Embedding Renewable Cryptographic Keys into Continuous Noisy Data

Ileana Buhan, Jeroen Doumen, Pieter Hartel, Qiang Tang, Raymond Veldhuis

Faculty of EWI, University of Twente, the Netherlands

Abstract. Fuzzy extractor is a powerful but theoretical tool to extract uniform strings from discrete noisy data. Before it can be used in practice, many concerns need to be addressed in advance, such as making the extracted strings renewable and dealing with continuous noisy data. We propose a primitive *fuzzy embedder* as a practical replacement for fuzzy extractor. Fuzzy embedder naturally supports renewability because it allows a randomly chosen string to be embedded. Fuzzy embedder takes continuous noisy data as input and its performance directly links to the property of the input data. We give a general construction for fuzzy embedder based on the technique of Quantization Index Modulation (QIM) and derive the performance result in relation to that of the underlying QIM. In addition, we show that quantization in 2-dimensional space is optimal from the perspective of the length of the embedded string. We also present a concrete construction for fuzzy embedder in 2-dimensional space and compare its performance with that obtained by the 4-square tiling method of Linnartz, *et al.* [13].

1 Introduction

Most cryptographic protocols rely on exactly reproducible key material. In fact, these protocols are designed to have a wildly different output if the key is perturbed slightly. Unfortunately, exactly reproducible keys are hard to come by, especially when they also need to have sufficient entropy. Luckily, it is relatively easy to find “fuzzy” sources, such as physically uncloneable functions (PUFs) [17] and biometrics [8]. However, such sources are inherently noisy and rarely uniformly distributed. The first (main) difficulty in transforming a fuzzy source into key material is to correct the noise and reproduce the same key every time. To solve this problem, the notion of secure sketch [12] has been proposed. The second difficulty lies in the fact the output of secure sketch may have a non-uniform distribution, while it should be as close to uniform as possible to serve as a cryptographic key. A strong randomness extractor could be used to turn the reproducible output into a nearly uniform string. In the literature, a common way of extracting keys from noisy data is to combine a secure sketch with a strong randomness extractor, which leads to the notion of a fuzzy extractor [8].

When deploying a fuzzy extractor in practice, more concerns need to be addressed. Firstly, even with the same input (noisy data), it should be possible to extract different keys (referred to as renewability). To achieve renewability, the (fixed) output of the fuzzy extractor must be randomized, for instance by using a common reference string. Unfortunately, this falls outside the scope of fuzzy extractor, even though it is recognized as an important and sensitive issue [2]. Secondly, fuzzy extractor only accepts discrete sources as input. Existing performance measures for secure sketches, such as entropy loss or min-entropy, lose their relevance when applied to continuous sources [12]. This limitation can be overcome by quantizing the continuous input. Li, *et al.* [12] propose to define relevant performance measures for secure sketch with respect to the chosen quantization method.

CONTRIBUTIONS. Our contribution is threefold. Firstly, we propose a new primitive *fuzzy embedder* which can be regarded as a practical replacement for fuzzy extractor. Fuzzy embedder can embed a uniformly distributed key while taking continuous noisy data as input. Its performance directly links to the property of the input data. Fuzzy embedder formalizes the concept of "key binding" in biometric template protection schemes surveyed by Uludag, *et al.* [20]. In fact, fuzzy embedder can also be regarded as a natural extension of fuzzy extractor, since it can embed a fixed string (for instance one obtained by applying a strong extractor to the input source) into a discrete source and thus achieve the same functionality, namely a randomized cryptographic key. However, a fuzzy embedder scheme can be directly used with any type of input to achieve the same goal as a fuzzy extractor scheme without the need to address those concerns mentioned previously.

Secondly, we propose a general construction for fuzzy embedder based on the technique of Quantization Index Modulation ($\mathtt{QIM}$) and derive the performance result in relation to that of the underlying $\mathtt{QIM}$. In the context of watermarking, using $\mathtt{QIM}$ can achieve efficient trade-offs between the information embedding rate, the reliability and the distortion [5]. The trade-offs of the underlying $\mathtt{QIM}$ give rise to similar trade-offs in fuzzy embedder performance measures. Note that shielding functions [13] can be regarded as a particular construction of a fuzzy embedder, as they focus on one particular type of quantizer. However, they only consider one-dimensional inputs.

Thirdly, we investigate different quantization strategies for high dimensional data and show that quantization in two dimensions gives an optimal length of the embedded uniform string. Finally, we propose a concrete construction of fuzzy embedder in 2-dimensional space and compare its performance with that obtained by the 4-square tiling method of Linnartz, *et al.* [13].

RELATED WORK. Dodis, *et al.* [8] consider discrete distributed noise and propose fuzzy extractors and secure sketches for different error models. These models are not directly applicable to continuously distributed sources. Linnartz, *et al.* [13] construct shielding functions for continuously distributed data and propose a practical construction which can be considered a 1-dimensional $\mathtt{QIM}$. The same approach is taken by Li, *et al.* [12] who propose quantization functions for extending the scope of secure sketches to continuously distributed data. Buhan, *et al.* [3] analyze the achievable performance of such constructions given the quality of the source in terms of the false acceptance rate and false rejection rate of a biometric system.

The process of transforming a continuous distribution to a discrete distribution influences the performances of secure sketches and fuzzy extractors. Quantization is the process of replacing analogue samples with approximate values taken from a finite set of allowed values. The basic theory of one-dimensional quantization is reviewed by

Gersho [9]. The same author investigates the influence of high dimensional quantization on the performance of digital coding for analogue sources [10]. `QIM` constructions are used by Chen and Wornell [5] in the context of watermarking. The same authors introduce dithered quantizers [6]. Moulin and Koetter [16] give an excellent overview of `QIM` in the general context of data hiding. Barron, *et al.* [1] develop a geometric interpretation of conflicting requirements between information embedding and source coding with side information.

Fuzzy embedder is somehow related to the concept of information theoretic key agreement [14,15]. However, the settings of the problem are different. In secure message transmission based on correlated randomness the attacker and the legitimate participants have a noisy share of the same source data, while, in the fuzzy embedder setting, the attacker does not have access to the data source.

ROADMAP. The rest of the paper is organized as follows. In *Section* 2 we describe our notation and provide some background knowledge. In *Section* 3 we present the definition of fuzzy embedder and highlight the differences with fuzzy extractor. In *Section* 4 we propose a general construction of a fuzzy embedder from any `QIM` and express the performance in terms of the geometric properties of the underlying quantizers. In *Section* 5 we present a concrete construction for fuzzy embedder in 2-dimensional space and compare its performance with that obtained by the 4-square tiling method of Linnartz, *et al.*. In the last section we conclude this paper.

2 Preliminaries

Let $\mathcal{M}$ be an n-dimensional discrete, finite set, which together with a distance function $d_{\mathcal{M}} : \mathcal{M} \times \mathcal{M} \rightarrow \mathbb{R}^+$ forms a metric space. Similarly, let $\mathcal{U}$ be an n-dimensional continuous domain, which together with the distance $d_{\mathcal{U}} : \mathcal{U} \times \mathcal{U} \rightarrow \mathbb{R}^+$ forms a metric space. For the purpose of this work, we use d for both $d_{\mathcal{M}}$ and $d_{\mathcal{U}}$. Capital letters are used to denote random variables while small letters are used to denote realizations of random variables. Continuous random variables are defined over the metric space $\mathcal{U}$ while discrete random variables are defined over the metric space $\mathcal{M}$. A random variable A is endowed with a probability density function $f_A(a)$. We use the random variable P when referring to public sketch data and R for random binary strings in the descriptions of fuzzy extractor and fuzzy embedder.

MUTUAL INFORMATION. By $I(A; B)$ we note the Shannon mutual information between the two random variables A and B, which measures the amount of uncertainty left about A when B is made public. We have $I(A; B) = 0$ if and only if A and B are independent random variables. Formal definitions of entropy, min-entropy, average min-entropy, and statistical distance SD can be found in [8].

FUZZY EXTRACTOR. According to the definition by Dodis, *et al.* [8], a fuzzy extractor extracts a uniformly random string r from a value x of random variable X in a noise-tolerant way with the help of some public sketch p (see, *Figure* 1). For a discrete metric space $\mathcal{M}$ with a distance measure d, fuzzy extractor [2,8] is formally defined as follows.

Definition 1 (Fuzzy Extractor). *An $(\mathcal{M}, m, l, t, \epsilon)$ fuzzy extractor is a pair of randomized procedures $\langle \mathsf{Generate}, \mathsf{Reproduce} \rangle$ with the following properties:*

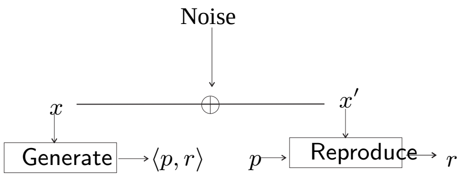

Fig. 1. *A fuzzy extractor is a pair of two procedures* $\langle$Generate, Reproduce$\rangle$. *The* Generate *function takes noisy data* x *as input and returns a random string* r *and a public sketch* p. *The* Reproduce *function takes noisy data* x' *and the public sketch* p *as input, and outputs* r *if* x *and* x' *are close.*

1. *The generation procedure on input of* $x \in \mathcal{M}$ *outputs an extracted string* $r \in R = \{0,1\}^l$ *and a public helper string* $p \in P = \{0,1\}^*$.
2. *The reproduction procedure takes an element* $x' \in \mathcal{M}$ *and the public string* $p \in \{0,1\}^*$ *as input. The* reliability *property of the fuzzy extractor guarantees that if* $d(x,x') \leq t$ *and* r, p *were generated by* $(r,p) \leftarrow \mathsf{Generate}(x)$, *then* $\mathsf{Reproduce}(x',p) = r$. *If* $d(x,x') > t$, *then no guarantee is provided about the output of the reproduction procedure.*
3. *The* security *property guarantees that for any random variable* X *with distribution* $f_X(x)$ *of min-entropy* m, *the string* r *is nearly uniform even for those who observe* p: *if* $(r,p) \leftarrow \mathsf{Generate}(X)$, *then* $SD((R,P),(N,P)) \leq \epsilon$ *where N is a random variable with uniform probability.*

In other words, a fuzzy extractor allows to generate the random string r from a value x. The reproduction procedure which uses the public string p produced by the generation procedure will output the string r as long as the measurement x' is close enough. This is the *reliability* property of the fuzzy extractor. The *security* property guarantees that r looks uniformly random to an attacker and her chance to guess its value from the first trial is approximately 2^{-m}. Security encompasses both *min-entropy* and uniformity of the random string r when p are known to an attacker.

We have two observations on the shortcomings of fuzzy extractor. One is that, the public string is from the discrete set $P = \{0,1\}^*$. However, there are biometric template protection schemes that fit the model of the fuzzy extractors for which P is drawn from $\mathbb{R}$ [13] or $\mathbb{Z}$ [18]. The other is that, defining min-entropy for X makes sense only if X has a discrete probability density function otherwise its min-entropy depends on the quantization of the variable [12].

QUANTIZATION. A continuous random variable A can be transformed into a discrete random variable by means of quantization, which we write as $Q(A)$. Formally, a quantizer is a function $Q : \mathcal{U} \rightarrow \mathcal{M}$ that maps $a \in \mathcal{U}$ into the closest *reconstruction point* in the set $\mathcal{M} = \{c_1, c_2, \cdots\}$ by

$$Q(a) = \mathsf{argmin}_{c_i \in \mathcal{M}} d(a, c_i)$$

where d is the distance measure defined on $\mathcal{U}$. The *Voronoi region* or the *decision region* of a reconstruction point c_i is the subset of all points in $\mathcal{U}$, which are closer to that particular reconstruction point than to any other reconstruction point. We denote

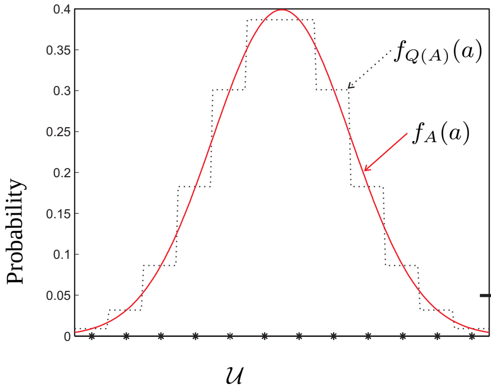

Fig. 2. *By quantization, $f_A(a)$ (continuous line) is transformed into $f_{Q(A)}(a)$ (dotted line).*

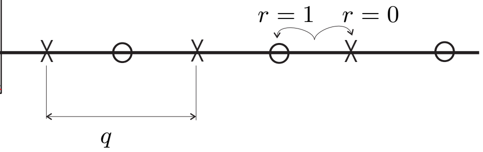

Fig. 3. *Quantization of X with two scalar quantizers Q_0 and Q_1 both with step size q.*

with V_{c_i} the Voronoi region of reconstruction point c_i. When A is 1-dimensional, Q is called a *scalar* quantizer. If all Voronoi regions of a quantizer are equal, the quantizer is *uniform*. In the scalar case, the length of the Voronoi region is then called the *step size.* If the reconstruction points form a lattice, the Voronoi regions of all reconstruction points are congruent. By quantization, the probability density function of the continuous random variable A, $f_A(a)$ which is continuous, is transformed into the probability density function $f_{Q(A)}(a)$ which is discrete (See *Figure* 2).

QUANTIZATION-BASED DATA HIDING CODES. Quantization based data hiding codes, introduced by Chen, *et al.* [5] (also known as `QIM`), can embed secret information into a real value. We start with the following example.

Example 1. We want to embed one bit of information, thus $r \in \{0, 1\}$ into a real value x. For this purpose we use a scalar uniform quantizer with step size q, given by

$$Q(x) = q \left[\frac{x}{q} \right].$$

The quantizer Q is used to generate a set of two new quantizers $\{Q_0, Q_1\}$ defined as:

$$v_0 = \frac{q}{4}, \;\; v_1 = -\frac{q}{4}, \;\; Q_0(x) = Q(x + v_0) - v_0, \;\; Q_1(x) = Q(x + v_1) - v_1.$$

In *Figure* 3 the reconstruction points for the quantizer Q_1 are shown as circles and the reconstruction points for the quantizer Q_0 are shown as crosses. The embedding is done by mapping the point x to the elements of these two quantizers. For example, if $r = 1$, x is mapped to the closest $\circ$ point. The result of the embedding is the distance vector to the nearest $\times$ or $\circ$ as chosen by r. During reproduction procedure, when x is perturbed by noise, the quantizer will assign the received data to the closest $\times$ or $\circ$ point, and output 0 or 1 respectively.

Formally, a *Quantization Index Modulation* data hiding scheme, can be seen as `QIM` $: \mathcal{U} \times R \to \mathcal{M}$ a set of individual quantizers $\{Q_1, Q_2, \ldots Q_{2^l}\}$, where $l = |R|$ and each quantizer maps $x \in \mathcal{U}$ into a reconstruction point. The quantizer is chosen by the input value $r \in R$ such that $\mathtt{QIM}(x, r) = Q_r(x)$. The set of all reconstruction points is $\mathcal{M} = \bigcup_{r \in R} \mathcal{M}_r$ where $\mathcal{M}_r \subset \mathcal{M}$ is the set of reconstruction points of the quantizer Q_r.

We define the *minimum distance* $\sigma_{\min}$ of a `QIM`, as the minimum distance between reconstructions points of all quantizers in the `QIM`:

$$\sigma_{\min} = \min_{r_1, r_2 \in R} \min_{c^i_{r_1} \in \mathcal{M}_{r_1}, c^j_{r_2} \in \mathcal{M}_{r_2}} d(c^i_{r_1}, c^j_{r_2})$$

where $\mathcal{M}_{r_1} = \{c^1_{r_1}, c^2_{r_1}, \cdots\}$ and $\mathcal{M}_{r_2} = \{c^1_{r_2}, c^2_{r_2}, \cdots\}$. Hence, balls with radius $\frac{\sigma_{\min}}{2}$ and centers in $\mathcal{M}$ are disjoint. Let ζ_r be the smallest radius ball such that balls centered in the reconstruction point of quantizer Q_r with radius ζ_r cover the universe $\mathcal{U}$. We define the *covering distance* $\lambda_{\max}$ as:

$$\lambda_{\max} = \max_{r \in \mathcal{R}} \zeta_r.$$

Any ball $B(c, \zeta_r)$ contains at least one ball $B(c_r, \sigma_{\min}/2)$ for $c_r \in \mathcal{M}_r, \forall r \in R$. Hence, balls with radius $\lambda_{\max}$ and centers in $\mathcal{M}_r$ cover the universe $\mathcal{U}$.

A *dithered* `QIM` [6] is a special type of `QIM` for which all Voronoi region of all individual quantizers are congruent polytopes (generalization of a polygon to higher dimensions). Each quantizer in the ensemble $\{Q_1, Q_2, \ldots Q_{2^l}\}$ can be obtained by shifting the reconstruction points of any other quantizer in the ensemble. The shifts correspond to dither vectors $\{v_1, v_2, \ldots v_{2^l}\}$. The number of dither vectors is equal to the number of quantizers in the ensemble.

The reliability (or, the amount of tolerated noise) of a `QIM` is determined by the minimum distance between two neighboring reconstruction points. The size and shape (for high dimensional quantization) of the Voronoi region determines the tolerance for error. The number of quantizers in the `QIM` set determines the amount of information that can be embedded. By setting the number of quantizers and by choosing the shape and size of the decision region the performance properties can be fine tuned.

3 Fuzzy Embedder

In this section, we define fuzzy embedder and show its relationship with fuzzy extractor. It is worth stressing that the random key r is not extracted from the random x, but is generated independently, as illustrated in *Figure* 4.

Definition 2 (Fuzzy Embedder). *A $(\mathcal{U}, \ell, \rho, \epsilon, \delta)$-fuzzy embedder scheme consists of two polynomial-time algorithms* $\langle$Embed, Reproduce$\rangle$*, which are defined as follows:*

- Embed*: $\mathcal{U} \times R \to P$, where $R = \{0, 1\}^l$. This algorithm takes $x \in \mathcal{U}$ and $r \in R$ as input, and returns a public sketch $p \in P$.*
- Reproduce*: $\mathcal{U} \times P \to R$. This algorithm takes $x' \in \mathcal{U}$ and $p \in P$ as input, and returns a string from R or an error symbol $\perp$.*

Given any random variable X over $\mathcal{U}$ and a random variable R, the parameter ρ, ϵ, δ are defined as follows:

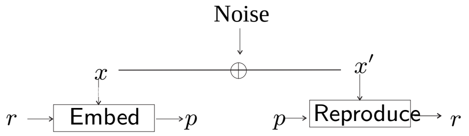

Fig. 4. *A fuzzy embedder is a pair of two procedures* $\langle\mathsf{Embed}, \mathsf{Reproduce}\rangle$. *The* Emded *function takes noisy data* x *and a binary string* r *as input, and outputs a public sketch* p. *The* Reproduce *function takes noisy data* x' *and the public sketch* p *as input, and outputs* r *if* x *and* x' *are close.*

- *The parameter* ρ *represents the probability that the fuzzy embedder can successfully reproduce the embedded key, and it is defined as*

$$\rho = \min_{r \in R} \max_{x \in \mathcal{U}} \Pr(\mathsf{Reproduce}(x', \mathsf{Embed}(x, r)) = r | x' \in X).$$

  In the above definition, the maximum over $x \in \mathcal{U}$ *ensures that we choose the best possible representative* x *for the random variable* X. *In most cases, this will be the mean of* X.
- *The security parameter* ϵ *is equal to the mutual information between the embedded key and the public sketch, and it is defined as* $\epsilon = I(R; \mathsf{Embed}(X, R))$.
- *The security parameter* δ *is equal to the mutual information of the noisy data and the public sketch and is defined as* $\delta = I(X; \mathsf{Embed}(X, R))$.

Since the public sketch p is computed both on X and R, ϵ measures the amount of information revealed about X and δ measures the amount of information P reveals about the cryptographic key R. When evaluating security of algorithms, which derive secret information from noisy data, entropy measures like min-entropy, average min-entropy, and entropy loss are appealing since these measures have clear security applicability. However, these measures can only be applied to discrete random variable. In the case of continuous random variables, these measures depend on the precision used to represent the values of a random variable, as shown in the following example.

Example. Assume that all points X are real numbers between $[0, 1]$ and are uniformly distributed. Assume further that points in X are represented with 2-digit precision, which leads to a min-entropy $H_\infty(X) = \log_2 100$. If we choose to represent points with 4-digit precision the min-entropy of X becomes $H_\infty(X) = \log_2 10000$, which is higher then $H_\infty(X) = \log_2 100$ although in both cases X is uniformly distributed over the interval $[0, 1]$.

More examples related to average min-entropy and entropy loss can be found in the work of Li *et al.* [12]. We have chosen mutual information because it captures the measure of dependence between two random variables regardless of their types of distributions (discrete or continuous).

FUZZY EXTRACTOR AND FUZZY EMBEDDER. From *Definitions* 1 and 2, we argue that a fuzzy embedder may be more appealing than fuzzy extractor in practice, due to the following reasons:

1. A fuzzy embedder scheme accepts continuous data as input and can embed different keys. In contrast, in a practical deployment, a fuzzy extractor scheme must be

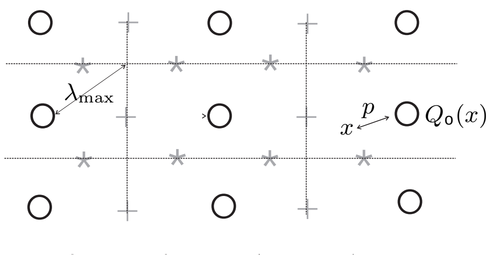

Fig. 5. Embed *function of* QIM*-fuzzy embedder*

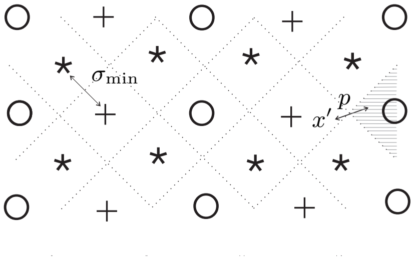

Fig. 6. Reproduce *function of a* QIM*-fuzzy embedder*

combined with quantization and re-randomization to achieve the same goals as a fuzzy embedder.

2. A fuzzy embedder construction leads to a fuzzy extractor construction. Given a $(\mathcal{U}, \ell, \rho, \epsilon, \delta)$-fuzzy embedder scheme, we can construct a fuzzy extractor scheme $\langle \mathsf{Generate}', \mathsf{Reproduce}' \rangle$ as follows:
   - $\mathsf{Generate}'$: $\mathcal{U} \rightarrow P \times R$. This algorithm takes $x \in \mathcal{U}$ as input, chooses $r \in R$, and returns $p = \mathsf{Embed}(x, r)$ and r.
   - $\mathsf{Reproduce}'$: $\mathcal{U} \times P \rightarrow R$. This algorithm takes $x' \in \mathcal{U}$ and $p \in P$ as input, and returns the value $\mathsf{Reproduce}(x, p)$.

4 A Practical Construction for Fuzzy Embedder

In this section, we present a general construction for fuzzy embedder using a QIM and analyze the performance of this construction in terms of reliability and security. We also investigate optimization issues when $\mathcal{U}$ is n-dimensional.

QIM-FUZZY EMBEDDER. A fuzzy embedder can be constructed from *any* QIM by defining the embed procedure as:

$$\mathsf{Embed}(x, r) = \mathtt{QIM}(x, r) - x,$$

and the reproduction procedure as the minimum distance Euclidean decoder:

$$\mathsf{Reproduce}(x', p) = \widetilde{Q}(x' + p),$$

where $\widetilde{Q} : \mathcal{U} \rightarrow R$ is defined as

$$\widetilde{Q}(y) = \underset{r \in R}{\operatorname{argmin}}\, d(y, \mathcal{M}_r).$$

Intuitively, our construction is a generalization of the scheme of Linnartz, *et al.* [13]. *Figures* 5 and 6 illustrate Embed and Reproduce, respectively, for a QIM ensemble of three quantizers $\{Q_{\mathsf{o}}, Q_{+}, Q_{\star}\}$. During embedding, the secret $r \in \{\mathsf{o}, \star, +\}$ selects a quantizer, say Q_{o}. The selected quantizer finds the reconstruction point $Q_{\mathsf{o}}(x)$ closest to x and the embedder returns the difference between the two as p, with $p \leq \lambda_{\max}$.

Reproduction from p and x' should return o only if $x' + p$ is in one of the Voronoi regions of $Q_{\mathtt{o}}$ (hatched area in *Figure* 6). Errors occur if $(x' + p)$ is not in any of the Voronoi regions of $Q_{\mathtt{o}}$, thus the size and shape (for $n \geq 2$) of the Voronoi region param eterized by the radius of the inscribed ball $\sigma_{\min}/2$ determines the probability of errors.

RELIABILITY. In the following lemma, we link the reliability of a `QIM`-fuzzy embedder to the size and shape of the Voronoi regions of the employed `QIM`.

Lemma 1 (Reliability). *Let* $\langle \mathsf{Embed}, \mathsf{Reproduce} \rangle$ *be a* $(\mathcal{U}, \ell, \rho, \epsilon, \delta)$ *`QIM`-fuzzy embedder, and let* X *be a random variable over* $\mathcal{U}$ *with joint density function* $f_X(x)$*. For any* $r \in R$*, we define*

$$\rho(r) = \int_{\mathcal{V}_r} f_X(y - \mathsf{Embed}(X, r))dy,$$

where $\mathcal{V}_r = \bigcup_{c \in \mathcal{M}_r} V_c$ *is the union of the Voronoi regions of all reconstruction points in* $\mathcal{M}_r$*. Then the reliability is equal to*

$$\rho = \min_{r \in R} \rho(r).$$

Proof: Since $\rho(r)$ is exactly the probability that an embedded key r will be reconstructed correctly, the statement follows from the definition. □

Most known noisy data, such as biometrics and PUFs, have two main properties: larger distances between x and the measurement x' are increasingly unlikely, and the noise is not directional. Thus the primary consideration for reliability is the size of the inscribed ball of the Voronoi regions, which has radius $\sigma_{\min}/2$.

Corrolary 1 (Bounding ρ) *In the settings of Lemma 1, the reliability parameter* ρ *can be bounded by*

$$\min_{r \in R} \sum_{c \in \mathcal{M}_r} \int_{B(c, \frac{\sigma_{\min}}{2})} f_X(y)dy \leq \rho$$

where $B(c, r)$ *is the ball centered in* c *with radius* r*.*

Proof. The above relation follows from the definition of reliability, since $S(c, \frac{\sigma}{2}) \subset V_c$ and $x + \mathsf{Embed}(X, r)$ is always a reconstruction point. □

Corollary 1 shows that reliability is at least the sum of all balls of radius $\frac{\sigma_{\min}}{2}$ inscribed in the Voronoi regions. Thus the size of the inscribed ball is an important parameter, which determines the reliability to noise.

SECURITY. In our construction, if an attacker learns the value x she can reproduce the value r from p. However, if it learns the secret key r, she could cannot exactly reproduce x, which is further illustrated in the following example

Example. In the fuzzy embedder example given in *Figure* 6, the attacker can choose between three different key values$\{\circ, +, \star\}$. Assume she learns the correct key, in our example $\circ$. To find the correct value for x she still has to decide which of the reconstruction points of the quantizer $Q_\circ$ is closest to x. Without any other information this is an impossible task since the quantizer $Q_\circ$ has an infinite number of reconstruction points.

Since the full disclosure of the string r is not enough to recover x, we can conclude that $\epsilon \leq \delta$. We now consider how large δ, the leakage on the key depending on P, which is a continuous variable in our construction. We know that any $p \in P$ has the property that $p \leq \lambda_{\max}$. A technical difficulty in characterizing the size of P arises as P is not necessarily discrete. Tuyls, *et al.* [19] show the following result, establishing a link between the continuous and the quantized version of P denoted here with P_d.

Lemma 2 (Tuyls et al. [19]). *For continuous random variables X, Y and $\xi > 0$, there exists a sequence of discretized random variables X_d, Y_d that converge pointwise to X, Y (when $d \to \infty$) such that for sufficiently large d, $I(X;Y) \geq I(X_d;Y_d) \geq I(X;Y) - \xi$.*

Since $I(R;P_d) \leq H(P_d) \leq |P_d|$, where $|P_d|$ is the size of the sketch. Thus it is best to have $|P_d|$ as small as possible. In our construction, we have $|P_d| \leq \lambda_{\max}$. Thus by bounding the size of p we bound the value of δ.

OPTIMIZATION. In this paragraph, we analyze the key length allowed by the restrictions placed by our performance criteria on the embed and reproduce procedures. Firstly, we take a look at the reproduce procedure which ties directly with the reliability. The minimum size of an error to produce a wrong decoding is $\sigma_{\min}/2$. Thus, the collection of balls centered in the reconstruction point of all quantizers with radius $\sigma_{\min}/2$ should be disjoint.

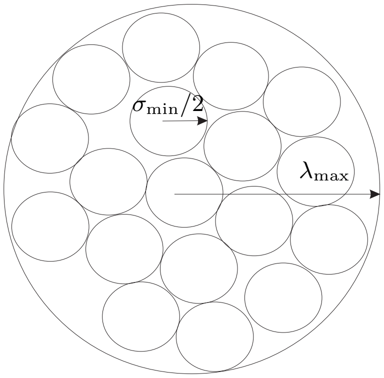

Fig. 7. *Optimization of reliability versus security. Reliability is determined by the size of the ball with radius $\sigma_{\min}/2$. Each small ball has associated to its center a different key $r \in R$. The number of small ball inside the large ball with radius $\lambda_{\max}$ is at least 2^l the number of elements in R. To have as many keys as possible we want to increase the number of small ball, thus we want* dense (sphere) packing. *The size of the public sketch $p \in P$ is at most $\lambda_{\max}$. Since for any $x \in \mathcal{U}$ we want to be within λ_{max} distance to a specific $r \in R$, large balls should* cover optimally *the space $\mathcal{U}$. When the point x falls in a region, which does not belong to any ball the reproduction procedure gives the closest center of a small ball, thus* we want polytopes which tile the space.

Secondly, the embed procedure has to be able to embed any key $r \in R$ into an arbitrary point x. Hence, for each key r the collection of balls centered in the reconstruction points of Q_k and with radius $\lambda_{\max}$ should cover the entire space $\mathcal{U}$. $\lambda_{\max}$ and $\lambda_{\min}$ can be linked as follows:

Lemma 3. *The covering distance of a `QIM` , defined in Section 2, is bounded by:*

$$\lambda_{\max} \geq \sqrt[n]{N}\frac{\sigma_{\min}}{2}$$

where n represents the dimension of the universe $\mathcal{U}$ and N is the number of different quantizers.

Proof: As noted above, all balls with radius $\sigma_{\min}/2$ centered in the centroids of the whole ensemble are disjoint. Each collection of balls with radius $\lambda_{\max}$ centered in the centroids of an individual quantizer gives a covering of the space $\mathcal{U}$, see *Figure* 7. Therefore, a ball with radius $\lambda_{\max}$, regardless of its center, contains at least the volume of N disjoint balls of radius $\sigma_{\min}/2$, one for each quantizer in the ensemble. Comparing the volumes, we have

$$s_n\lambda_{\max}^n \geq s_n N(\frac{\sigma_{\min}}{2})^n$$

where s_n is a constant only depending on the dimension. □

Consider the case when an intruder has partial knowledge about the random variable X. For example, she could know the average distribution of all (fingerprint) biometrics, or the average distribution of the PUFs. This average distribution is known in the literature as background distribution. While any `QIM`-fuzzy embedder achieves equiprobable keys if the background distribution on $\mathcal{U}$ is uniform, the equiprobability can break down when this background distribution is non-uniform and known to the intruder. A legitimate question is: *how can a `QIM`-fuzzy embedder achieve equiprobable keys when the background distribution is not uniform?*

In the literature [4,7,13], it is often assumed that the background distribution is a multivariate Gaussian. We make a much weaker assumption, namely the background distribution is not uniform but spherically symmetrical and decreasing. In other words, we assume that measurement errors of the noisy data only depend on the distance, and not on the direction, and that larger errors are less likely.

Thus, to achieve equiprobable keys given this background distribution, the reconstruction points must be equidistant as for example the construction in *Figure* 8 (a). Note that putting more small balls inside the large ball is not possible since they are not equiprobable. The problem with the construction in *Figure* 8 (a) is the size of the sketch which becomes large.

The natural question, which arise is: *what is the minimum sketch size attainable such that all keys are equiprobable for a given desired reliability?* This question naturally leads us to consider the kissing number $\tau(n)$, which is defined to be the maximum number of white n-dimensional spheres touching a black sphere of equal radius, see *Figure* 8 (b). The radius of the small balls determines reliability and the minimum $\lambda_{\max}$, such that a `QIM`-fuzzy embedder can be built is equal to the radius of the circumscribed ball of as shown in *Figure* 8 (b).

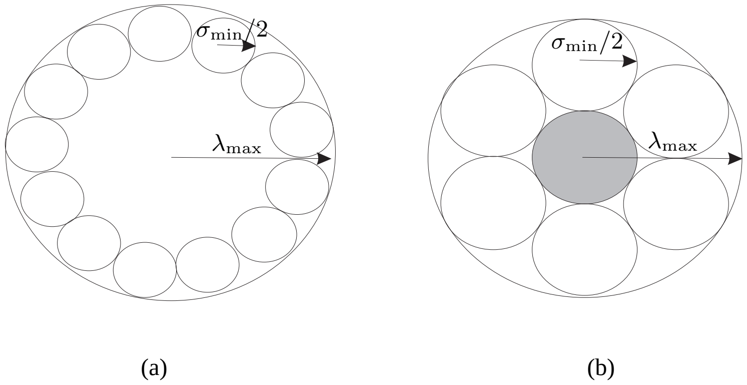

Fig. 8. *(a) Construction which yields equiprobable keys in case the background distribution is spherical symmetrical in the two dimensional space. (b) Optimal construction which results in minimal public sketch size and has equiprobable keys in the two dimensional space.*

The next question we ask is: *for a minimum sketch size and a given reliability, are there dimensions which are better then others?* For example why not pack spheres in the three dimensional space where the kissing number is 12. For the same reliability it is possible to obtain more keys? For most dimensions, only bounds on the kissing number are known [11,21]. Assuming a spherically symmetrical and decreasing background distribution, we have the following bound on equiprobable keys.

Theorem 1 (Optimal high dimensional packing.). *Assume the background distribution to be spherically symmetrical and decreasing. For a* $(\mathcal{U}, \ell, \rho, \epsilon, \delta)$ `QIM`*-fuzzy embedder with* $\dim(\mathcal{U}) = n$ *with equiprobable keys and minimal sketch size, we have that* $\ell \leq \tau(n)$.

Proof sketch: The target reliability ρ_0 will translate to a certain radius σ_0. In other words, we need to stack balls of radius σ_0 optimally. To achieve the maximum number of equiprobable keys without the sketch size getting too big, the best construction is to center the background distribution in one such ball, and to assign a different key to each touching ball. Thus the amount of possible equiprobable keys is upper bounded by the kissing number $\tau(n)$. □

From the known bounds on the kissing number [11,21], we have the following somewhat surprising conclusion:

Corrolary 2 *Assuming a spherically symmetrical and decreasing background distribution on* $\mathcal{U}$ *and equiprobable keys, for a* $(\mathcal{U}, \ell, \rho, \epsilon, \delta)$ `QIM`*-fuzzy embedder the most equiprobable keys are attained by quantizing two dimensions at a time, leading to* $N(n)$ *different keys, where*

$$N(n) = 6^{\lfloor \frac{n}{2} \rfloor} 2^{(n - 2\lfloor \frac{n}{2} \rfloor)}.$$

Proof: Known upper bounds [11] on the kissing number in n dimensions state that $\tau(n) \leq 2^{0.401n(1+o(1))}$. This means that $N(n) \geq \tau(n)$ in all dimensions, since $N(n) \approx$

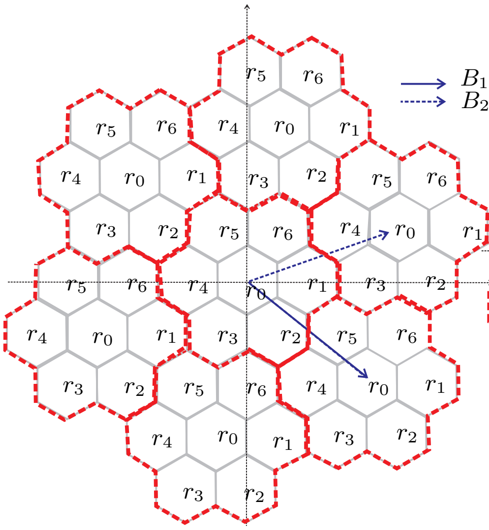
r_5 r_6
r_5 r_6 r_4 r_0 r_1
r_4 r_0 r_1 r_3 r_2 r_5 r_6
r_3 r_2 r_5 r_6 r_4 r_0 r_1
r_5 r_6 r_4 r_0 r_1 r_3 r_2
r_4 r_0 r_1 r_3 r_2 r_5 r_6
r_3 r_2 r_5 r_6 r_4 r_0 r_1
r_4 r_0 r_1 r_3 r_2
r_3 r_2
B_1
B_2

Fig. 9. Reproduce *function of 7-hexagonal tiling*

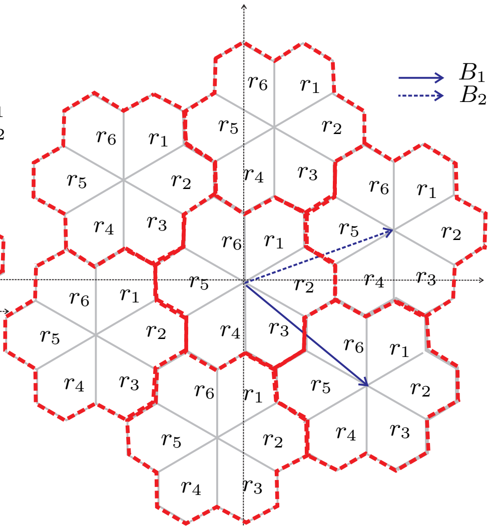
r_6 r_1
B_1
B_2
r_6 r_1 r_5 r_2
r_5 r_2 r_4 r_3 r_6 r_1
r_4 r_3 r_6 r_1 r_5 r_2
r_6 r_1 r_5 r_2 r_4 r_3
r_5 r_2 r_4 r_3 r_6 r_1
r_4 r_3 r_6 r_1 r_5 r_2
r_5 r_2 r_4 r_3
r_4 r_3

Fig. 10. Reproduce *function of 6-hexagonal tiling*

$2^{1.3n}$ and small dimensions can easily be verified by hand. Also note that $N(n_1 + n_2) \leq N(n_1)N(n_2)$. Thus quantizing dimensions pairwise gives the largest number of equiprobable keys for any spherically symmetric distribution. □

5 QIM-fuzzy embedder from 2-dimensional quantization

In this section we present our main construction, referred to as 6-hexagonal tiling, of QIM-fuzzy embedder by quantizing 2-dimensional subspaces of continuous and noisy data. We compare the performance with the 4-square tiling method introduced by Linnartz, *et al.* [13].

Preliminary concept Let the continuous and noisy data be represented with a n-dimensional variable $X = (X_1, X_2, \cdots X_n)$. We assume that n is even; otherwise one of the vector elements can be quantized with a 1-dimensional QIM as the one in our example in *Section* 2. Thus, X can be partitioned into $\frac{n}{2}$ 2-dimensional subspaces and each one can be considered separately. We take the subspace (X_1, X_2) as an example in the rest of this section. On the x-axis in *Figure* 9 we have the values for X_1 and on the y-axis we have the values of X_2. Along the z-axis (not shown in the figure) we have the joint probability density $f_{X_1X_2}(x)$.

Naturally, we want to choose the densest circle packing for the 2-dimensional space, where all circles have equal radius and the center of the circle is the reconstruction point which is associated with a key value. However, the circles do not tile the space so that, when x (the realization of X) falls into the non-covered region it cannot be associated with any reconstruction point. Therefore, we need to approximate the circle with some polygons that can tile the space. In 2-dimensional space, there are only three types of polygons: triangle, square, and hexagon. Since we assume a spherical symmetrical distribution for $f_{X_1X_2}$, hexagon is the best approximation to the circle from the reliability point of view.

5.1 Description of 6-hexagonal tiling

First attempt. In our construction, the reconstruction points of all quantizers are shifted versions of some base quantizer Q_0. A dither vector $\overrightarrow{v_r}$ is defined for each possible $r \in \mathcal{R}$. We define the *tiling polygon* as the repeated structure in the space that is obtained by decoding to the closest reconstruction point. It follows from this definition that the *tiling polygon* contains exactly one Voronoi region for each quantizer in the ensemble. In *Figures* 9 the *tiling polygons* are delimited by the dotted line. More specifically, we define a dithered QIM using an ensemble of 7 quantizers. The reconstruction points of the base quantizer Q_0 are defined by the lattice spanned by the vectors $\overrightarrow{B_1} = (5, \sqrt{3})q$, $\overrightarrow{B_2} = (4, -2\sqrt{3})q$, where q is the scaling factor of the lattice. In *Figure* 9 these points are labeled r_0. The other reconstruction points of quantizers Q_i $(1 \leq i \leq 6)$ are obtained by shifting the base quantizer by the dither vectors $\{\overrightarrow{v_1}, \cdots, \overrightarrow{v_6}\}$ such that $Q_i(x) = Q_0(\overrightarrow{x} + \overrightarrow{v_i})$. The values for these dither vectors are: $\overrightarrow{v_1} = (2, 0)$, $\overrightarrow{v_2} = (-3, \sqrt{3})$, $\overrightarrow{v_3} = (-1, -\sqrt{3})$, $\overrightarrow{v_4} = (-2, 0)$, $\overrightarrow{v_5} = (3, -\sqrt{3})$, and $\overrightarrow{v_6} = (1, \sqrt{3})$. The embed and reproduce procedures are defined in *Section* 4.

This construction (referred to as 7-hexagonal tiling) can embed $n \times \frac{\log_2 7}{2}$ bits, where n is the dimensionality of random variable X. It is optimal from the reliability point of view. However, assume that the background distribution is a spherical symmetrical distribution with mean centered in the origin of the coordinates. In the construction above the hexagon centered in the origin will typically have a higher associated probability than the off-center hexagons. This effect grows as we increase the scaling factor q of the lattice. Therefore, keys might be not equiprobable when the background distribution is not flat enough.

Improved construction. In the improved construction, namely 6-hexagonal tiling, we eliminate the middle hexagon to make all keys equiprobable (see *Figure* 10). Consequently, the tiling polygon is formed by 6 decision regions and thus there are only 6 dither vectors. As a result, the dither vectors, $\{\overrightarrow{v_1}, \cdots, \overrightarrow{v_6}\}$ are used to construct the quantizers, but the basic quantizer Q_0 itself is not used. The embed and reproduce procedures remain the same.

Our main construction can embed $n \times \frac{\log_2 6}{2}$ bits, where n is the dimensionality of random variable X. Compare with the first attempt, this construction is not optimal from the key length point of view. However, keys are equiprobable regardless of the background distribution, which we regard to be more favorable in cryptographic applications.

5.2 Comparison with 4-square tiling

We compare the performance between 6-hexagonal tiling and 4-square tiling in terms of reliability, the key length, and mutual information. Here we consider identically and in-

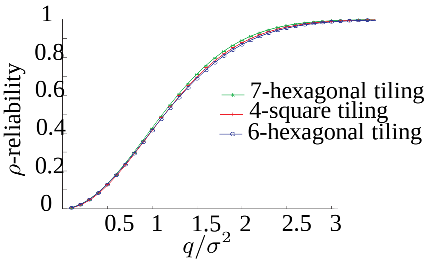

Fig. 11. *Reliability of the three* `QIM`*-fuzzy embedder constructions.*

dependently distributed (i.i.d) Gaussian sources. We assume that the background distribution has mean $(0, 0)$ and standard deviation ${\sigma_{X_1X_2}}^2$. We also assume that for any random $(X_1, X_2) \in \mathcal{U}^2$, the probability distribution of $f_{X_1X_2}(x)$ has mean $\mu = (\mu_1, \mu_2)$ and standard deviation σ_x^2. Note that these assumptions are abstracted from the area of biometrics (as an example of continuous and noisy data).

To evaluate the reliability relative to the quality of the source data (i.e., the amount of noise measured in the terms of standard deviation from mean), we compute probabilities associated with equal area decision regions, and the reconstruction point centered in the mean μ of the distribution $f_X(x)$. The curves in Figure 11 were obtained by progressively increasing the area of the Voronoi regions. The size of Voronoi region is controlled by the scaling factor of the lattice, namely q. From the figure, our 6-hexagonal tiling construction has a slightly better performance than the 4-square tiling method. This is because the regular hexagon best approximates a circle, the optimal geometrical form for a spherical symmetrical distribution. The key-length comparison is shown in Figure 12. Clearly, our 6-hexagonal tiling construction has a significantly better performance than the 4-square tiling method. Note that maximizing the key length means minimizing the probability for an attacker to guess the key correctly on her first try. The comparison of mutual information for the key when publishing the sketch is shown in Figure 13. Note that the values are scaled to the number of bits lost from each bit that is made public. From the figure, our 6-hexagonal tiling construction has a slightly better performance than the 4-square tiling method.

6 Conclusion

We have proposed a new primitive *fuzzy embedder* as a practical replacement for fuzzy extractor. Fuzzy embedder has solved two practical problems encountered when a fuzzy extractor scheme is used in practice: (1) fuzzy embedder naturally supports renewability, and (2) it supports direct analysis of quantization effects. We have also proposed a general construction of fuzzy embedder using a `QIM`. The `QIM` performance measures (in the context of watermarking) can be directly translated into the reliability and security properties of the constructed fuzzy embedder. When considering equiprobable keys, we have shown that quantizing dimensions pairwise gives the largest key length. We have proposed a concrete construction, namely 6-hexagonal tiling, and shown that

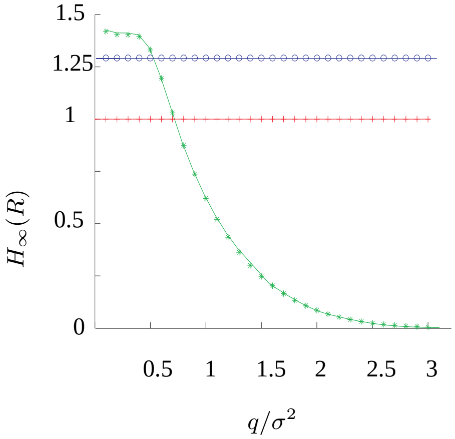

Fig. 12. *Key length comparison for the three* `QIM`*-fuzzy embedder constructions-scaled to one dimension*

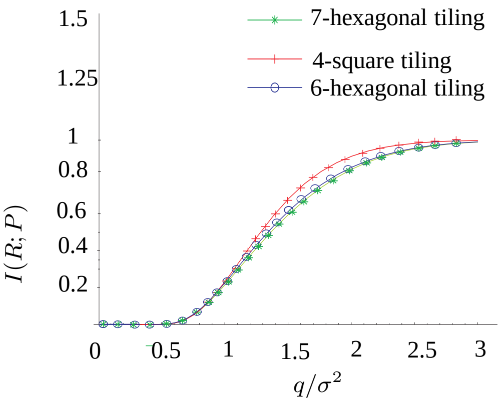

Fig. 13. *Mutual information between the key and the public sketch for the three* `QIM`*-fuzzy embedders*

it has a better performance than the 4-square tiling method introduced by Linnartz, *et al.* [13].

References

1. R.J. Barron, B. Chen, and G.W. Wornell. The duality between information embedding and source coding with side information and some applications. *IEEE Transactions on Information Theory*, 49(5):1159–1180, 2003.
2. X. Boyen. Reusable cryptographic fuzzy extractors. In Vijayalakshmi Atluri, Birgit Pfitzmann, and Patrick Drew McDaniel, editors, *ACM Conference on Computer and Communications Security*, pages 82–91. ACM, 2004.
3. I. Buhan, J. Doumen, P.H Hartel, and R.N.J Veldhuis. Fuzzy extractors for continuous distributions. In R. Deng and P. Samarati, editors, *Proceedings of the 2nd ACM Symposium on Information, Computer and Communications Security (ASIACCS), Singapore*, pages 353–355, New York, March 2007. ACM.
4. Y.J. Chang, W. Zhang, and T. Chen. Biometrics-based cryptographic key generation. In *International Conference on Multimedia and Expo (ICME)*, pages 2203–2206. IEEE, 2004.
5. B. Chen and G.W. Wornell. Quantization Index Modulation Methods for Digital Watermarking and Information Embedding of Multimedia. *The Journal of VLSI Signal Processing*, 27(1):7–33, 2001.
6. B. Chen and G.W. Wornell. Dither modulation: a new approach to digital watermarking and information embedding. *Proceedings of SPIE*, 3657:342, 2003.
7. C. Chen, R.N.J. Veldhuis, T.A.M. Kevenaar, and A.H.M. Akkermans. Multi-bits biometric string generation based on the likelyhood ratio. In *IEEE conference on Biometrics: Theory, Applications and Systems*, pages 1–6. IEEE, 2007.
8. Y. Dodis, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In Christian Cachin and Jan Camenisch, editors, *Advances in Cryptology - Eurocrypt 2004, International Conference on the Theory and Applications*

of Cryptographic Techniques, Interlaken, Switzerland, May 2-6, 2004, Proceedings, volume 3027 of *LNCS*, pages 523–540. Springer, 2004.

9. A. Gersho. Principles of quantization. *IEEE Transactions on Circuits and Systems*, 25(7):427–436, 1978.
10. A. Gersho. Asymptotically optimal block quantization. *IEEE Transactions on Information Theory,*, 25(4):373–380, 1979.
11. G.A. Kabatiansky and V.I Levenshtein. Bounds for packings on a sphere and in space. *Problemy Peredachi Informatsii*, 1:3–25, 1978.
12. Q. Li, Y. Sutcu, and N. Memon. Secure sketch for biometric templates. In Xuejia Lai and Kefei Chen, editors, *ASIACRYPT*, volume 4284 of *LNCS*, pages 99–113. Springer, 2006.
13. J.P. Linnartz and P. Tuyls. New shielding functions to enhance privacy and prevent misuse of biometric templates. In Josef Kittler and Mark S. Nixon, editors, *AVBPA*, volume 2688 of *LNCS*, pages 393–402. Springer, 2003.
14. U. Maurer. Perfect cryptographic security from partially independent channels. In *Proceedings of the 23rd ACM Symposium on Theory of Computing (STOC)*, pages 561–572. ACM Press, August 1991.
15. U. Maurer. Secret key agreement by public discussion. *IEEE Transaction on Information Theory*, 39(3):733–742, May 1993.
16. P. Moulin and R. Koetter. Data-hiding codes. *Proceedings of the IEEE*, 93(12):2083–2126, 2005.
17. B. Skoric, P. Tuyls, and W. Ophey. Robust key extraction from physical uncloneable functions. In John Ioannidis, Angelos D. Keromytis, and Moti Yung, editors, *ACNS*, volume 3531 of *Lecture Notes in Computer Science*, pages 407–422, 2005.
18. P. Tuyls, A. Akkermans, T. Kevenaar, G. Schrijen, A. Bazen, and R. Veldhuis. Practical biometric authentication with template protection. In Takeo Kanade, Anil K. Jain, and Nalini K. Ratha, editors, *AVBPA*, volume 3546 of *LNCS*, pages 436–446. Springer, 2005.
19. P. Tuyls and J. Goseling. Capacity and examples of template-protecting biometric authentication systems. In Davide Maltoni and Anil K. Jain, editors, *ECCV Workshop BioAW*, volume 3087 of *LNCS*, pages 158–170. Springer, 2004.
20. U. Uludag, S. Pankanti, S. Prabhakar, and A. K. Jain. Biometric cryptosystems: Issues and challenges. *Proceedings of the IEEE*, 92(6):948–960, 2004.
21. K. Zeger and A. Gersho. Number of nearest neighbors in a euclidean code. *IEEE Transactions on Information Theory*, 40(5):1647–1649, 1994.