MITSUBISHI ELECTRIC RESEARCH LABORATORIES
http://www.merl.com

# Secure Storage of Fingerprint Biometrics Using Slepian-Wolf Codes

Stark Draper, Ashish Khisti, Emin Martinian, Anthony Vetro, Jonathan Yedidia

TR2007-006 January 2007

**Abstract**

We describe a method to encode fingerprint biometrics securely for use, e.g., in encryption or access control. The system is secure because the stored data suffices to validate a probe fingerprint but not to recreate the original fingerprint biometric. Therefore, a breach in database security does not lead to the loss of biometric data. We present a model for a secure biometric system for which we can make strong encryption-like security guarantees. We derive a fundamental trade off between system security and the robustness of authentication. The trade off is quantified for simple statistical models. We outline an implementation and report the effectiveness of our method as tested on a data base consisting of 579 datasets, each containing roughly 15 measurements of a single finger.

*Workshop on Information Theory and Applications*

# Secure Storage of Fingerprint Biometrics Using Slepian-Wolf Codes

Stark C. Draper*, Ashish Khisti†, Emin Martinian‡, Anthony Vetro*, and Jonathan S. Yedidia*
*Mitsubishi Electric Research Labs
201 Broadway
Cambridge, MA 02139
{draper, avetro, yedidia}@merl.com
†Massachusetts Institute of Tech.
77 Massachusetts Ave.
Cambridge, MA 02139
khisti@mit.edu
‡Bain Capital
111 Huntington Ave.
Boston, MA 02199
emin@alum.mit.edu

***Abstract*— We describe a method to encode fingerprint biometrics securely for use, e.g., in encryption or access control. The system is secure because the stored data suffices to validate a probe fingerprint but not to recreate the original fingerprint biometric. Therefore, a breach in database security does not lead to the loss of biometric data. We present a model for a secure biometric system for which we can make strong encryption-like security guarantees. We derive a fundamental trade off between system security and the robustness of authentication. The trade off is quantified for simple statistical models. We outline an implementation and report the effectiveness of our method as tested on a data base consisting of 579 datasets, each containing roughly 15 measurements of a single finger.**

## I. Introduction

Securing access to physical locations and to data is of primary concern in many personal, commercial, and governmental contexts. Classic solutions include carrying an identifying document or remembering a password. Problems with the former include forgeries and with the latter include poorly-chosen or forgotten passwords.

Computer-verifiable biometrics provide a third approach. In these systems a sensor measures a biological feature of a person, for example, a fingerprint or an iris scan. It then compares this new sample, termed the probe, with a previously stored sample, termed the enrollment. If the samples match then, depending on the application, the person could be granted access or given a cryptographic key. Advantages of biometrics include the fact that they cannot be forgotten, they can be hard to guess, and they can be difficult to forge.

Biometrics have certain characteristics that pose novel challenges and can create new security holes. A central characteristic that differs biometrics from passwords is that each time a biometric is measured the observation differs. In the case of fingerprints the reading might change because of elastic deformations in the skin when placed on the sensor, dust or oil between finger and sensor, or a cut to the finger. Biometric authentication systems must be robust to such variations.

Most biometric systems deal with such variability by relying on pattern matching. To perform matching the enrollment biometric is stored on the device. This results in a serious security problem. If a malicious attacker gains access to the device, the attacker also gains access to the biometric. In contrast, password-based systems only store hashes. When a user types in a password, the computer compares the hash of the probe password to the stored hash. Access is granted if they match. Since the hash function is effectively impossible to invert, security is not compromised even if an attacker learns the stored hash. Several researchers have attempted to develop "secure" biometric systems with similar characteristics.

Davida, Frankel, and Matt [1] consider the use of error correction coding as a solution to this problem. Juels and Sudan [2] introduce the idea of a fuzzy vault to formalize the use of error correcting codes for such applications. Several researchers have explored cryptographic aspects of the problem in more depth [3], [4], [5]. Some constructions for fingerprints exist, e.g., [6], [7], [8], but yield high false reject rates. A main stumbling block is how to model and exploit the statistical relationship between enrollment biometric and probe. From an information theoretic perspective the secure biometric problem is a problem of "common randomness" [9]. Different parties observe correlated random variables (the enrollment and the probe) and then attempt to agree on a shared secret key (the enrollment biometric). The basic tool used to extract the secret is a distributed source code [10].

Our formulation and proposed solution build on both sets of works. In our implementation we develop a statistical model of the "fingerprint channel" relating the enrollment to the probe and use a graphical code to compress and scramble the enrollment biometric. Iterative decoding using belief-propagation is performed across *both* graphs. This successfully captures both the structure of the code and that of the measurement channel. Our initial work in this area considered iris biometrics [11].

The outline of the remainder of the paper is as follows. In Section II we develop our model of a secured biometric systems and describe the operation of the system. In Section III we quantify a fundamental security-robustness trade off. In Section IV we describe our model of the fingerprint channel, and in Section V evaluate performance on a database of roughly 8100 test fingerprints. More details on implementation and testing can be found in [12].

## II. Secure biometric model

The objective of a (classic, unsecured) biometric system is to provide reliable access control to registered users and to deny access to unregistered users. This is done by comparing a probe biometric with the stored enrollment biometrics. Performance is measured in terms of the false-rejection rate (FRR)

and the false-acceptance rate (FAR). Typically the question of access boils down to a hypothesis test controlled by a threshold.

Secured biometric systems operate under the additional constraint that an enrollment biometric should not be easy to reconstruct from stored data. The more difficult it is to determine any enrollment biometric from the stored data the more secure the system is. At the same time, the stored data must be informative enough that the original enrollment biometric can be recovered (with high probability) when presented with a second measurement of the biometric.

The objective of a secured biometric system is slightly different from that of a classic system. Its underlying objective is not a binary decision. Rather, its objective is to recover the original biometric measured at enrollment. Successful recovery can be validated by storing a cryptographic hash of the original. Only if the hash of the estimated biometric equals the stored hash is recovery successful. The original biometric is a secret shared by encoder and decoder. Certain applications not possible for a classical biometric system are enabled by the existence of a shared secret. We give an example of an encryption application.

The performance of a secured biometric system is quantified using the rates of false-rejection and successful-attack. In unsecured systems the enrollment database is assumed private and the FAR is measured by testing a probe biometric against other users' enrollment biometrics and calculating how often the probe is given access. In the secured system, we define the successful-attack-rate (SAR) under the assumption that an adversary has gained access to the database. Security is measured by how many guesses the adversary must then make to determine any particular user's enrollment biometric. The adversary need not constrain its attack to submitting guesses to the system's decoding rule. It can synthesize any input sequence as the probe and use whatever decoding rule it desires. We will give an example of this added flexibility in Sec. III-B. We define the SAR to be the reciprocal of the number of guesses an attacker needs to make to identify (with high probability) the original biometric.

### A. Enrollment and authentication

We describe our model of the operation of a secured biometric system in terms of an access-control application. During enrollment a user is selected and their raw biometric $\mathbf{b}$ is determined by nature. The biometric is a length-$n$ random vector drawn according to some distribution $P_{\mathbf{b}}(\mathbf{b})$. A joint feature extraction and quantization function $f_{\mathrm{feat}}(\cdot)$ then maps the raw biometric into the enrollment biometric $\mathbf{x} = f_{\mathrm{feat}}(\mathbf{b})$. The user's enrollment biometric $\mathbf{x}$ is the secret shared between the legitimate user and the access control system. Next, a function $f_{\mathrm{sec}}(\cdot)$ maps the enrollment biometric $\mathbf{x}$ into the secure biometric $\mathbf{s} = f_{\mathrm{sec}}(\mathbf{x})$. The access control point stores $\mathbf{s}$, $\mathbf{c}$, and a cryptographic hash of the enrollment $f_{\mathrm{hash}}(\mathbf{x})$. It does not store $\mathbf{b}$ or $\mathbf{x}$.

In the authentication phase, a user requests access and provides a biometric probe $\mathbf{y}$. We model the biometrics of different users as statistically independent. Therefore, if the user is not the legitimate user $P_{\mathbf{y},\mathbf{b}}(\mathbf{y},\mathbf{b}) = P_{\mathbf{b}}(\mathbf{y})P_{\mathbf{b}}(\mathbf{b})$. On the other hand, if $\mathbf{y}$ comes from the legitimate user $P_{\mathbf{y},\mathbf{b}}(\mathbf{y},\mathbf{b}) = P_{\mathbf{b}'|\mathbf{b}}(\mathbf{y}|\mathbf{b})P_{\mathbf{b}}(\mathbf{b})$ where $P_{\mathbf{b}'|\mathbf{b}}(\cdot|\cdot)$ is the biometric channel.

The decoder $g_{\mathrm{dec}}(\cdot,\cdot)$ combines the secure biometric $\mathbf{s}$ with the probe $\mathbf{y}$ and either produces an estimate of the enrollment $\hat{\mathbf{x}}$ or a special symbol $\varnothing$ indicating decoding failure. If $f_{\mathrm{hash}}(\hat{\mathbf{x}})$ matches the stored $f_{\mathrm{hash}}(\mathbf{x})$ access is granted.[1]

### B. Performance measures

The probability of authentication error (false-rejection) is

$$P_{\mathrm{FR}} = \Pr\left[\mathbf{x} \neq g_{\mathrm{dec}}(\mathbf{y}, f_{\mathrm{sec}}(f_{\mathrm{feat}}(\mathbf{b})))\right],$$

where $P_{\mathbf{y},\mathbf{b}}(\mathbf{y},\mathbf{b}) = P_{\mathbf{b}'|\mathbf{b}}(\mathbf{y}|\mathbf{b})P_{\mathbf{b}}(\mathbf{b})$.

It must be assumed that an attacker makes many attempts to guess the desired secret. Therefore, measuring the probability that a single attack succeeds in not particularly meaningful. Instead, security should be assessed by measuring how many attempts an attack algorithm must make to have a reasonable probability of success. As a result, security failure is more complicated to define than authentication failure.

Let $\mathcal{L} = \mathcal{A}_{R_{\mathrm{sec}}}[\cdot]$ be a list of $2^{nR_{\mathrm{sec}}}$ guesses for $\mathbf{x}$ produced by an attack algorithm that uses knowledge of $P_{\mathbf{b}}(\cdot)$, $P_{\mathbf{b}'|\mathbf{b}}(\cdot|\cdot)$, $f_{\mathrm{feat}}(\cdot)$, $f_{\mathrm{sec}}(\cdot)$, $f_{\mathrm{hash}}(\cdot)$, $g_{\mathrm{dec}}(\cdot,\cdot)$, and $\mathbf{s}$. The attacking algorithm does not have access to a probe generated according to $P_{\mathbf{b}'|\mathbf{b}}(\cdot|\cdot)$ because it does not have a measurement of the original biometric. A system is said to be $\epsilon$-secure to rate-$R_{\mathrm{sec}}$ attacks if the probability of successful-attack $P_{\mathrm{SA}}(R_{\mathrm{sec}}) < \epsilon$. This probability equals the probability that the enrollment biometric is on the attacker's list, $P_{\mathrm{SA}}(R_{\mathrm{sec}}) =$

$$\Pr\left[\mathbf{x} \in \mathcal{A}_{R_{\mathrm{sec}}}\left[P_{\mathbf{b}}(\cdot), P_{\mathbf{b}'|\mathbf{b}}(\cdot|\cdot), f_{\mathrm{feat}}(\cdot), f_{\mathrm{sec}}(\cdot), f_{\mathrm{hash}}(\cdot), g_{\mathrm{dec}}(\cdot,\cdot), \mathbf{s}\right]\right].$$

Equivalently, we refer to a scheme with $P_{\mathrm{SA}}(R_{\mathrm{sec}}) = \epsilon$ as having $n \cdot R_{\mathrm{sec}}$ bits of security with confidence $1-\epsilon$. With probability $1-\epsilon$ an attacker must search a key space of $n \cdot R_{\mathrm{sec}}$ bits to crack the system security. In other words the attacker must make $2^{nR_{\mathrm{sec}}}$ guesses. The parameter $R_{\mathrm{sec}}$ is a logarithmic measure of security, quantifying the rate of the increase in security as a function of block length $n$. For instance, 128-bit security requires $nR_{\mathrm{sec}} = 128$.

### C. Goal

Our objective is to construct an encoder and decoder pair that obtains the best combination of robustness (as measured by $P_{\mathrm{FR}}$) and security (as measured by $P_{\mathrm{SA}}(R_{\mathrm{sec}})$) as a function of $R_{\mathrm{sec}}$. In general, improvements in one dimension necessitate a decrease in another. For example, if $P_{\mathrm{SA}}(0.5) = \epsilon$ and $P_{\mathrm{FR}} = 2^{-10}$ at one operating point, increasing the security to $0.75n$ might yield another operating point at $P_{\mathrm{SA}}(0.75) = \epsilon$ and $P_{\mathrm{FR}} = 2^{-8}$.

For authentication failure, the error exponent $-(1/n)\log P_{\mathrm{FR}}$ is the appropriate logarithmic performance measure. For a fixed $\epsilon > 0$, we define the security-robustness

[1] In a data encryption application an encryption key is generated from $\mathbf{x}$ and the matching decryption key from $\hat{\mathbf{x}}$.

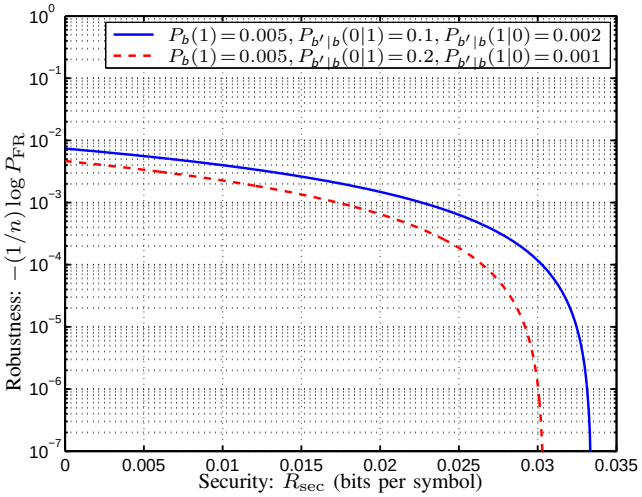


Fig. 1. Example security-robustness regions. The horizontal axis represents the maximum security rate $R_{\rm sec}$ such that $P_{\rm SA}(R_{\rm sec}) < \epsilon$, while the vertical axis represents robustness. The security-robustness region of the system corresponding to the solid curve dominates that of the dashed curve.

region $\mathcal{R}_\epsilon$ as the set of pairs $(r,\gamma)$ where rate-$r$ security is possible with an authentication failure exponent of $\gamma$:

$$\mathcal{R}_\epsilon = \{(R,\gamma)|P_{\rm SA}(R) \le \epsilon, \gamma \ge -(1/n)\log P_{\rm FR}\}\,.$$

The goal is to maximize $\mathcal{R}_\epsilon$. As illustrated in Fig. 1 one secure biometric system dominates another if the security-robustness region of the first is strictly larger than that of the latter.

## III. Quantifying Security

To provide a conceptual framework for our solution we describe an analytic model of a system that is information-theoretically secure. We quantify the trade off between $P_{\rm FR}$ and $P_{\rm SA}(\cdot)$ using information theory and random codes.

### A. Information theoretically secure biometrics

The feature extraction function $f_{\rm feat}(\cdot)$ induces a distribution on $\mathbf{x}$. We assume that $\mathbf{x}$, $\mathbf{b}$ and $\mathbf{y}$ are jointly ergodic and take values in finite sets. In particular, $\mathbf{x} \in \mathcal{X}^n$.

We use a rate-$R_{\rm SW}$ random "binning" function (a Slepian-Wolf code [10]) to encode $\mathbf{x}$ into the secured biometric $\mathbf{s}$. Specifically, we assign each possible sequence $\mathbf{x} \in \mathcal{X}^n$ an integer selected uniformly from $\{1,2,\ldots,2^{nR_{\rm SW}}\}$. The secure biometric is this index $s = f_{\rm sec}(f_{\rm feat}(\mathbf{b}))$. Each possible index $s \in \{1,2,\ldots,2^{nR_{\rm SW}}\}$ indexes a set or "bin" of enrollment biometrics, $\{\mathbf{x}|f_{\rm sec}(\mathbf{x}) = s\}$. The secure biometric can be thought of as a scalar index $s$ or its binary expansion, a uniformly distributed bit sequence $\mathbf{s}$ of length $nR_{\rm SW}$.

During authentication, a user provides a probe biometric $\mathbf{y}$ and claims to be a particular user. The decoder $g_{\rm dec}(\mathbf{y},\mathbf{s})$ searches for a vector $\hat{\mathbf{x}} \in \mathcal{X}^n$ such that is $\hat{\mathbf{x}}$ is jointly typical with $\mathbf{y}$ under the joint distribution $p_{\mathbf{x},\mathbf{b}'}$ and is in bin $\mathbf{s}$, i.e., $f_{\rm sec}(\hat{\mathbf{x}}) = \mathbf{s}$. If a unique $\hat{\mathbf{x}}$ is found, then the decoder outputs this result. Otherwise, an authentication failure is declared and the decoder returns $\varnothing$.

According to the Slepian-Wolf Theorem [10], [13], the decoder will succeed with probability approaching 1 as $n$ increases provided that $R_{\rm SW} > (1/n)H(\mathbf{x}|\mathbf{b}')$. Thus, $P_{\rm FR}$ approaches zero for long block lengths. The theory of error exponents for Slepian-Wolf coding [14] tells us that $-(1/n)\log P_{\rm FR} \ge E_{\rm SW}(R_{\rm SW})$ where $E_{\rm SW}(R_{\rm SW})$ is defined as

$$\max_{0\le\rho\le 1} \rho R_{\rm SW} - \frac{1}{n}\log \sum_{\mathbf{b}'} p_{\mathbf{b}'}(\mathbf{b}') \left[\sum_{\mathbf{x}} p_{\mathbf{x}|\mathbf{b}'}(\mathbf{x}|\mathbf{b}')^{\frac{1}{1+\rho}}\right]^{1+\rho}. \quad (1)$$

If the source is memoryless, the second term of (1) simplifies to $-\log\sum_{b'} p_{b'}(b')[\sum_x p_{x|b'}(x|b')^{\frac{1}{1+\rho}}]^{1+\rho}$.

Next we consider the probability of successful attack, i.e., how well an attacker can estimate $\mathbf{x}$ given the secure biometric $\mathbf{s}$. According to the asymptotic equipartition property [15], under the fairly mild technical condition of ergodicity it can be shown that conditioned on $\mathbf{s} = f_{\rm sec}(\mathbf{x})$, $\mathbf{x}$ is approximately uniformly distributed over the typical set of size $2^{H(\mathbf{x}|\mathbf{s})}$. Therefore, with high probability, it will take approximately this many guesses to identify $\mathbf{x}$. We compute $H(\mathbf{x}|\mathbf{s})$ as

$$H(\mathbf{x},\mathbf{s}) - H(\mathbf{s}) = H(\mathbf{x}) - H(\mathbf{s}) = H(\mathbf{x}) - nR_{\rm SW}. \quad (2)$$

Note that in the classic attack used to calculate the FAR, $\mathbf{y}$ is chosen from $p_{\mathbf{b}'}(\cdot)$ independently of $\mathbf{x}$. This attack fails unless the $\mathbf{y}$ chosen is jointly typical with $\mathbf{x}$. This takes approximately $2^{H(\mathbf{b}')-H(\mathbf{b}'|\mathbf{x})} = 2^{H(\mathbf{x})-H(\mathbf{x}|\mathbf{b}')}$ guesses. Since $R_{\rm SW} > (1/n)H(\mathbf{x}|\mathbf{b}')$ an FAR-type attack will almost always take many more guesses than an attack that makes its guesses conditioned on $\mathbf{s}$. We use (1) and (2) to bound the security-robustness region.

*Proposition 1:* For any $\epsilon > 0$ as $n \to \infty$ an inner bound to the security-robustness region $\mathcal{R}_\epsilon$ is found by taking a union over all possible feature extraction functions $f_{\rm feat}(\cdot)$ and secure biometric encoding rates $R_{\rm SW}$

$$\mathcal{R}_\epsilon \supset \bigcup_{f_{\rm feat}(\cdot),R_{\rm SW}} \left\{ r,\gamma \middle| r \le \frac{1}{n}H(\mathbf{x}) - R_{\rm SW}, \gamma \le E_{\rm SW}(R_{\rm SW}) \right\}$$

where $E_{\rm SW}(R_{\rm SW})$ is given by (1) for the $p_{\mathbf{x},\mathbf{b}'}(\cdot,\cdot)$ induced by the chosen $f_{\rm feat}(\cdot)$.

Figure 1 plots an example of the security-robustness region for a memoryless insertion and deletion channel somewhat akin to the fingerprint channel. The biometric $\mathbf{b}$ is an independent identically distributed (i.i.d.) Bernoulli sequence with $p_b(1) = 0.05$. The true biometric is observed through the asymmetric binary channel with deletion probability $p_{b'|b}(0|1)$ and insertion probability $p_{b'|b}(1|0)$. We examine the case where $f_{\rm feat}(\cdot)$ is the identity function, i.e., $\mathbf{x} = \mathbf{b}$, and plot the resulting security-robustness regions for two choices of $p_{b'|b}(\cdot|\cdot)$ and $p_{b'|b}(\cdot|\cdot)$. The choice $\mathbf{x} = \mathbf{b}$ allows for the maximum security region as it gives the largest $H(\mathbf{x})$.

It is important to emphasize that an attack that identifies a biometric $\tilde{\mathbf{x}}$ such that $f_{\rm sec}(\tilde{\mathbf{x}}) = \mathbf{s}$ is not necessarily a successful attack. In fact, it can be quite easy to find a $\tilde{\mathbf{x}}$ that satisfies $f_{\rm sec}(\tilde{\mathbf{x}}) = \mathbf{s}$. However, if $\tilde{\mathbf{x}} \ne \mathbf{x}$. then $f_{\rm hash}(\tilde{\mathbf{x}}) \ne f_{\rm hash}(\mathbf{x})$ and access will not be granted. Indeed, in the bounds

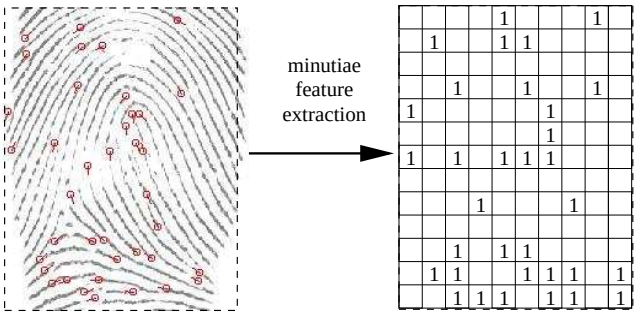


Fig. 2. Fingerprint and extracted feature vector.

on security provided by Prop. 1 the attacker is assumed to limit itself to guessing sequence $\tilde{\mathbf{x}}$ that do satisfy $f_{\text{sec}}(\tilde{\mathbf{x}}) = \mathbf{s}$.

### B. Syndrome decoding and the zero-probe attack

When $nR_{\text{SW}} > H(\mathbf{x})$ the system is not information-theoretically secure. However, recovering $\mathbf{x}$ from $\mathbf{s}$ can still be very difficult. The recovery of $\mathbf{x}$ from $\mathbf{s}$ is a syndrome decoding problem with $\mathbf{x}$ playing the role of the error sequence. In this context syndrome decoding requires storage of a look-up table of size $2^{nR_{\text{SW}}}$. In the fingerprint problem $n = 7000$ and $\mathbf{x}$ is modeled as a Bernoulli-$0.0046$ i.i.d. source. This means that the table size $2^{nR_{\text{SW}}} > 2^{294}$, so syndrome decoding is intractable. However, as $nR_{\text{SW}}$ gets much larger than $H(\mathbf{x})$ other approaches can tractably recover $\mathbf{x}$.

We introduce the "zero-probe" attack to test this security. The attacker know $\mathbf{s}$, it knows the code structure, and it can use any attack it likes. In the zero-probe attack it guesses the all-0 probe $\mathbf{y} = 0$ and uses BP to try to solve the syndrome decoding problem. If $R_{\text{SW}}$ is large enough this BP-based attack will recover $\mathbf{x}$. However, when $nR_{\text{SW}}$ is close to $H(\mathbf{x})$ this attack fails. We report the efficacy of this attack, as well as that of the standard biometric attack of using some other fingerprint in conjunction with $\mathbf{s}$ to decode. The success rate of the latter attack is given by the false-acceptance rate (FAR).

## IV. Fingerprint feature set and statistical modeling

A popular method for working with fingerprint data is to extract a set of "minutiae points" and to perform all subsequent operations on them. Figure 2 gives an example of a fingerprint, the minutiae points, and the extracted feature vector that we work with. Each minutiae is a discontinuity in the ridge map of a fingerprint, indicated by the circles in the left-hand plot. The quantized coordinates of a particular minutia location is indicated by a '1' in the right-hand plot.

We create a statistical model for the fingerprint channel which captures three effects: (1) movement of enrollment minutiae when subsequently observed in the probe, (2) deletions–minutiae observed at enrollment, but not in the probe, and (3) insertions–"spurious" minutiae observed in the probe but not during enrollment.

Figure 3 depicts the factor graph [16] model we develop. The presence of a minutiae point at position $t$ in the enrollment grid is represented by the binary random variable $x_t$ that takes on the value $x_t = 1$ only if a minutiae is present during enrollment. For simplicity, the figure shows a one-dimensional movement model. The results reported in this paper all use a two-dimensional movement model. We model the enrollment biometric $\mathbf{x}$ as a Bernoulli-$p_{\text{p}}$ i.i.d. random vector. These prior probabilities are denoted by the white-square factor nodes (□).

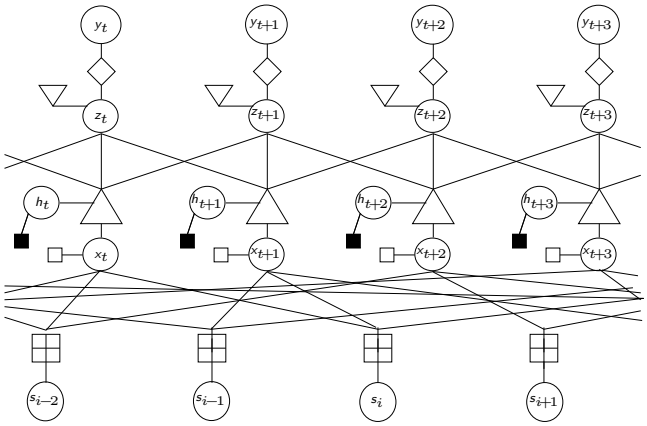


Fig. 3. Factor graph of minutiae movement model.

For each position in the enrollment grid there is a corresponding position in the probe grid. The presence of a minutiae point at grid position $t$ in the probe is represented by the binary random variable $y_t$ taking on value $y_t = 1$.

Some minutiae observed during enrollment are not observed in the probe. The binary random variable $h_t$ represents one such erasure. It takes on value $h_t = 1$ if $x_t$ is erased. The black-square factor nodes (■) represent the prior probability on $h_t$. We model $\mathbf{h}$ as an i.i.d. Bernoulli-$p_{\text{e}}$ sequence.

Our model captures the local elastic deformations in the skin that occur when a finger is placed on a sensor.[2] For each enrollment position $t$ the model specifies a neighborhood $\mathcal{N}(t)$ of positions to which the enrollment minutiae can move. The $z_t$ variables in Fig. 3 capture the relative change in position of enrollment minutiae, and $\mathbf{z}_{\mathcal{N}(t)} = \{z_i | i \in \mathcal{N}(t)\}$ are the set of these variables in the neighborhood of enrollment position $t$. The upside-down triangle factor nodes (▽) represent the prior probability distribution both on minutiae movement and the event that a spurious minutiae is generated at this position. If a minutia moves beyond its neighborhood, the model treats it as a deletion and an insertion.

The variables $z_t$ take values in the set $z_t \in \{\text{Ⓢ}, *, \Delta\mathcal{N}(t)\}$. If $z_t = \text{Ⓢ}$ a spurious minutiae unrelated to the enrollment was generated at position $t$ in the probe. If $z_s = *$ there is no minutiae at position $t$ in the probe (i.e., $y_t = 0$). The diamond factor nodes (◊) connecting each $y_t$ to its corresponding $z_t$ capture the notion that each probe minutiae $y_t$ can only be non-zero if there is a corresponding $z_t \neq *$. Finally, $\Delta\mathcal{N}(t)$ is the set of relative shifts that define the possible movements and

[2]We assume that global translations and rotations of a fingerprint are corrected through a combination of pre-processing and a search over small (rigid) shifts.

hence the neighborhood $\mathcal{N}(t)$. For example, in the simple one-dimensional movement model of Fig. 3, $\Delta\mathcal{N}(t) = \{-1, 0, 1\}$.

Both the support of minutiae movement (the choice of the $\Delta\mathcal{N}(t)$) and the prior on the movement (the distribution on $z_t$) are design choices. While a larger neighborhood helps to capture the tails of minutiae movement, it also incurs greater computational complexity and adds loops to the graphical model. These extra loops can ultimately pose problems for the graph-based inference algorithm we used to decode; we use belief propagation (BP).

Each enrollment minutia $x_t$ is constrained to move only within its neighborhood $\mathcal{N}(t)$. Furthermore, it can move to only one point, and therefore can explain only a single minutiae point observed in the probe. The triangular factor nodes ($\triangle$) in Fig. 3 capture these movement constraints.

The complete model of the biometric source and channel is $p_{\mathbf{x},\mathbf{y}}(\mathbf{x},\mathbf{y}) = p_{\mathbf{x}}(\mathbf{x})p_{\mathbf{y},\mathbf{x}}(\mathbf{y},\mathbf{x}) =$

$$\sum_{\{h_i\}}\sum_{\{z_i\}}\prod_t \square(x_t)\blacksquare(h_t)\nabla(z_t)\triangle(x_t,h_t,\mathbf{z}_{\mathcal{N}(t)})\Diamond(z_t,y_t).$$

To the biometric model we add the code constraints. The local code constraints $\boxplus(s_j,\mathbf{x})$ are indicator functions equaling one if the value of each $\mathbf{s}_j$ is compatible with $\mathbf{x}$ and zero otherwise. The complete model is $p_{\mathbf{x},\mathbf{y},\mathbf{s}}(\mathbf{x},\mathbf{y},\mathbf{s}) = p_{\mathbf{x},\mathbf{y}}(\mathbf{x},\mathbf{y})\prod_j \boxplus(s_j,\mathbf{x})$. In this paper the $\mathbf{s}_j$ are the mod-2 sum of the enrollment minutiae to which $\mathbf{s}_j$ is connected by the code graph. The connections defining the code graph are generated according to a low-density parity-check (LDPC) code. The graphical nature of LDPCs makes it easy to merge their description into that of the biometric channel, as is shown in Fig. 3.

Given the graphical model for $p_{\mathbf{x},\mathbf{y},\mathbf{s}}$, the raw message passing rules for use in belief propagation can be derived using standard techniques [16]. In order to make the computations tractable we introduce a number of computational optimizations. These optimizations exploit the particular structure of the messages, the graph, and the quantities being computed. Due to space constraints, we do not further discuss these optimizations here

## V. Experimental Results

We evaluate our approach on a Mitsubishi Electric (MELCO) fingerprint database. The database consists of 579 data sets, each containing roughly 15 measurements of a single finger. The measurement field is $70 \times 100$ pixels, and the average number of minutiae is about 32. We select one measurements from each data set as the enrollment and attempt to decode using the remaining measurements as probes. All syndrome calculations use a rate $0.94$ LDPC code.[3] In the movement model we allow minutiae to move up to 3 pixels in the vertical or horizontal directions, resulting in a neighborhood size of 49. The zero-probe attack fails to decode any of the enrollment prints. Test parameters and FRRs and FARs are given in Table. I. While the failure of the zero-probe attack is one indication that we have some computational security, examination of the test parameter reveals that our codes are not yet strong enough to get into the information theoretically secure region. This is the focus of current work. A fuller description of the experiments can be found in [12].

[3]Note that the relationship between the rate of the channel code $R_{\mathrm{LDPC}}$ and the Slepian-Wolf coding rate $R_{\mathrm{SW}}$ is $R_{\mathrm{LDPC}} = 1 - R_{\mathrm{SW}}$.

| # enrollment minutiae (ent) | Num. files | SAR 0-probe | FRR | | FAR | |
|---|---|---|---|---|---|---|
| | | | rate | probes | rate | probes |
| 31 (0.0410) | 195 | 0 | 11.6e-2 | 2736 | 0.98e-2 | 11e4 |
| 32 (0.0421) | 139 | 0 | 13.3e-2 | 1944 | 0.33e-2 | 7.8e4 |
| 33 (0.0432) | 107 | 0 | 14.9e-2 | 1506 | 0.24e-2 | 6.0e4 |
| 34 (0.0443) | 79 | 0 | 20,2e-2 | 1101 | 0.11e-2 | 4.4e4 |
| 35 (0.0454) | 59 | 0 | 32.3e-2 | 824 | 0.03e-2 | 3.3e4 |

TABLE I

Test parameters, zero-probe SAR, FRR, and FAR results.

## VI. Conclusions

We present a secure biometrics systems for fingerprints. The design is based on a statistical model of minutia movement and graphical codes. Our current focus is on the refined design of LDPC codes, better matched to the asymmetric (and not memoryless) nature of the fingerprint channel.

## References


[1] G. I. Davida, Y. Frankel, and B. J. Matt, "On enabling secure applications through off-line biometric identification," in *IEEE Symp. Secur. Priv.*, Oakland, CA, May 1998, pp. 148–157.

[2] A. Juels and M. Sudan, "A fuzzy vault scheme," in *IEEE Int. Symp. Inform. Theory*, Lausanne, Switzerland, Jul 2002, p. 408.

[3] Y. Dodis, L. Reyzin, and A. Smith, "Fuzzy extractors: How to generate strong keys from biometrics and other noisy data," in *EUROCRYPT*, Interlaken, Switzerland, 2004, pp. 523–540.

[4] X. Boyen, Y. Dodis, J. Katz, R. Ostrovsky, and A. Smith, "Secure remote authentication using biometric data," in *EUROCRYPT*, Aarhus, Denmark, 2005, pp. 147–163.

[5] A. Sahai and B. Waters, "Fuzzy identity-based encryption," in *EUROCRYPT*, Aarhus, Denmark, 2005, pp. 457–473.

[6] T. C. Clancy, N. Kiyavash, and D. J. Lin, "Secure smartcardbased fingerprint authentication," in *ACM SIGMM Work. Biom. Meth. Apps.*, Berkeley, CA, 2003, pp. 45–52.

[7] S. Yang and I. M. Verbauwhede, "Secure fuzzy vault based fingerprint verification system," in *Asilomar Conf.*, Monterey, CA, Nov 2004, pp. 577–581.

[8] U. Uludag and A. Jain, "Fuzzy fingerprint vault," in *Workhop: Biometrics: Challenges Arising from Theory to Practice*, Cambridge, UK, Aug 2004, pp. 13–16.

[9] R. Ahlswede and I. Csiszar, "Common randomness in information theory and cryptography I: Secret sharing," *IEEE Trans. Inform. Theory*, pp. 1121–1132, Jul 1993.

[10] D. Slepian and J. K. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inform. Theory*, pp. 471–480, Jul 1973.

[11] E. Martinian, S. Yekhanin, and J. S. Yedidia, "Secure biometrics via syndromes," in *Allerton Conf.*, Monticello, IL, Sep 2005.

[12] S. C. Draper, A. Khisti, E. Martinian, A. Vetro, and J. S. Yedidia, "Using distributed source coding to secure fingerprint biometrics," in *Int. Conf. Acoutics Speech Signal Proc.*, Honolulu, HI, 2007.

[13] T. M. Cover, "A proof of the data compression theorem of Slepian and Wolf for ergodic sources," no. 2, pp. 226–228, Mar 1975.

[14] R. G. Gallager, "Source coding with side information and universal coding," Massachusetts Institute of Tech., Tech. Rep. LIDS P-937, 1976.

[15] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: John Wiley & Sons, Inc., 1991.

[16] F. R. Kschischang, B. J. Frey, and H. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Trans. Inform. Theory*, pp. 498–519, Feb 2001.