained in T and Y is composed of four parts: The set of minutiae T, the set of chaff points, the y_i-values for the minutiae, and the y_i-values for the chaff points. The entropy of the $r - t$ chaff points is given by $\log\binom{n-t}{r-t}$, because they are randomly selected from all $n-t$ potential points that are distinct from the t minutiae. Given T, there is a one-to-one correspondence between the y_i-values for the minutiae and the random polynomial P; hence, their entropy is $k\log q$. Finally, the y_i-values for the chaff points are randomly selected from $\mathbf{F}_q \setminus \{P(i)\}$, and therefore their entropy is $(r-t)\log q - 1$. This sums up to

$$\mathbf{H}_\infty(T, Y) = \mathbf{H}_\infty(T) + \log\binom{n-t}{r-t} + k\log q + (r-t)\log(q-1).$$

On the other hand, since the minutiae and chaff points in Y are in lexicographic order, we have $2^\lambda = \binom{n}{r} q^r$. Using $(r-t)\log\frac{q}{q-1} < q\log\frac{q}{q-1} \leq 2$ and

$$\binom{n}{r}\binom{r}{t} = \binom{n}{t}\binom{n-t}{r-t}$$

this yields the result. Q.E.D.

This result can be interpreted as follows:

- The term $(t-k)\log q$ represents the information leaked by the redundantly encoded secret polynomial. Precisely, this term is composed of the $t\log(q)$ bits of information revealed by the y_i-values corresponding to the genuine minutiae and the $k\log(q)$ of information contained in the secret polynomial.
- The term $\log\binom{r}{t}$ estimates the amount of security contributed by "hiding" the t genuine minutiae among the r chaff points.
- The term $\log\binom{n}{t}$ refers to the information leaked by publishing T as part of the vault.

Since $\mathbf{H}_\infty(T) \leq \log\binom{n}{t}$, the lower bound (1) is positive (and hence meaningful) only if $q^{t-k} \leq \binom{r}{t} \leq \binom{q}{t} \leq q^t/(t!)$, which implies $q > (t/\mathrm{e})^{t/k}$. The exponent t/k defines the error correction capacity of the scheme and, according to our experiments, must be larger than 1.5 to achieve a satisfactory false rejection rate (FRR). Therefore, we can obtain a scheme with provable security according to Theorems 1 and 4 only if q is considerably greater than $(t/\mathrm{e})^{1.5}$.

The bound provided by Theorem 4 is not tight. In particular, in the estimation of λ, the number of possible values for $(y_1,\ldots,y_r)$ is smaller than q^r, because, by construction, $(y_1,\ldots,y_r)$ can only assume those vectors, for which at least t of the pairs (i,y_i) lie on a common polynomial of degree smaller than k. This is exactly the set of words in the Reed-Solomon code $\mathrm{RS}_q(r,k)$ having error distance, i.e., Hamming distance to the next codeword, at most $r-t$. We are not aware of any estimation on their number that could be used to improve Theorem 4. On the other hand, for $t < (r+k)/2$ the Hamming spheres of radius $r-t$ around the code words overlap and hence already cover a significant part of $\mathbf{F}_q^r$. Thus, for $t \ll (r+k)/2$ it is not clear whether a better estimate for λ would result in a significant improvement of Theorem 4.

If we chose the chaff point according to a distribution that resembles that of minutiae locations (instead of uniformly from $\mathcal{M} \setminus T$), we would end up with a smaller bound for $\widetilde{\mathbf{H}}_\infty(T\,|\,Y)$. This reduction of provable security is paradox, as a more natural distribution of the chaff points makes them less distinguishable from the genuine minutiae, and hence, strengthens the security. However, the proof techniques used in Lemma D.1 of [13] measure the information leakage not by the entropy of Y but by the number of its possible values. Therefore, a non-uniform distribution does not change the estimate of the leaked information while it reduces the entropy added.

2) The case of non-trivial minimum distance: For the case $d \geq 2$, we have to analyze the effect of the minimum distance to the number of possible choices for the chaff points and the possible values for the vault Y. Subsequently, we will use the following definitions.

For a point $\mathbf{m} \in \mathcal{M}$ let $B_d(\mathbf{m})$ denote the set of points in $\mathcal{M}$ that have Euclidean distance to $\mathbf{m}$ smaller than d, and let $V_d = 1 + 4\sum_{i=1}^{\lceil d-1\rceil}\lceil\sqrt{d^2-i^2}\rceil$ be the number of integer points $\mathbf{m} \in \mathbf{Z}^2$ with Euclidean norm smaller than d. Obviously, $|B_d(\mathbf{m})| \leq V_d$.

Since the minutiae and chaff points are selected with minimum distance d, the d-sphere centered at a selected point is excluded from the potential values for subsequent points. If the d-sphere neither juts out beyond $\mathcal{M}$ nor intersects with the d-spheres of the previously selected points, the number of possible choices for the next point is reduced by exactly V_d; otherwise, the reduction is smaller.

These effects make an exact estimation of the number of possible choices for the chaff points or the number of potential values for Y virtually impossible. However, for $rV_d \ll n$, the likelihood that a selected point is too close to the boundary of $\mathcal{M}$ or to a previously selected point is small. In this case, the approximation that, on average, each point reduces the number of choices for the subsequent points by V_d is quite accurate. Subsequently, we assume $rV_d \ll n$, and thus, approximate the number of chaff points by $V_d^{r-t}\binom{n/V_d-t}{r-t}$ and the number of possible values for Y by $V_d^r\binom{n/V_d}{r}$. Analogously to Theorem 4, we obtain the following result:

Theorem 5. *For $rV_d \ll n$, the maximal loss of entropy is approximately $(t-k)\log q - \log\binom{r}{t} + \log\binom{n/V_d}{t} + t\log V_d + 2$, i.e., $\widetilde{\mathbf{H}}_\infty(T\,|\,Y) \geq E$ with*

$$E \approx \mathbf{H}_\infty(T) - (t-k)\log q + \log\binom{r}{t} - \log\binom{n/V_d}{t} - t\log V_d - 2. \quad (2)$$

D. Entropy of the feature vector

The entropy of the feature vector T is defined by the maximum likelihood that it takes a certain instance M. Since for the parameters of interest the number of possible instances by far exceeds the number of persons, for which minutiae information is available, we can estimate the entropy of T only by modeling its probability distribution. Several publications have proposed models for minutiae distributions, e.g., [44] and [45]. However, their analysis already takes into consideration the error

tolerance of the minutiae matching algorithm and is therefore not applicable for the determination of the raw entropy $\mathbf{H}_\infty(T)$.

We model the probability distribution of T by a probabilistic process **Select_T**, where the t minutiae are successively chosen. The first minutia $\mathbf{m}_1$ is selected according to a distribution $\mathcal{D}$ defined over $\mathcal{M}$. All subsequent minutiae $\mathbf{m}_i$ are selected to the same distribution $\mathcal{D}$ restricted to the areas in $\mathcal{M}$ not covered by the d-spheres $B_d(\mathbf{m}_1), \ldots, B_d(\mathbf{m}_{i-1})$ around the previously chosen minutiae.

Like all previous models for the distribution of minutiae, we do not assume any statistical dependency between the locations of the individual minutiae, except that they have the minimum distance d. Although it is known that minutiae tend to overdisperse on a small scale (precisely, between 11 and 20 pixels for 500 dpi) and to cluster on a large scale [46]. The overdispersion on a small scale can be partially explained by minimum distances typically enforced by minutiae extraction algorithms to avoid ambiguous results, e.g., see [47], but in [46] biological arguments taken from [48] are used. Due to the enforcement of a minimum distance d during template selection this effect is in line with our model, at least for sufficiently large d. Furthermore, the overdispersion reported in [46] is rather weak. On the other hand, in [46] the observed clustering on a large scale is explained by a higher minutiae frequency around core or delta points. This effect is addressed in our model by using a non-uniform distribution $\mathcal{D}$, in which higher probabilities refer to such cluster points. Of course, there could be more complex dependencies between the location of individual minutiae. However, to our knowledge, there are no observations or models implying such dependencies (we refer to [49] for a detailed discussion of this aspect).

Using our statistical model, we can show the following result:

Theorem 6. *If T is chosen according to the random process* **Select_T** *and the maximum likelihood of a minutiae location is $1/\psi$, then*

$$\mathbf{H}_\infty(T) \geq \log \binom{\psi/V_d}{t} + t \log V_d$$

Proof: Let $\mathrm{P}(A)$ denote the probability of random event A. Furthermore, for $i = 1, \ldots, t$ let M_i let be the random variable of the i-th point output by **Select_T**. By M we denote the random variable chosen according to $\mathcal{D}$. Then by definition

$$\begin{aligned} 2^{-\mathbf{H}_\infty(T)} &= \max\left(\mathrm{P}\left(\{M_1, \ldots, M_t\} = \{\mathbf{m}_1, \ldots, \mathbf{m}_t\}\right)\right) \\ &\leq t! \max\left(\mathrm{P}\left(M_1 = \mathbf{m}_1, \ldots, M_t = \mathbf{m}_t\right)\right), \quad (3) \end{aligned}$$

where the maximum is taken over all $\mathbf{m}_1, \ldots, \mathbf{m}_t$. The latter probability $\mathrm{P}\left(M_1 = \mathbf{m}_1, \ldots, M_t = \mathbf{m}_t\right)$ can be expanded to

$$\prod_{i=1}^{t} \mathrm{P}\left(M_i = \mathbf{m}_i \,|\, \forall j < i : \; M_j = \mathbf{m}_j\right).$$

The first term has an empty condition and is limited by $1/\psi$, while the other factors can be estimated as follows:

$$\begin{aligned} &\mathrm{P}\left(M_i = \mathbf{m}_i \,|\, M_1 = \mathbf{m}_1, \ldots, M_{i-1} = \mathbf{m}_{i-1}\right) \\ &\quad = \mathrm{P}\left(M = \mathbf{m}_i \,|\, M \notin B_d(\mathbf{m}_1) \cup \ldots \cup B_d(\mathbf{m}_{i-1})\right) \\ &\quad = \frac{\mathrm{P}\left(M = \mathbf{m}_i \wedge M \notin B_d(\mathbf{m}_1) \cup \ldots \cup B_d(\mathbf{m}_{i-1})\right)}{\mathrm{P}\left(M \notin B_d(\mathbf{m}_1) \cup \ldots \cup B_d(\mathbf{m}_{i-1})\right)} \\ &\quad \leq \frac{\mathrm{P}\left(M = \mathbf{m}_i\right)}{1 - \mathrm{P}\left(M \in B_d(\mathbf{m}_1) \cup \ldots \cup B_d(\mathbf{m}_{i-1})\right)} \end{aligned}$$

By assumption, the numerator is at most $1/\psi$, while the probability in the denominator is limited by the term $\left|B_d(\mathbf{m}_1) \cup \ldots \cup B_d(\mathbf{m}_{i-1})\right| / \psi$, which is at most $(i-1)V_d/\psi$. This results in

$$\begin{aligned} &\mathrm{P}\left(M_i = \mathbf{m}_i \,|\, M_1 = \mathbf{m}_1, \ldots, M_{i-1} = \mathbf{m}_{i-1}\right) \\ &\qquad \leq \frac{1}{\psi - i \cdot V_d} \quad (4) \end{aligned}$$

Consequently, with (3) we obtain

$$2^{-\mathbf{H}_\infty(T)} = t! \prod_{i=0}^{t-1} \frac{1}{\psi - i \cdot V_d}.$$

The desired result now follows by elementary transformations. Q.E.D.

By combining Theorem 5 with Theorem 6 we obtain the following Theorem.

Theorem 7. *For $d \geq 1$, $\widetilde{\mathbf{H}}_\infty(T \,|\, Y) \geq E$ with*

$$E \approx \log \binom{\psi/V_d}{t} - (t-k) \log q + \log \binom{r}{t} - \log \binom{n/V_d}{t} - 2,$$

where $1/\psi$ is the maximum likelihood of a minutiae location.

E. Practical Security

In the previous sections, we have focused on provable security in terms of lower bounds for the number of trials for attacks. However, these bounds are not at all sharp, as existing attacks are much less efficient than these bounds would allow. For this reason, we now consider practical attacks and analyze the security of the fuzzy vault with respect to these attacks.

For the recovery of the original template and the secret polynomial from a single vault two kinds of brute force attacks can be distinguished: exhaustive search on the templates or exhaustive search on the polynomials.

1) Fingerprint dictionary attack: In the *fingerprint dictionary attack* an attacker collects a large number of realistic templates, either from real fingerprints or artificially. For all these templates he simulates the authentication procedure using the vault until the secret polynomial has been found. Although it has been shown in [3] that for typical parameters there is with high probability a large number of polynomials "fitting" the vault, i.e., there are many polynomials of degree smaller than k such that exactly t of the stored points lie on the polynomial, the attacker can check the correctness of the polynomial using the stored hash value (see Section III-B2). If the templates are chosen with the same probability distribution as they occur within the group of users of the biometric application, the success probability of each attempt equals the False Accept Rate (FAR) and the attacker needs FAR^{-1} trials on average. (For this reason, this attack is also referred to as *FAR attack* [36].) This results in an overall workload of $N_\mathrm{v} \cdot \mathrm{FAR}^{-1}$, where N_v is the effort for a single verification.

Usually, the FAR is determined empirically by performing a sufficiently large number of impostor matches, i.e., matches with fingerprints of other individuals. Of course, the empirical determination has to be done for every set of parameters separately. Subsequently, we explore if the FAR can also be estimated theoretically as a function in dependence of the parameters of the scheme.

Let m_c denote the number of *correct matches*, i.e., the matches between the query fingerprint and the genuine minutiae, and let m_f be the number of *false matches*, i.e., the matches between the query fingerprint and the chaff points. According to Section III-B3, the reconstruction of the polynomial is only possible if $m_\mathrm{c} \geq m_\mathrm{f} + k$. Therefore, we obtain

$$\mathrm{FAR} = \sum_{a=k}^{t} \sum_{m_\mathrm{f}=0}^{b-k} \mathrm{P}(m_\mathrm{c} = a \wedge m_\mathrm{f} = b).$$

The probability $\mathrm{P}(m_\mathrm{c} = a \wedge m_\mathrm{f} = b)$ that there are exactly a correct and b false matches depends on the specific method used to identify minutiae correspondences between the query fingerprint and the vault. This method usually searches for the correct relative alignment of the minutiae set in order to compensate global rotations and translations of the fingerprints. The precision and reliability of the alignment method has great impact on the probability $\mathrm{P}(m_\mathrm{c} = a \wedge m_\mathrm{f} = b)$. Therefore, a reasonably accurate theoretical estimation of the FAR is unfeasible unless very simplifying assumptions are made on the alignment, e.g., that the fingerprints are perfectly aligned.

2) Polynomial reconstruction: An attacker can try to recover the polynomial from the stored points directly, i.e., without exploiting knowledge about the distribution of minutiae and the corresponding feature vectors. The underlying computational problem is known as Reed-Solomon decoding problem or *polynomial reconstruction* problem. It is believed to be hard for $k < t < \sqrt{r(k-1)}$, and it is known that random instances of this problem are as hard as the worst case [50]. For very large fields sizes, it is known to be NP-complete [51]. For these reasons, it has been repeatedly suggested as a basis for cryptographic constructions [50].

According to [52], two approaches are most efficient for the polynomial reconstruction: Either, after guessing k genuine minutiae, the polynomial is reconstructed using polynomial interpolation, e.g., by Lagrange interpolation, or it is determined by Reed-Solomon list decoding after guessing $\Delta = r - \frac{t^2}{k-1} + 1$ of the chaff points. Let aside the fact that polynomial interpolation is much more efficient than Reed-Solomon list decoding, for typical parameters (and all parameters suggested so far), $\Delta > k$ and therefore, it is more efficient to guess k genuine minutiae among the stored points.[1] This approach has been used by the attack of Mihailescu, et. al. [12], which systematically searches through all subsets $\{j_1, \ldots, j_k\}$ of $\{1, \ldots, r\}$, computes the unique polynomial P satisfying $P(j_i) = y_{j_i}$ by polynomial interpolation, and checks the correctness of this polynomial. Assuming (as done in [12]) that all points in the vault are equally likely to be a genuine minutia, this attack needs $\binom{r}{k}/\binom{t}{k}$ trials on average. In [12], the number of operations needed for each interpolation is estimated as $6.5k\log^2(k)$ using results from [53]. However, this estimation is incorrect, as 6.5 is the explicit constant for the running time of fast polynomial interpolation only if it is expressed in terms of the running time $\mathsf{M}(k)$ for multiplication of polynomials of degree k (see [53], Corollary 10.2). Dissolving $\mathsf{M}(k)$ to $O(k\log(k))$ introduces another factor of 18 (see Corollary 8.19 in [53]). Thus, we have to correct the running time estimation for polynomial interpolation used by [12] to $117k\log^2(k)$. This results in an average number of

$$W \leq 117k\log^2(k)\binom{r}{k}\binom{t}{k}^{-1}, \tag{5}$$

operations required for the attack. Note that the term "operations" refers to additions, subtractions, multiplication and division over $\mathbf{F}_q$.

Of course, the assumption that all points in the vault are genuine minutiae with the same probability, is an

[1] This is in contrast to the (obviously wrong) statement in [52].

oversimplification. There are (at least) two effects resulting in a non-uniform distribution of these probabilities, which are subsequently discussed.

Firstly, it has been shown in [11] that since the chaff points are selected after the genuine minutiae in the vault were determined, the average free area (not occupied by the d-spheres $B_d(\mathbf{m})$ of other points) in the proximity of chaff points is smaller than that around genuine minutia. This tendency can be exploited to tell apart genuine minutiae from chaff points more efficiently than by mere guessing. In [11], the method has been shown to be efficient in the case of a maximum number of chaff points; given a density 0.45 for random sphere packings [5], the maximum number r of points in the vault is $0.45 \cdot n/V_{\lceil d/2 \rceil}$. In this case, the polynomial reconstruction attack can be sped up considerably by preferring those points having more free area in their neighborhood than others. However, it has been shown in [5] that if the number of chaff points is considerably smaller than their maximum, the effect exploited by the attacker is much weaker. We assume that r is chosen considerably smaller than its maximum value, i.e., that $r \ll 0.45 \cdot n/V_{\lceil d/2 \rceil}$, and thus, this attack method is less efficient than the second approach for selecting points in the polynomial reconstruction (see following paragraph). We will critically review this assumption on the basis of our results in Section VI-D.

Secondly, the locations of minutiae are not uniformly distributed. Even if chaff points were selected using a "natural" distribution, i.e., a distribution resembling that of minutiae, they were less likely to occupy frequent minutiae locations than the genuine minutiae in the vault, because the latter ones are selected before the chaff points are chosen. As we assumed that chaff points are selected according to a uniform distribution, this effect is even stronger. Subsequently, we discuss the advantage an attacker can gain from this effect.

Obviously, the best strategy for speeding up the polynomial reconstruction is to try points $\mathbf{m}_i \in R$ with higher conditional probability $\mathrm{P}(\mathbf{m}_i \in T \,|\, \mathbf{m}_i \in R)$ first. In particular, an optimized attack would first determine the minutiae occurrence frequency for all locations in $\mathcal{M}$, sort the points $\mathbf{m}_1, \ldots, \mathbf{m}_r \in R$ according to the frequency p_i corresponding to their location so that $p_1 \geq \ldots \geq p_r$, and would then search through all subsets $\{j_1, \ldots, j_k\}$ of $\{1, \ldots, K\}$ with increasing $K \geq k$. We call this optimized attack the *smart polynomial reconstruction*. Up to our knowledge, this (quite obvious) improvement of the polynomial reconstruction attack on the fuzzy fingerprint vault has not been proposed in the literature so far, although in [32], the basic idea "to exploit the non-uniform nature of biometric features and develop attacks based on statistical analysis of points in the vault" has already been phrased.

The following results enables us to deduct an approximate upper bound for the success probability of this attack method. As for the proof of Theorem 5, we assume $rV_d \ll n$ and use the approximation that, on average, each point selected for R reduces the number of choices for the subsequent points by V_d. We will critically review this assumption on the basis of our results in Section VI-D.

Lemma 8. *Let $rV_d \ll n$ and $\mathbf{m}_{j_1}, \ldots, \mathbf{m}_{j_k}$ be an arbitrary subset of points from the vault R. Then, the probability p that these points are all genuine minutiae, is approximately upper bounded by*

$$p \lessapprox \prod_{i=1}^{k} \frac{(t-i+1)(n-tV_d)}{(r-t)\psi_i + (t-i+1)n - t(r-i+1)V_d},$$

where $1/\psi_i$ is the probability of a minutiae occurrence at position $\mathbf{m}_{j_i}$.

Proof: For the ease of reading, we use $\mathrm{P}^{(i-1)}(A)$ to denote a probability of an event A under the condition that points $\mathbf{m}_{j_1}, \ldots, \mathbf{m}_{j_{i-1}}$ are genuine minutiae, i.e.,

$$\mathrm{P}^{(i-1)}(A) = \mathrm{P}(A \,|\, \mathbf{m}_{j_1}, \ldots, \mathbf{m}_{j_{i-1}} \in T).$$

The probability p that the points $\mathbf{m}_{j_1}, \ldots, \mathbf{m}_{j_k}$ are all genuine minutiae is given by

$$p = \prod_{i=1}^{k} \mathrm{P}^{(i-1)}(\mathbf{m}_{j_i} \in T \,|\, \mathbf{m}_{j_i} \in R). \tag{6}$$

We can estimate

$$\begin{aligned} &\mathrm{P}^{(i-1)}(\mathbf{m}_{j_i} \in T \,|\, \mathbf{m}_{j_i} \in R) \\ &\qquad = \frac{\mathrm{P}^{(i-1)}(\mathbf{m}_{j_i} \in T)}{\mathrm{P}^{(i-1)}(\mathbf{m}_{j_i} \in R)} \\ &\qquad = \left(1 + \frac{\mathrm{P}^{(i-1)}(\mathbf{m}_{j_i} \in R \setminus T)}{\mathrm{P}^{(i-1)}(\mathbf{m}_{j_i} \in T)}\right)^{-1} \end{aligned} \tag{7}$$

For the estimation of $\mathrm{P}^{(i-1)}(\mathbf{m}_{j_i} \in T)$, we again assume that the t genuine minutiae in T are successively selected by the probabilistic process **Select_T** introduced in Section IV-D.

Since $rV_d \ll n$, we can approximate that, on average, each point selected for R reduces the number of choices for the subsequent points by V_d. This estimation is actually an upper bound, which holds independently from our assumption $rV_d \ll n$, and we will use this bound to obtain an exact upper bound for the denominator in (7). For the nominator, however, we need a lower bound, and therefore, we use this estimation on the reduction of potential points as an approximation and not as a bound.

By (4), the probability that minutia $\mathbf{m}_{j_i}$ is chosen as the i-th minutia if $i-1$ genuine minutiae are already fixed is at most $(\psi_i-(i-1)V_d)^{-1}$. If $\mathbf{m}_{j_i}$ is not chosen as the i-th minutia, it can be selected as $(i+1)$-th minutiae only, if the i-minutiae has not been chosen from $B_d(\mathbf{m}_{j_i})$. Thus, the probability, that $\mathbf{m}_{j_i}$ is selected as $(i+1)$-th minutia, given that the first $i-1$ minutiae in T are fixed, is at most

$$\left(1-\frac{V_d}{\psi_i-(i-1)V_d}\right)\frac{1}{\psi_i-iV_d}=\frac{1}{\psi_i-(i-1)V_d}.$$

Analogously, for all $m \geq i$, the probability that $\mathbf{m}_{j_i}$ is selected as m-th minutia in T given that the first $i-1$ minutiae are fixed can be upper bounded by $(\psi_i-(i-1)V_d)^{-1}$. Thus, we obtain

$$\mathrm{P}^{(i-1)}(\mathbf{m}_{j_i} \in T) \leq \frac{t-i+1}{\psi_i-(i-1)V_d}. \quad (8)$$

For the estimation of the denominator of (7), we observe that $\mathbf{m}_{j_i}$ can be a chaff point only if none of the points in its d-sphere $B_d(\mathbf{m}_{j_i})$ is in T, i.e., if $B_d(\mathbf{m}_{j_i}) \cap T = \emptyset$. Thus, we have

$$\mathrm{P}^{(i-1)}(\mathbf{m}_{j_i} \in R \setminus T) = \mathrm{P}^{(i-1)}(Z) \cdot \mathrm{P}(\mathbf{m}_{j_i} \in R \,|\, Z), \quad (9)$$

where we have abbreviated the event $B_d(\mathbf{m}_{j_i}) \cap T = \emptyset$ by Z. Note, that the second probability does not depend on the condition that points $\mathbf{m}_{j_1}, \ldots, \mathbf{m}_{j_m}$ are genuine minutiae, because the specific configuration of T is irrelevant for the chances of $\mathbf{m}_{j_i}$ being selected as chaff point, as long as these points do not lie within the d-sphere of $\mathbf{m}_{j_i}$.

Using (8), we obtain

$$\begin{aligned}\mathrm{P}^{(i-1)}(Z) &\geq 1-\frac{(t-i+1)V_d}{\psi_i-(i-1)V_d}\\ &= \frac{\psi_i-tV_d}{\psi_i-(i-1)V_d}. \quad (10)\end{aligned}$$

On the other hand, we have

$$\mathrm{P}\left(\mathbf{m}_{j_i} \in R \,|\, Z\right) = \sum_{l=1}^{r-t} \mathrm{P}(M_l = \mathbf{m}_{j_i} \,|\, Z), \quad (11)$$

where M_l is the random variable defined by the selection of the l-th chaff point. Since we assume that the chaff points are chosen according to a uniform distribution and that each point selected for R reduces the number of choices for the subsequent points by V_d, the probability that the first chaff point chosen is $\mathbf{m}_{j_i}$, provided that none of the points in its d-sphere is in T, is approximately $(n-tV_d)^{-1}$.

If $\mathbf{m}_{j_i}$ is not chosen as the first chaff point, it can be selected as second chaff point only, if the first chaff point has not been chosen from $B_d(\mathbf{m}_{j_i})$. Thus, the probability, that $\mathbf{m}_{j_i}$ is selected as second chaff point, given that none of the points in its d-sphere is in T, is approximately

$$\left(1-\frac{V_d}{n-tV_d}\right)\frac{1}{n-(t+1)V_d}=\frac{1}{n-tV_d}.$$

Analogously, for all $m \geq 1$, the probability that $\mathbf{m}_{j_i}$ is selected as m-th chaff point, given that none of the points in its d-sphere is in T, can be approximated by $(n-tV_d)^{-1}$. Thus, we obtain from (11)

$$\mathrm{P}(\mathbf{m}_{j_i} \in R \,|\, Z) \approx \frac{r-t}{n-tV_d}, \quad (12)$$

and with (10) and (9)

$$\mathrm{P}^{(i-1)}(\mathbf{m}_{j_i} \in R \setminus T) \gtrapprox \frac{\psi_i-tV_d}{\psi_i-(i-1)V_d} \cdot \frac{r-t}{n-tV_d}. \quad (13)$$

Combining (7) with (8) and (13), we get

$$\begin{aligned}&\mathrm{P}^{(i-1)}(\mathbf{m}_{j_i} \in T \,|\, \mathbf{m}_{j_i} \in R)\\ &\qquad \lessapprox \frac{(t-i+1)(n-tV_d)}{(r-t)\psi_i+(t-i+1)n-t(r-i+1)V_d},\end{aligned}$$

and with (6) this yields the desired result. Q.E.D.

As in Theorem 6, let $1/\psi$ be the maximum likelihood of a minutiae location within $\mathcal{M}$. Then, by Lemma 8, the success probability for each individual trial of the smart polynomial reconstruction is approximately limited by

$$\prod_{i=1}^{k} \frac{(t-i+1)(n-tV_d)}{(r-t)\psi+(t-i+1)n-t(r-i+1)V_d}.$$

This general bound, together with the estimation of $117k\log^2(k)$ operations for a polynomial interpolation, provides an approximate lower bound for the average number of operations needed for the smart polynomial reconstruction.

Theorem 9. *For $rV_d \ll n$, the expected number W of operations for the smart polynomial reconstruction is $W = 117k\log^2(k)S$, where S is approximately lower bounded by*

$$S \gtrapprox \prod_{i=1}^{k} \frac{(r-t)\psi+(t-i+1)n-t(r-i+1)V_d}{(t-i+1)(n-tV_d)} \quad (14)$$

Note, that the estimation (14) depends on the number f of fingers used per person, as both n and ψ scale linearly with f. Precisely, the estimate for S decreases with f, i.e., the estimated workload of the attack is minimal for $f=1$. This dependency may surprise at first sight, but indeed, with increasing number of minutiae per finger they become more distinguishable from chaff points because less space in the area frequently assumed by minutiae is left for chaff points.

Estimating the probability ψ_i of minutiae occurrence at location $\mathbf{m}_{j_i}$ by ψ is of course quite rough. With

increasing K, the average of ψ_i for $j_i \leq K$ will decrease and so does the success probability. However, the rate of this decrease depends on the specific distribution of the minutiae locations and can only be determined on the basis of extensive data evaluation, which would go beyond the scope of this paper. Therefore, in Section VI-D, we will use Theorem 9 as a lower bound for workload of the smart polynomial reconstruction attack and complement this estimation by using the expected run time (5) of the conventional polynomial reconstruction as an upper bound.

3) Discussion: As explained in Section IV-E1, we are not able to provide a reasonably accurate run time estimation for the fingerprint dictionary attack, because theoretical analysis of the FAR is not possible without very simplifying assumptions. Therefore, the FAR needs to be determined empirically for each set of parameters used. Unfortunately, determination of very small FAR values is computationally very expensive: while the FAR for the multi-finger setting can be extrapolated from the FAR of a single-finger setting, determination of latter one requires considerably more than FAR^{-1} matching operations. Since security of the fuzzy fingerprint vault against the fingerprint dictionary attack requires a very low FAR, this task can be quite challenging.

A potential advantage of the fingerprint dictionary attack over polynomial reconstruction is that it takes optimal advantage of the actual statistical distribution of the feature vectors in the considered population. While the smart polynomial reconstruction attack exploits the non-uniformity of the minutiae locations in the considered area $\mathcal{M}$, the fingerprint dictionary attack can also take advantage from statistical dependencies among the minutiae locations. However, as discussed in Section IV-D, the dependencies reported in the literature are quite weak.

On the other hand, the effort N_{v} for each trial in the fingerprint dictionary attack is computationally expensive, as it comprises feature extraction, minutiae set alignment and Reed-Solomon decoding. For instance, we have implemented a matching algorithm that aligns the set of minutiae from the query fingerprints with the vault by determining the rotation and translation for optimal alignment (see [10] for details). For typical parameters the matching using this algorithm needs between 0.3 and 1 second on a standard PC. Of course, the matching process could be accelerated by using more sophisticated methods, but the alignment is definitely a complex task, which consumes considerable time. Moreover, the extraction of minutiae from (real or artificial) fingerprints requires extensive image pre-processing and edge detection, which is also very time consuming. In contrast, the run time estimations (5) and (14) of the polynomial reconstruction attacks counts elementary operations, i.e., additions, subtractions, multiplication or division over $\mathbf{F}_q$. In an implementation of the polynomial reconstruction attack reported in [12], the number of polynomials interpolated and tested per second of CPU time on a standard PC was greater than 8000 for $k = 14$. Based on the estimate $117k\log^2(k)$ for the number of operations needed per polynomial interpolation, and an optimistic estimate of 0.25 seconds of CPU time for a feature extraction and matching operation, we can roughly estimate that a single trial in the fingerprint dictionary attack takes 50 million times more computation time than the finite field operations counted in (5) and (14).

In this paper, we will subsequently estimate the practical security of the fuzzy fingerprint vault by the workload W of smart polynomial reconstruction attack, for which we use (5) as an upper and (14) as a lower bound. Nevertheless, we stress, that a security assessment of a concrete implementation of the fuzzy fingerprint vault should also comprise an empirical evaluation of the FAR and a resulting estimation for the workload of the fingerprint dictionary attack.

V. Optimization of Parameters

In this section we try to determine criteria for the optimal selection of parameters for both provable security and security against existing attacks. Furthermore, we derive estimates on the achievable security according to Theorems 1 and 7. We do this by estimating the maximum of E over t, k and r for a given decoding complexity.

A. Minimizing the fields size

In order to maximize the approximate lower bound for the remaining entropy according to Theorem 7, we set $q = r$; this minimization of the finite field has no influence on the security against existing attacks. Furthermore, since $n > \psi \gg tV_d$, we have $\binom{n/V_d}{t}/\binom{\psi/V_d}{t} \approx (n/\psi)^t$. In general, we cannot assume $r \gg t$; therefore, we use the approximation

$$\binom{r}{t} \approx r^t \left(1 - \frac{t-1}{2r}\right)^t /(t!),$$

which is much tighter than $\binom{r}{t} \approx r^t/(t!)$. With Stirling's approximation for $t!$, this results in the estimate

$$E \lessapprox k\log r - t\log\left(\frac{nt}{\mathrm{e}\psi}\left(1 - \frac{t-1}{2r}\right)\right) - \frac{1}{2}\log(2\pi t) - 2. \quad (15)$$

B. Selecting the minimum distance for minutiae

In (15), the remaining entropy is independent of the minimum distance d enforced for minutiae and chaff points. However, the parameter d limits the maximum r to approximately $1 \leq r \leq 0.45n/V_{\lceil d/2 \rceil}$, where the factor

0.45 represents the maximum density of a random sphere packing [5].

On the other hand, d should not be smaller than the tolerance parameter δ used for minutiae matching, to limit false matchings of minutiae in the query fingerprint with chaff points during authentication. Setting $d = 2\delta$ will already completely prevent such false matchings with minutiae that are also present in T, but smaller values might already reduce their number to a minimum. According to [10], setting $\delta \approx (3/2)d$ is a good compromise. In the following, we base our analysis on this choice for d and will use $0.45n/V_{\lceil (3/4)\delta \rceil}$ as maximum value for r.

C. Optimizing the degree of the polynomial

The parameter k must be set, so that with sufficient probability the secret polynomial can be recovered efficiently from a genuine query fingerprint. Subsequently we analyze the expected complexity of this task. As in Section IV-E1, we denote the number of *correct matches*, i.e., the matches between the query fingerprint and the genuine minutiae, with m_c, and the number of *false matches*, i.e., the matches between the query fingerprint and the chaff points, with m_f. From Section III-B3 we know that decoding is only possible if $m_\mathrm{c} \geq m_\mathrm{f} + k$.

It has been shown in [5] that, on average, the Reed-Solomon decoding of the polynomial using ℓ points requires

$$\binom{m_\mathrm{c}+m_\mathrm{f}}{\ell} \left(\sum_{i=\max(\lceil \frac{\ell+k}{2} \rceil, \ell-m_\mathrm{f})}^{\min(\ell,m_\mathrm{c})} \binom{m_\mathrm{f}}{\ell-i} \binom{m_\mathrm{c}}{i} \right)^{-1}$$

trials, where the parameter ℓ must fulfill $k \leq \ell \leq \min(2m_\mathrm{c} - k, m_\mathrm{c} + m_\mathrm{f})$. This expression is difficult to analyze theoretically. Numerical evaluation shows that for $m_\mathrm{c} - k \leq m_\mathrm{f} \leq m_\mathrm{c} + 2m_\mathrm{c}/(m_\mathrm{c} - k)$, the decoding complexity is minimized for $\ell = 2m_\mathrm{c} - k$. In this case, the sum collapses to the term for $i = m_\mathrm{c}$ and hence the minimum decoding complexity is

$$C_{\min}(m_\mathrm{c}, m_\mathrm{f}, k) = \binom{m_\mathrm{c}+m_\mathrm{f}}{2m_\mathrm{c}-k} \binom{m_\mathrm{f}}{m_\mathrm{c}-k}^{-1}. \qquad (16)$$

In the case $m_\mathrm{f} = m_\mathrm{c} - k$, we have $\ell = 2m_\mathrm{c} - k = m_\mathrm{c} + m_\mathrm{f}$ and $C_{\min}(m_\mathrm{c}, m_\mathrm{f}, k)$ evaluates to 1. For $m_\mathrm{f} = m_\mathrm{c} - k + i$ with $i = 1, 2, \ldots, m_\mathrm{c}/(m_\mathrm{c} - k) - 1$ equation (16) yields

$$C_{\min}(m_\mathrm{c}, m_\mathrm{f}, k) = \frac{(2m_\mathrm{c} - k + 1) \cdots (2m_\mathrm{c} - k + i)}{(m_\mathrm{c} - k + 1) \cdots (m_\mathrm{c} - k + i)}.$$

This equation shows that, for $m_\mathrm{c} - k \leq m_\mathrm{f} < m_\mathrm{c} - k + m_\mathrm{c}/(m_\mathrm{c} - k)$, the minimum decoding complexity increases exponentially with $i = m_\mathrm{f} - m_\mathrm{c} + k < m_\mathrm{c}/(m_\mathrm{c} - k)$. Numerical evaluation reveals that the exponential growth continues (with slowing pace) for $m_\mathrm{f} - m_\mathrm{c} + k \geq m_\mathrm{c}/(m_\mathrm{c} - k)$. Consequently, we find that the decoding complexity is an exponential function in $m_\mathrm{f} - m_\mathrm{c} + k$.

On the other hand, the number m_c of correct matches will typically disperse considerably between different authentications due to variations in the fingerprint image quality. Thus, if k is larger than the expectation of $m_\mathrm{c} - m_\mathrm{f}$, the fraction of cases, in which decoding is not feasible anymore, can become quite high. As a consequence, we set k to the expectation of $m_\mathrm{c} - m_\mathrm{f}$ in order to optimize the remaining entropy while limiting the decoding complexity.

Depending on the specific distribution of the number of correct matches and the requirements on decoding complexity imposed by the application scenario, it may be appropriate to select smaller or larger values for k. For instance, if the False Reject Rate (FRR) observed for a certain k is too high, k must be decreased until the FRR becomes acceptable. On the other hand, if the FRR is very low, k could be carefully increased. We will investigate the impact of increasing or decreasing k in our numerical evaluation in Section VI-C.

We estimate the mean values for m_c and m_f as follows:

- It is reasonable to assume that the average number of correct matches is a linear function of t, i.e., $m_\mathrm{c} = \mu t$, where μ is the average match rate independent of t.
- If $rV_\delta \ll n$, the number of points in $\mathcal{M}$ covered by the tolerance areas $B_\delta(\mathbf{m}_i)$ around the chaff points $\mathbf{m}_i$ can be estimated as $(r - t)V_\delta$. (Since minutiae of the query fingerprint that lie within the tolerance area of a chaff points can still be correctly matched with a minutiae in T, this estimate is even conservative.) Therefore, we can estimate the average number m_f of false matches by $sf(r - t)V_\delta/n$, where s is the average number of surplus minutiae per query fingerprint, i.e., the average number of minutiae in the query fingerprints that do not match with the stored minutiae, and f is the number of fingers used.

Remark: As the surplus minutiae are those *not matching with genuine minutiae*, their number depends on the match rate. Precisely, we could estimate the number s of surplus minutiae per finger as $s \approx w - \mu t$, where w is the average number of (all) minutiae per query fingerprint. However, this would result in a term t^2 in the estimation of E, which would render analytical determination of the maximum achievable entropy much more difficult. Furthermore, the number w of minutiae per finger is also not constant but depends on the feature extraction algorithm used and quality filtering applied, and hence, we would end up with the same number of variable parameters in our results.

As we set k to the expectation of $m_c - m_f$, these estimations yield

$$k = t\mu - (r-t)\frac{sfV_\delta}{n}. \quad (17)$$

Using approximation (15) this yields $E \lessapprox f(t,r)$ with

$$f(t,r) = \left(t\mu - (r-t)\frac{sfV_\delta}{n}\right)\log r - t\log\left(\frac{nt}{e\psi}\left(1-\frac{t-1}{2r}\right)\right) - \frac{1}{2}\log(2\pi t) - 2.$$

We also use (17) to eliminate parameter k from the estimations (5) and (14) for the workload W of the polynomial reconstruction attacks, which allows numerical optimization of t and r with respect to practical attacks in Section VI-D.

D. Maximizing the Bound for the Entropy

For fixed δ, n, μ, s and f, we try to estimate the maximum remaining entropy E by finding the maximum of the function $f(t,r)$ over r. The maximum is assumed, where the first derivation $\frac{\partial f(t,r)}{\partial r}$ is zero. It is easy to see that this is equivalent to $t^2 + a(r)t + b(r) = 0$ with $a(r) = 2\mu nr + sfV_\delta r(3+\ln(r))$ and $b(r) = -2sfV_\delta r^2(\ln(r)+1)$. For $r > 0$, one of the two solutions is negative and can thus be neglected. Consequently, for every r, $f(t,r)$ takes its maximum at

$$t_0(r) = -a(r)/2 + \sqrt{a(r)^2/4 - b(r)}.$$

Consequently, the function $f(t_0(r),r)$ upper bounds E for a given r, and the maximum of $f(t_0(r),r)$ over r yields a general upper bound for E. Thus, we can estimate the best provable security bound according to Theorems 1 and 4 that can be achieved for given δ, n, μ, s and f, by numerically determining the maximum of $f(t_0(r),r)$ over the relevant range of r. As argued in Section V-A, it is reasonable to set $d = \lceil (3/2)\delta \rceil$; hence, the relevant range is given by $1 \le r \le 0.45n/V_{\lceil (3/4)\delta \rceil}$ (see Section V-B), where the factor 0.45 represents the density of a random sphere packing [5].

Since for fixed $t, r \ge 1$, the value $f(t,r)$ is monotonically increasing with the match rate μ, we can determine the minimum value $\mu_{\min}$, for which the maximum of $f(t_0(r),r)$ is greater than a certain security level S. Since $E \lessapprox f(t_0(r),r)$, this value $\mu_{\min}$ is an approximate lower bound for the average match rate required to obtain a scheme with security 2^S according to Theorem 1 and Theorem 7, so that in the average case the polynomial can be recovered with one trial.

VI. Results

We evaluate whether and to what extent a (heuristically) provably secure fuzzy fingerprint vault is feasible. In particular, for different values for δ and for typical values for n, ψ and s we determine the minimum match rates required to achieve a security of 2^{50} according to Theorem 1 and Theorem 7. We compare these minimum match rates with match rates observed in practice.

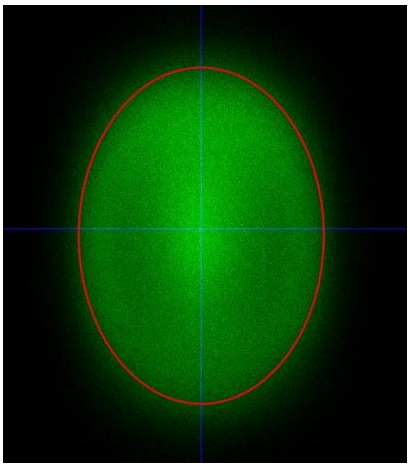

Figure 2. Spatial distribution of minutiae extracted with feature extractor MINDTCT [47] from 82800 fingerprints and the ellipse $\mathcal{E}$ from where minutiae are considered. The brightness of pixels corresponds to the frequency of minutiae occurrence at this position.

A. Evaluation of Minutiae Distribution

In order to estimate n and ψ, we have empirically determined the spatial distribution of minutiae within the fingerprint image. We evaluated the location of 5.8 million minutiae extracted with NIST's MINDTCT feature extraction algorithm [47] from 82800 imprints that were taken from 9200 fingers with 3 different sensors having 500 DPI. The fingerprints were taken from a non-public database set up in the course of a previous project of the German Federal Office for Information Security (Bundesamt für Sicherheit in der Informationstechnik). For this evaluation, the fingerprints had been pre-aligned, so that the center of mass of all minutiae coincides with the image center and the longest distance between two minutiae locations was vertically aligned.

It turned out that 83% of all minutiae occurred in an area defined by an ellipse that covers approximately 87000 pixels, which roughly corresponds to 2.25 cm^2. Outside this ellipse, the density of minutiae decreases drastically. Therefore, it is reasonable to restrict the fuzzy vault to minutiae and chaff points inside this area. This gives an estimate $n \approx 87000 \cdot f$, where f is the number of fingers, from which the minutiae are gathered. The distribution of the minutiae positions and the ellipse are shown in Figure 2. This yields $fs/n \approx s/87000$, which makes our estimation $E \lessapprox f(t,r)$ for the maximum achievable entropy bound E independent of the number f of fingers.

Table I
MINIMUM MATCH RATES REQUIRED TO ACHIEVE A PROVABLE SECURITY OF 50 BITS.

	$\delta = 5$	$\delta = 7$	$\delta = 10$
$s = 20$	82.2%	89.6%	97.0%
$s = 35$	87.9%	95.5%	-
$s = 50$	91.7%	99,1%	-

Table II
LINEAR FACTOR, BY WHICH THE MINIMUM MATCH RATES GIVEN IN TABLE I DECREASE WITH INCREASING k.

	$\delta = 5$	$\delta = 7$	$\delta = 10$
$s = 20$	0.67%	0.54%	0.41%
$s = 35$	0.54%	0.38%	-
$s = 50$	0.45%	0.28%	-

The maximum frequency of a minutiae location was 112, which corresponds to a maximum probability of a minutiae location inside the ellipse of approximately $112/5800000/0.83 \approx 2^{-15.4}$. This results in an approximation $n/\psi \approx 2$. This approximation is independent from the number f of fingers used for the fuzzy vault, as both ψ and n scale linearly with f.

We stress that our estimate is valid for minutiae extracted with the MINDTCT algorithm. As shown in [54], other feature extractors exhibit considerably different minutiae placement density functions, and thus, the maximum probability of a minutiae location may differ as well.

B. Estimating the number of surplus minutiae

According to [4], a good-quality live-scan fingerprint has 20–70 minutiae. Since $f(t, r)$ decreases with an increasing average number s of minutiae per query fingerprint not matching with genuine minutiae, it might be a good idea to use only the most reliable minutiae of the query fingerprints, e.g., by evaluating minutiae quality indices output by the feature extraction algorithm. However, the extent of the filtering should be carefully balanced with the match rates achieved with the reduced number of minutiae. We will subsequently consider the range $20 \leq s \leq 50$.

C. Numerical Parameter Optimization for Provable Security

In the previous sections, we found the approximations $n/\psi \approx 2$ and $fs/n \approx s/87000$ from empirical data. Using these estimations and various values for δ and s, we applied the method described in Section V-D to determine the minimum match rate required to achieve a security level of 2^{50} according to Theorem 1 and Theorem 7. We numerically computed the maximum value of the function $f(t_0(r), r)$ over the range $1 \leq r \leq 0.45n/V_{\lceil (3/4)\delta \rceil}$ (see Section V-B) with the maximum value $n = 10 \cdot 87000$, i.e., for maximum range of chaff points possible for 10 fingers, using the computer algebra system PARI/GP. The minimum match rates, at which this maximum exceeds 2^{50}, are listed in Table I for different values of δ and s. A "–" denotes that a remaining entropy of 50 is not achieved at all.

The security bounds are very sensitive to changes of the match rate. For instance, for the parameters given in Table I, a decrease of the match rate by only 2% results in a reduction of the achievable security of 12 to 38 bits; a larger reduction is observed for higher match rates.

As explained in Section V-C, under specific circumstances it may be reasonable to select k greater than our choice $k_0 := t\mu - (r - t)sfV_\delta/n$, particularly if the dispersion of the number of correct matches is small, or if a larger decoding complexity is acceptable. On the other hand, if the False Reject Rate (FRR) observed for $k = k_0$ is too high or the decoding of the polynomial takes too much time, k must be decreased. Any decrease of k from the assumed optimal value k_0 results in an increase of the minimum match rate required for a certain security level, and any increase of k results in a decrease of the minimum match rate. In particular, setting $k = k_0 + \epsilon$ with $\epsilon > 0$ increases the entropy estimation 15 by $\epsilon \log(r)$. For a given match rate μ, this results in the same amount of entropy as setting $k = k_0$ with match rate $\mu + \epsilon/t$. Thus, for a given security level, decreasing k by ϵ compensates an decrease of the match rate by ϵ/t. As a consequence, the minimum match rates required for a security level of 2^{50} with $k = k_0 + \epsilon$ can be estimated by subtracting $\epsilon/t_{\max}$ from the values given in Table I, where $t_{\max}$ is the value of $t_0(r)$, for which $f(t_0(r), r)$ is maximal. Analogously, the minimum match rates with $k = k_0 - \epsilon$ can be estimated by adding $\epsilon/t_{\max}$ to the values given in Table I. We give the respective values of $1/t_{\max}$ in Table II.

We give an example how Table II can be used: According to Table I, for $\delta = 5$ and $s = 35$, at least a match rate $\mu = 87.9\%$ is required to achieve a security of 2^{50}, given that we set $t\mu - (r - t)sfV_\delta/n$ and select r and $t = t_0(r)$ so that $f(t, r)$ is maximized. However, if the False Reject Rate (FRR) observed for these parameters is too high or the decoding of the polynomial takes too much time, k must be decreased. If setting $k = t\mu - (r - t)sfV_\delta/n - 3$ results in an acceptable FRR and decoding performance, we have $\epsilon = 3$. This decrease of k implies that the match rate has to be at least $\mu' = 87.9\% + 3 \cdot 0.54\% = 89.52\%$ to achieve the desired security of 2^{50}, where the value 0.54 is taken from Table II.

To get a feeling for the number of minutiae and thus for the number of fingers needed for a provable secure scheme, we evaluate the minimum value t, for which we still obtain a remaining entropy of 2^{50} for a given μ. For this evaluation we apply the following method.

First, we observe that $t_0(r)$ is continuous and unbounded for $r > 0$ and is zero for $1/\mathrm{e}$. Thus, for every $t' > 0$ there is a r' with $t' = t_0(r')$; by definition of $t_0(r)$, this pair (t', r') maximizes the function $f(t, r')$ over t. Consequently, it suffices to search through all pairs $(t_0(r), r)$ to find the minimal t with $f(t, r) \geq 2^{50}$.

On the other hand, the approximation of the remaining entropy E by the continuous function $f(t, r)$ will result in an artificially smooth curve for the minimal t. In particular, in the definition of f we have replaced k by a real number, whereas in practice, k can only take integer values. The small deviations of the truncated integer k from its real valued optimum imply a corresponding deviation of the achievable security E and hence, of the minimal t required for a certain value of E. To obtain a more realistic estimation of the minimal t, we set $k_0(r) = \lfloor t_0(r)\mu - (r - t_0(r))sfV_\delta/n \rfloor$ and determine the minimal $t_0(r)$ for that (15) yields at least a value of $E \geq 2^{50}$ with $t = t_0(r)$ and $k = k_0(r)$.

Figure 3 shows the minimal number t of minutiae required for a security of 2^{50} as a function of the average match rate μ for various parameters δ and s.

These curves also allow estimating the impact of selecting a larger k to the minimum value t of minutiae. As explained above, selecting $k = k_0(r) + \epsilon$ compensates a decrease of the match rate by ϵ/t, and analogously, choosing $k = k_0(r) - \epsilon$ equates an increase of the match rate by ϵ/t. Therefore, for small ϵ, the minimum value of t yielding a security of 2^{50} with $k = k_0(r) \pm \epsilon$ and a match rate μ can be estimated as the value of t corresponding to $\mu \mp \epsilon/t_0$ in Figure 3, where t_0 is the value of t indicated in Figure 3 for μ.

We give an example: for $\delta = 5$ and $s = 35$, a match rate $\mu = 0.9$ requires at least $t = 68$ minutiae in the template. If for this t, the corresponding optimal r (maximizing function $f(t, r)$, i.e., the r with $t = t_0(r)$) and $k_0 = \lfloor t\mu - (r - t)sfV_\delta/n \rfloor$, the False Reject Rate (FRR) observed is too high, k must be decreased until the FRR becomes acceptable. If setting $k = \lfloor t\mu - (r-t)sfV_\delta/n \rfloor - 3$ results in an acceptable FRR, we have $\epsilon = 3$ and, using Table II, obtain a minimum match rate of $\mu' = 90\% + 3 \cdot 0.54\% = 91.62\%$. For this value, we get from Figure 3 that only a minimum of $t = 58$ minutiae are required. We have taken the exact values from our evaluation data, from which the curves in Figure 3 have been drawn.

D. Numerical Parameter Optimization for Practical Security

In this section, we determine the minimal number of minutiae required for a given match rate μ and fixed parameters δ and s, for which a security of 2^{66} can be achieved against existing attacks. We do this by numerically evaluating the estimates from (5) and Theorem 9 for the expected number of operations needed for polynomial reconstruction. Precisely, for each t we maximize the estimates for W according to (5) and Theorem 9, respectively, with respect to r over the relevant range $t + 1 \leq r \leq 0.45n/V_{\lceil (3/4)\delta \rceil}$ (see Section V-B), and identify the minimal t, for which an r from this range exists so that the respective security exceeds 2^{66}. Again, we deploy the computer algebra program PARI/GP.

Figure 4 shows the dependency of the minimal number of minutiae in the template required to achieve a security of 2^{66} against the polynomial non-optimized attack proposed in [12] according to (5). This estimation (5) also provides an upper bound for the workload of the smart polynomial reconstruction attack, and hence, the minimum number of minutiae indicated in Figure 4 provides a lower bound for the minimum number of minutiae needed to ensure the same security level of 2^{66} with respect to the smart polynomial reconstruction, which we consider as the best known attack. An approximate lower bound of the security against this attack is given by Theorem 9, and consequently, this estimation can be used to determine an upper bound for the number of minutiae needed to achieve a certain security level. Figure 5 shows the dependency of the minimal number of minutiae in the template required to achieve a security of 2^{66} against the smart polynomial reconstruction attack based on the estimation of Theorem 9. Since our empirical results in [10] indicates that the number of minutiae required according to Figure 4 can only be obtained by used $f \geq 2$ fingers per person, and as the estimate of Theorem 9 decreases with f, we will assume $f = 2$ (the estimate (5) is independent of f).

For all considered parameters, the optimal value for k is between 18 and 49. This implies that the number $117k \log^2(k)$ of operations needed for a polynomial interpolation (see Section IV-E2) is between $2^{15.1}$ and $2^{17.5}$, and thus, the 2^{66} operations used as security level roughly correspond to 2^{50} trials. As a consequence, our security bound considered for practical attacks is comparable to the security bound used for provable security in Section VI-C.

An interesting observation is that increasing the number of chaff points does not generally increase security: for each t, there is an optimal value for parameter r, for which the estimates given by (5) and Theorem 9 are maximized. If r is increased further, the estimated workload of the polynomial reconstruction attack for optimally chosen k, decreases. This observation can be explained by the influence of r on the number of false matches. Specifically, the average number of false matches increases linearly with the number $(r-t)$ of chaff points. This increased number of false matches requires a smaller value for parameter k in order to ensure efficient

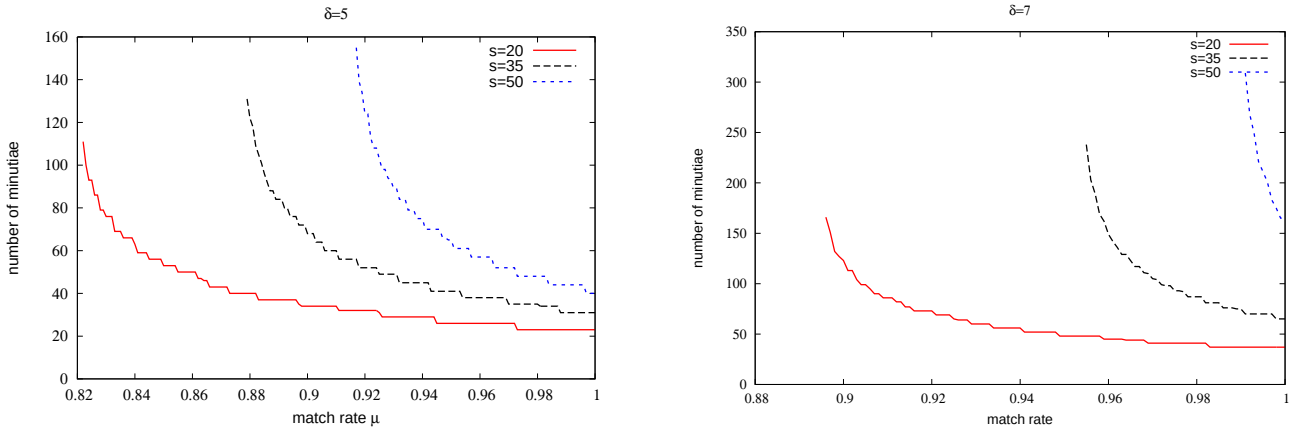

Figure 3. Dependency of the minimal number t of minutiae on the average match rate μ for a security of $E \geq 2^{50}$, for (a) $\delta = 5$ and (b) $\delta = 7$, respectively, and different numbers s of surplus minutiae per finger.

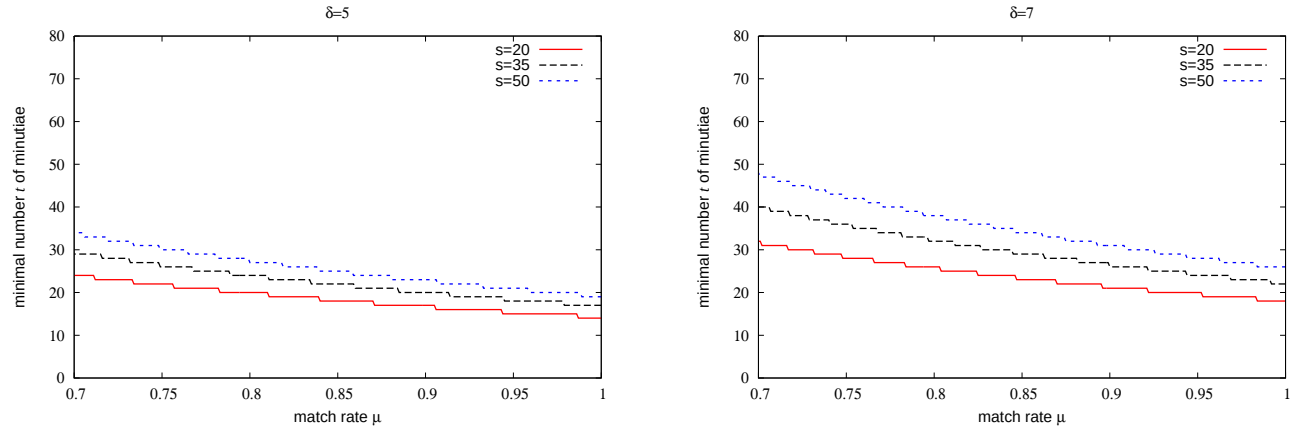

Figure 4. Dependency of the minimal number t of minutiae on the average match rate μ for $\delta = 5$ and a security of 2^{66} against the non-optimized polynomial reconstruction attack of [12] for (a) $\delta = 5$ and (b) $\delta = 7$ and for different numbers s of surplus minutiae per finger.

decoding of the polynomial, and a smaller k reduces the workload of the attack significantly.

Our results allow a critical review of our assumptions $r \ll 0.45n/V_{\lceil d/2 \rceil}$ and $r \ll n/V_d$ used in Section IV. For all parameters μ and s considered, the optimal r fulfills $r < 408$ for $\delta = 5$ and $r < 312$ for $\delta = 5$. This implies that, for $\delta = 5$ and $f \geq 2$, the optimal r is by a factor 4 smaller than the maximum value $0.45n/V_{\lceil d/2 \rceil}$ with $d \approx (3/2)\delta$ (see Section V-B), and for $\delta = 7$ and $f \geq 3$, it is by a factor 3.5 smaller than the maximum value. Consequently, at least for $\delta = 5$ and $f \geq 2$, or for $\delta = 7$ and $f \geq 3$, respectively, our assumption that the attack method of [11] is not very efficient is justified. Unfortunately, the validity of our assumption $rV_d \ll n$ used for the proof of Lemma 8 is less clear: for $\delta = 5$ and $f = 2$ as well as for $\delta = 7$ and $f = 3$, the fraction $n/(rV_d)$ is approximately 2.2. Thus, unless r is chosen considerably smaller than its optimum, which is possible by increasing the number t of minutiae, we must expect some inaccuracy in our approximation that, on average, each selected chaff point reduces the number of choices for the subsequent points by V_d. This inaccuracy propagates to the estimation of Theorem 9. On the other hand, our upper bound $1/\psi$ for the probability of a minutiae occurring at location $\mathbf{m}_{j_i}$ for all minutiae $\mathbf{m}_{j_i}$ processed by the smart polynomial reconstruction attack is very conservative, because from Figure 2 we can see that the area with the highest frequency of minutiae occurrence, which is the bright area in the center, is quite small. Therefore, we are very confident that Theorem 9 still underestimates the number of trials needed for the smart polynomial reconstruction attack.

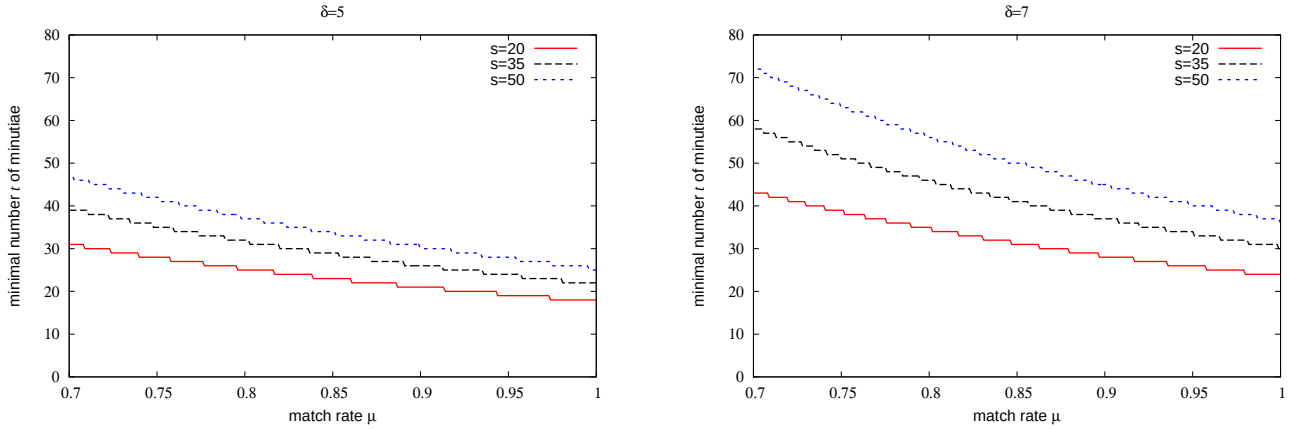

Figure 5. Dependency of the minimal number t of minutiae on the average match rate μ for $\delta = 5$ and a security of 2^{66} against the smart polynomial reconstruction attack presented in Section IV-E2 for (a) $\delta = 5$ and (b) $\delta = 7$ and for different numbers s of surplus minutiae per finger.

E. Comparison with Empirical Data

The minimum match rates required for provable security are quite high. According to [4], matchings conducted by a human expert results in rates of approximately 90%. Automatic matching algorithms only operating on the minutiae data will yield considerably lower rates, depending on the method, by which minutiae correspondences are identified and the matching tolerance applied. For instance, the distribution of the distance of matching minutiae reported in literature (see [55]) implies that a tolerance (with respect to Euclidean distance) of $\delta < 10$ will significantly reduce the match rates. The presence of chaff points will further reduce the performance of minutiae matcher algorithms.

On the other hand, the match rates can be greatly enhanced by using several minutiae measurements (per finger) during enrollment to minimize measurement noise. For instance, in [10], we present an implementation of the fuzzy fingerprint vault, which applies several minutiae measurements per finger during enrollment and uses only those minutiae that have been detected in all measurements. Furthermore, the minutiae locations are set to the mean value over the measurements. Empirical evaluations show that this considerably increases the average match rates, but, at the same, reduces the number of minutiae available per finger. Hence, several fingers must be used to achieve the minimum values for t indicated in Figure 3. Furthermore, the dispersion found in the match rates is large so that, in order to limit the False Rejection Rate, is seems necessary to choose k smaller than the expectation of $m_c - m_f$ (see Section V-C), which further increases the required minimum match rates as shown in Table II.

We found minutiae quality filtering during authentication using quality indices provided by the feature extractor to be quite effective to reduce the number s of surplus (non-matching) minutiae per query fingerprint and, consequently, the number of false matches. However, the filtering should not exceed a certain extent in order to avoid disproportional reduction of the match rate. Furthermore, the false match rates observed were 20% to 60% higher than our estimation $sf(r-t)V_\delta/n$ in Section V-C predicts. This effect is presumably due to failures in the alignment of the query fingerprint to the vault.

Based on the observed statistics on match rates and number of reliable minutiae, and given the results in Table I, we conclude that provable security seems out of reach, unless the average number of surplus minutiae per query fingerprint can be further reduced by improved quality filtering methods. Details on the empirical data and our interpretation are given in [10].

On the other hand, our evaluation in [10] shows that strong security (comparable to 64 bit keys) against the (non-optimized) polynomial reconstruction attack can be achieved using 2 fingers per individual. Considering our improvement by the smart polynomial reconstruction, an additional security margin should be added. Based on the data provided in [10] we can estimate that the same security can be achieved against our optimized smart polynomial reconstruction attack using 3 fingers per user. As explained in Section IV-E, the number of chaff point should not be too close to the maximum possible to render the attack method described in [11] inefficient, and the FAR should be determined for the chosen parameters to allow estimation of the effort for a fingerprint dictionary attack.

VII. Conclusions

Our analysis shows that a provably secure fuzzy fingerprint vault can hardly be achieved in practice. The required rate of minutiae in the vault matching with those in the query fingerprints so high that it seems only achievable by powerful quality filtering during enrollment. However, this filtering approach conflicts with the requirement for a large number of minutiae in the vault. Given the empirical data on match rates in the literature, in particular our analysis in [10], provable security seems out of reach.

The usage of minutiae orientations as additional discriminating data could surely increase the information content of the templates. However, minutiae directions bear strong dependencies with their spatial location and with directions of nearby minutiae: according to [45], "minutiae points in different regions of the fingerprint domain are observed to be associated with different region-specific minutiae directions", and "minutiae points that are spatially close tend to have similar directions with each other". Consequently, a template using both spatial location and orientation of minutiae contains considerable redundancy and makes an analysis of the entropy of the feature vector very difficult. Moreover, since the estimated entropy loss in the security bounds in Section IV-C increases linearly with the number of bits in the template, this decreases the percentage of entropy that actually contributes to the provable security estimates.

On the other hand, our investigation of the most efficient attack methods indicates that the theoretical lower bounds for security are far from being tight. The underlying computational problem (polynomial reconstruction) is believed to be hard and has been repeatedly proposed as a basis for the security of cryptographic techniques [52]. As a consequence, the match rates and number of minutiae required to achieve security against the existing attacks are much lower than the numbers for provable security. Still, the empirical data presented in [10] show that at least two fingers per user must be used to achieve a level of security equivalent to a 50 bit cryptographic key. However, we stress that there is no evidence that our optimized polynomial reconstruction method is indeed the most efficient attack. In particular, before an implementation of the fuzzy fingerprint vault can be claimed to be secure, it must be verified that the False Accept Rate (FAR) is in a range that ensures that a fingerprint dictionary attack is inefficient. Unfortunately, this requires a very large number of impostor matches, precisely, in the range of 1/FAR. The numbers of fingers used in existing investigations for the fuzzy fingerprint vault, in particular in [6], [7], [8], [9] and [10], were far too small to assess if an adequate security level against the fingerprint dictionary attack can be achieved.

Summarizing, our results seem to indicate that although a provable secure fuzzy fingerprint vault is out of reach, it can provide sufficient security against practical attacks if several fingers are used.

Finally, secure biometric template protection schemes may also be achievable using completely different constructions. For instance, there exist approaches to apply the fuzzy commitment scheme to fingerprints. As shown in [23], the entropy loss in the fuzzy commitment is much lower than in the fuzzy vault. However, since the fuzzy commitment scheme only tolerates errors that are small with respect to the Hamming metric [13], sophisticated encoding and signal processing techniques must be applied to compensate spatial rotations and translations of the fingerprint, as well as permutations, detections and insertions of the detected minutiae. Several promising techniques have been proposed, in particular, usage of fingerprint ridge patterns as biometric feature [20][56], transformation of the minutiae data to the frequency domain [57] and using the characteristic vector of minutiae occurrence with respect to a grid [58][59]. However, we are not aware of any comprehensive security analysis for these approaches based on estimations for the feature vector's entropy and the error correction required without manual alignment of the fingerprints.

Acknowledgments

This work was conducted as part of the project "BioKeyS Pilot-DB" of the German Federal Office for Information Security (Bundesamt für Sicherheit in der Informationstechnik).

We would like to thank Patrick Jäger for his assistance in the numerical calculations and visualization.

References

[1] J. Merkle, M. Niesing, M. Schwaiger, H. Ihmor, and U. Korte, "Provable security for the fuzzy fingerprint vault," in *Proceedings of the 5th International Conference Internet Monitoring and Protection (ICIMP 2010)*, J. L. Mauri and M. Popescu, Eds. IEEE Computer Society, 2010, pp. 65–73.

[2] J. Breebaart, C. Busch, J. Grave, and E. Kindt, "A reference architecture for biometric template protection based on pseudo identities," in *BIOSIG 2008: Biometrics and Electronic Signatures*, ser. Lecture Notes in Informatics, A. Brömme and C. Busch, Eds., vol. 137. Gesellschaft für Informatik, 2008, pp. 25–38.

[3] A. Juels and M. Sudan, "A fuzzy vault scheme," in *Proceedings of the 2002 IEEE International Symposium on Information Theory*. IEEE, 2002, p. 408.

[4] U. Uludag, S. Pankanti, S. Prabhakar, and A. Jain, "Biometric cryptosystems: Issues and challenges," *Proceedings of the IEEE*, vol. 92, no. 6, pp. 948–960, 2004.

[5] C. Clancy, N. Kiyavash, and D. Lin, "Secure smartcard-based fingerprint authentication," in *Proceedings of the 2003 ACM SIGMM Workshop on Biometrics Methods and Applications (WBMA '03)*. ACM, 2003, pp. 45–52.

[6] U. Uludag, S. Pankanti, and A. Jain, "Fuzzy vault for fingerprints." in *Proceedings of the 5th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, ser. Lecture Notes in Computer Science, vol. 3546. Springer, 2005, pp. 310–319.

[7] U. Uludag and A. Jain, "Securing fingerprint template: Fuzzy vault with helper data," in *Proceedings of the 2006 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2006), Workshop on Privacy Research In Vision*, C. Schmid, S. Soatto, and C. Tomasi, Eds. IEEE Computer Society, 2006, pp. 163–169.

[8] K. Nandakumar, A. Jain, and S. Pankanti, "Fingerprint-based fuzzy vault: Implementation and performance." *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 4, pp. 744–757, 2007.

[9] J. Li, X. Yang, J. Tian, P. Shi, and P. Li, "Topological structure-based alignment for fingerprint fuzzy vault," in *Proceedings of the 19th International Conference on Pattern Recognition (ICPR '08)*. Omnipress, 2008, pp. 1–4.

[10] J. Merkle, M. Niesing, M. Schwaiger, H. Ihmor, and U. Korte, "Performance the fuzzy vault for multiple fingerprints (short version)," in *BIOSIG 2010: Biometrics and Electronic Signatures*, ser. Lecture Notes in Informatics, A. Brömme and C. Busch, Eds., vol. P-164. Gesellschaft für Informatik, 2010, pp. 57–72.

[11] E.-C. Chang, R. Shen, and F. W. Teo, "Finding the original point set hidden among chaff," in *Proceedings of the 2006 ACM Symposium on Information, Computer & Communication Security (ASIACCS)*. ACM, 2006, pp. 182–188.

[12] P. Mihailescu, A. Munk, and B. Tams, "The fuzzy vault for fingerprints is vulnerable to brute force attack," in *BIOSIG 2009: Biometrics and Electronic Signatures*, ser. Lecture Notes in Informatics, A. Brömme and C. Busch, Eds., vol. 155. Gesellschaft für Informatik, 2009, pp. 43–54.

[13] Y. Dodis, R. Ostrovsky, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data." *SIAM Journal on Computing*, vol. 38, no. 1, pp. 97–139, 2008.

[14] C. Soutar, D. Roberge, A. Stoianov, R. Gilroy, and B. V. Kumar, "Biometric encryption using image processing," in *Proceedings of the SPIE, Optical Security and Counterfeit Deterrence Techniques II*, R. L. van Renesse, Ed., vol. 3314, 1998, p. 178Ű188.

[15] N. K. Ratha, J. H. Connell, and R. M. Bolle, "Enhancing security and privacy of biometric-based authentication systems," *IBM Systems Journal*, vol. 40, 2001.

[16] J.-P. Linnartz and P. Tuyls, "New shielding functions to enhance privacy and prevent misuse of biometric templates," in *Proceedings of the 4th international conference on Audio- and video-based biometric person authentication (AVBPA'03)*. Springer, 2003, pp. 393–402.

[17] Y. Chang, W. Zhang, and T. Chen, "Biometrics-based cryptographic key generation," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME '04)*. IEEE Computer Society, 2004, pp. 2203–2206.

[18] C. Chen, R. Veldhuis, T. Kevenaar, and A. Akkermans, "Multibits biometric string generation based on the likelyhood ratio," in *Proceedings of the IEEE Conference on Biometrics: Theory, Applications and Systems (BTAS '07)*. IEEE Computer Society, 2007, pp. 1–6.

[19] A. Juels and M. Wattenberg, "A fuzzy commitment scheme," in *Proceedings of the 6th ACM Conference on Computer and Communication Security*. ACM, 1999, pp. 28–36.

[20] U. Martini and S. Beinlich, "Virtual PIN: Biometric encryption using coding theory," in *BIOSIG 2003: Biometrics and Electronic Signatures*, ser. Lecture Notes in Informatics, A. Brömme and C. Busch, Eds., vol. 31. Gesellschaft für Informatik, 2003, pp. 91–99.

[21] T. A. M. Kevenaar, G. J. Schrijen, M. van der Veen, A. H. M. Akkermans, and F. Zuo, "Face recognition with renewable and privacy preserving binary templates," in *Proceedings of the Fourth IEEE Workshop on Automatic Identification Advanced Technologies*. IEEE Computer Society, 2005, pp. 21–26.

[22] F. Hao, R. Anderson, and J. Daugman, "Combining crypto with biometrics effectively," *IEEE Transaction on Computers*, vol. 55, pp. 1081–1088, 2006.

[23] U. Korte, M. Krawczak, J. Merkle, R. Plaga, M. Niesing, C. Tiemann, H. Vinck, and U. Martini, "A cryptographic biometric authentication system based on genetic fingerprints," in *Sicherheit 2008 - Sicherheit, Schutz und Zuverlässigkeit*, ser. Lecture Notes in Informatics, A. Alkassar and J. Siekmann, Eds., vol. 128. Gesellschaft für Informatik, 2008, pp. 263–276.

[24] Y. Lee, K. Bae, S. Lee, K. Park, and J. Kim, "Biometric key binding: Fuzzy vault based on iris images," in *Advances in Biometrics*, ser. Lecture Notes in Computer Science, S.-W. Lee and S. Li, Eds. Springer, 2007, vol. 4642, pp. 800–808.

[25] A. Juels and M. Sudan, "A fuzzy vault scheme," *Designs, Codes and Cryptography*, vol. 38, pp. 237–257, 2006.

[26] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *Advances in Cryptology - EUROCRYPT 2004*, ser. Lecture Notes in Computer Science, C. Cachin and J. Camenisch, Eds., vol. 3027. Springer-Verlag, 2004, pp. 523–540.

[27] S. Yang and I. Verbauwhede, "Automatic secure fingerprint verification system based on fuzzy vault scheme," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'05)*, 2005, pp. 609–612.

[28] P. Li, X. Yang, K. Cao, X. Tao, R. Wang, and J. Tian, "An alignment-free fingerprint cryptosystem based on fuzzy vault scheme," *Journal of Network and Computer Applications*, vol. 33, pp. 207–220, 2010.

[29] A. Nagar, K. Nandakumar, and A. K. Jain, "Securing fingerprint template: Fuzzy vault with minutiae descriptors," in *Proceedings of the 19th International Conference on Pattern Recognition (ICPR '08)*. Omnipress, 2008, pp. 1–4.

[30] W. J. Scheirer and T. E. Boult, "Cracking fuzzy vaults and biometric encryption," in *Proceedings of the Biometrics Symposium 2007*, 2007, pp. 1–6.

[31] A. Kholmatov and B. Yanikoglu, "Realization of correlation attack against the fuzzy vault scheme," in *Proceedings of the SPIE, Security, Forensics, Steganography, and Watermarking of Multimedia Contents X*, E. J. Delp, Ed., vol. 6819, 2008, pp. 68 190O–68 190O–7.

[32] K. Nandakumar, A. Nagar, and A. K. Jain, "Hardening fingerprint fuzzy vault using password," in *Proceedings of Advances in Biometrics, International Conference (ICB 2007)*, ser. Lecture Notes in Computer Science, S.-W. Lee and S. Z. Li, Eds., vol. 4642. Springer, 2007, pp. 927–937.

[33] J. Gu and J. Zhou, "A novel model for orientation field of fingerprints," in *Proceeding of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2003)*. IEEE Computer Society, 2003, pp. 2–493.

[34] S. C. Dass, Y. Zhu, and A. K. Jain, "Statistical models for assessing the individuality of fingerprints," *IEEE Transactions on Information Forensics and Security*, vol. 2, pp. 391–401, 2007.

[35] R. Plaga, "Biometric keys: Suitable uses and achievable information content," *International Journal on Information Security*, vol. 8, no. 6, pp. 447–454, 2009.

[36] T. Kevenaar, U. Korte, J. Merkle, M. Niesing, H. Ihmor, C. Busch, and X. Zhou, "A reference framework for the privacy assessment of biometric encryption systems," in *BIOSIG 2010: : Biometrics and Electronic Signatures*, ser. Lecture Notes in Informatics, A. Brömme and C. Busch, Eds., vol. P-164. Gesellschaft für Informatik, 2010, pp. 45–56.

[37] R. Canetti, "Towards realizing random oracles: Hash functions that hide all partial information," in *Advances in Cryptology - CRYPTO '97*, ser. Lecture Notes in Computer Science, B. S. Kaliski, Ed., vol. 1294. Springer, 1997, pp. 455–469.

[38] M. Liu, X. Jiang, and A. C. Kot, "Fingerprint reference point detection," *EURASIP Journal on Applied Signal Processing*, vol. 2005, pp. 498–509, 2005.

[39] P. Tuyls and J. Goseling, "Capacity and examples of template-protecting biometric authentication systems," in *Biometric Authentication: ECCV 2004 International Workshop, BioAW 2004*. Springer, 2004, pp. 158–170.

[40] A. Bhargav-Spantzel, A. Squicciarini, and E. Bertino, "Privacy preserving multi-factor authentication with biometrics," in *DIM '06: Proceedings of the Second ACM Workshop on Digital Identity Management*. ACM, 2006, pp. 63–72.

[41] T. Ignatenko, "Secret-key rates and privacy leakage in biometric systems," Ph.D. dissertation, Eindhoven University of Technology, 2009.

[42] J. O. Pliam, "On the incomparability of entropy and marginal guesswork in brute-force attacks," in *Progress in Cryptology - INDOCRYPT 2000*, B. K. Roy and E. Okamoto, Eds. Springer, 2000, pp. 67–79.

[43] I. R. Buhan, "Cryptographic keys from noisy data, theory and applications," Ph.D. dissertation, University of Twente, 2008.

[44] S. Pankanti, S. Prabhakar, and A. Jain, "On the individuality of fingerprints," *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, vol. 1, p. 805, 2001.

[45] Y. Zhu, S. C. Dass, and A. Jain, "Statistical models for assessing the individuality of fingerprints." *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3-1, pp. 391–401, 2007.

[46] J. Chen and Y. S. Moon, "A statistical study on the fingerprint and minutiae distribution," in *Proceedings of the 2006 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, P. Duhamel and L. Vandendorpe, Eds. IEEE Computer Society, 2006, pp. 169–172.

[47] C. Watson, M. Garris, E. Tabassi, C. Wilson, M. McCabe, S. Janet, and K. Ko, *User's Guide to NIST Biometric Image Software (NBIS)*, National Institute of Standards and Technology, 2007.

[48] D. A. Stoney, "Distribution of epidermal ridge minutiae," *American Journal of Physical Anthropology*, vol. 77, pp. 367–376, 1988.

[49] S. Z. Li and A. K. Jain, Eds., *Encyclopedia of Biometrics*. Springer US, 2009.

[50] A. Kiayias and M. Yung, "Cryptographic hardness based on the decoding of reed-solomon codes," *IEEE Transactions on Information Theory*, vol. 54, no. 6, pp. 2752–2769, 2008.

[51] V. Guruswami and A. Vardy, "Maximum-likelihood decoding of reed-solomon codes is NP-hard," *IEEE Transactions on Information Theory*, vol. 51, no. 7, pp. 2249–2256, 2005.

[52] A. Kiayias and M. Yung, "Directions in polynomial reconstruction based cryptography," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E87-A, no. 5, pp. 978–985, 2004.

[53] J. v. z. Gathen and J. Gerhard, *Modern Computer Algebra*, 2nd ed. Cambridge University Press, 2003.

[54] E. Tabassi, P. Grother, W. Salamon, and C. Watson, "Minutiae interoperability," in *BIOSIG 2009: Biometrics and Electronic Signatures*, ser. Lecture Notes in Informatics, A. Brömme and C. Busch, Eds., vol. P-155. Gesellschaft für Informatik, 2009, pp. 13–30.

[55] A. Jain, S. Prabhakar, and S. Pankanti, "On the individuality of fingerprints," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001)*. IEEE Computer Society, 2001, pp. I:805–812.

[56] P. Tuyls, A. Akkermans, T. Kevenaar, G. J. Schrijen, A. Bazen, and R. Veldhuis, "Practical biometric authentication with template protection." in *Proceedings of the 5th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, ser. Lecture Notes in Computer Science, T. Kanade, A. K. Jain, and N. K. Ratha, Eds., vol. 3546. Springer, 2005, pp. 436–446.

[57] H. Xu and R. N. Veldhuis, "Spectral minutiae representations for fingerprint recognition," in *Proceedings of the Sixth International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP 2010)*, I. Echizen, J.-S. Pan, D. Fellner, A. Nouak, A. Kuijper, and L. C. Jain, Eds. IEEE Computer Society, 2010, pp. 341–345.

[58] A. Arakala, J. Jeffers, and K. Horadam, "Fuzzy extractors for minutiae-based fingerprint authentication," in *Advances in Biometrics: Proceedings of Second International Conference on Biometrics (ICB 2007)*. Springer, 2007, pp. 760–769.

[59] S. Draper, A. Khisti, E. Martinian, A. Vetro, and J. Yedidia, "Using distributed source coding to secure fingerprint biometrics," in *Proceedings of the 32nd International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2007, pp. II–129–II–132.