Chapter 13

Algorithms for Processing Biometric Data Oriented to Privacy Protection and Preservation of Significant Parameters

Vladimir B. Balakirsky and A. J. Han Vinck

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/51800

1. Introduction

We address a general theory of transformations of biometric data, independently on particular biometric features taken into the processing. An application assumes representation of data in the specified format and the use of the proposed technique.

The content of a biometric database can be used for different purposes, and their complete list is usually unknown in advance. These purposes include different variants of processing noisy data, such as authentication and identification of a person on the basis of his biometric observations, detection of certain features, etc. General requirements to data processing schemes in any of these applications contain the design of a verification scheme consisting of the enrollment and the verification stages. At the enrollment stage, the input data should be converted to the data stored in the database under the identifier of the person. At the verification stage, the presented data have to be analyzed to make the decision whether they belong to the chosen person or not. Privacy protection of the data includes protection against attackers of three types:

- a blind attacker, who wants to guess the content of the database using his knowledge about the probability distribution over input biometric data;
- an attacker, who has access to the database and wants to find the input biometric vector processed at the enrollment stage;
- an attacker, who has access to the database and wants to generate artificial data to pass through the verification stage with the acceptance decision.

We will present the class of transformations having the following features.

- If input data are represented by the vector whose components are generated by a stationary memoryless source having the known probability distribution, then they can be converted to the vector $\boldsymbol{x}$ whose components are uniformly distributed over the $(-1,+1)$ interval. Therefore, the scheme has a perfect protection against blind attackers. A generalized version of the procedure brings the same property for non-stationary sources. Moreover, if the source has memory or the input probability distribution is unknown, an approximate uniform distribution can be created.
- The enrollment can be organized in such a way that the constructed vector $\boldsymbol{x}$ is encrypted and mapped to the vector $\hat{\boldsymbol{x}}$, which is stored in the database. The encryption is understood as replacement of certain components of the vector $\boldsymbol{x}$ by randomly chosen components. As a result, the input biometric data cannot be reconstructed from the vector $\hat{\boldsymbol{x}}$. We show that the encryption can be organized in such a way that the properties of the constructed uniform distribution are conserved, but the union of $(-1,-a)$ and $(+a,+1)$ intervals replaces the $(-1,+1)$ interval. The value of parameter $a \in (0,1)$ controls the trade-off between privacy protection and the verification performance, and it has be assigned in advance. As a result, the scheme becomes protected against attackers, who have access to the database and want to guess the input vector by reading the corresponding vector stored in the database.
- The case, when biometrics of the same person is measured at the enrollment and the verification stages, is simulated as transmission of outcomes of the enrollment stage measurements over an observation channel. We create another channel between results of transformations of these outcomes. It turns out that this channel has the property that the level of noise essentially depends on the magnitude of the transmitted component, and it is very low when the magnitude is large. Since a large magnitude at the output of the transformer is obtained when the input magnitude is large, we talk about the preservation of significant parameters under the noise, provided that these parameters are represented in a vector of outcomes of biometric measurements by components having large magnitudes. We present a verification algorithm designed on the basis of these properties.

Verification algorithms are considered in the most of publications, related to biometrics (see, in particular [1]–[4]), and they are usually introduced as algorithms of combinatorial matching of the outcomes of observations received at the enrollment and at the verification stages. However, secrecy requirements to a biometric system to be designed do not allow storage of the outcomes received at the enrollment stage, and lead to the schemes where only relatively few outcomes characterizing the person are taken into account. In the present chapter, these outcomes are referred to as significant components. The algorithms with similar understanding are presented in [6]–[8] where significance is determined by the values of the probability distribution function computed at the observed values. This approach follows the lines of information theory when data processing is represented as secret sharing [9]–[12]. Some part of the secret is published, while another part is hidden by the so–called one-way hash function [13] and it has to be decoded after a noisy version of the observation data is available. The transformation of the outcomes, received at the enrollment stage, to a published part of the secret is also viewed as wrapping in a number of applications where the unwrapping is possible only when outcomes, received at the verification stage, are close to the wrapped data [14]. The use of possibly non-stationary probability distributions also allows us to include multi–biometric measurements (see, for example [15], [16]) into the

http://dx.doi.org/10.5772/51800

processing. Furthermore, the addressed issues are relevant to constructing fault–tolerant passwords from biometric data [17]–[19]. The cited list of publications is certainly far from being complete, but we only indicated directions that are relevant to the material of the chapter. We also understand that specific applications of the presented results need research on probability distributions of the measured biometric data. In particular, the approaches can be fruitful for the schemes where different lengths are processed (for example, distances between certain points on the face or hand).

The chapter is organized as follows. We begin with the introduction and the notation sections and then introduce the *F*-transformation for the input data and their noisy observations in Sections 3,4. A possible implementation of the verification scheme, constructed using the derived properties, is presented in Section 5. Some general conclusions are included in Section 6.

Suppose that outcomes of biometric measurements of a person, received at the enrollment stage, are represented by a float–valued vector $\boldsymbol{r} = (r_1, \ldots, r_n) \in \mathsf{R}^n$. The t-th component of the vector $\boldsymbol{r}$ has the sign, which is understood as

$$\mathrm{sgn}(r_t) = \begin{cases} 1, \text{ if } r_t \geq 0 \\ 0, \text{ if } r_t < 0 \end{cases} \tag{1}$$

and the magnitude

$$|r_t| = \begin{cases} +r_t, \text{ if } r_t \geq 0 \\ -r_t, \text{ if } r_t < 0 \end{cases} \tag{2}$$

Thus, the specification of the component r_t is equivalent to the specification of the pair $(\mathrm{sgn}(r_t), |r_t|)$. Such a representation is introduced, because we assume that $\mathrm{sgn}(r_t)$ and $|r_t|$ are independent parameters: if one knows the sign, then he has no information about the magnitude; if one knows the magnitude, then he has no information about the sign. Furthermore, we will assume that the magnitude $|r_t|$ characterizes "significance" of the t-th component: for example, if r_t is defined as the difference between the value of the measured parameter and the average or the expected value, then $|r_t|$ determines the deviation of the t-th parameter for the particular person.

We will analyze the scheme where one of results of the processing of the vector $\boldsymbol{r}$ at the enrollment stage is expressed by a float–valued vector $\tilde{\boldsymbol{x}} = (\tilde{x}_1, \ldots, \tilde{x}_n) \in \mathsf{F}^n$ whose components have magnitudes less than 1, i.e.,

$$\mathsf{F} = \left\{ x \in \mathsf{R} : \ |x| < 1 \right\} \tag{3}$$

The vector $\tilde{\boldsymbol{x}}$ is stored in the database under the identifier of the person, associated with the vector $\boldsymbol{r}$. We call it the wrapped version of the input data. The transformation $\boldsymbol{r} \to \tilde{\boldsymbol{x}}$ is not a one-to-one mapping, and some data are lost. This transformation is divided into two steps. We first introduce a one-to-one mapping $\boldsymbol{r} \leftrightarrow \boldsymbol{x}$, where $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathsf{F}^n$, and

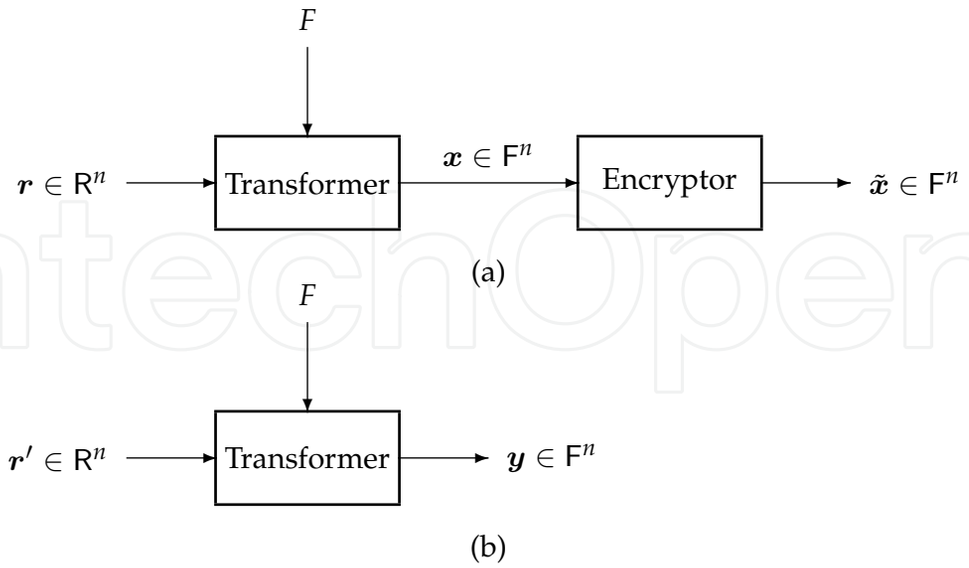

Figure 1. (a) Transformations of the vector $\boldsymbol{r}$ to the vector $\tilde{\boldsymbol{x}}$ stored in the database under the identifier of the person. (b) Transformation of the vector $\boldsymbol{r}'$ to the vector $\boldsymbol{y}$ at the verification stage for its comparison with the vector $\tilde{\boldsymbol{x}}$ and investigation of the closeness of the vectors $\boldsymbol{r}$ and $\boldsymbol{r}'$.

then encrypt the vector $\boldsymbol{x}$ in such a way that the change of the t-th component x_t is possible only if $|r_t|$ is small.

At the verification stage, the verifier observes a vector $\boldsymbol{r}' = (r'_1, \ldots, r'_n) \in \mathsf{R}^n$. This vector is mapped to the vector $\boldsymbol{y}$ in the same way as $\boldsymbol{r}$ was mapped to $\boldsymbol{x}$, and there should be an algorithm that analyzes the pair $(\boldsymbol{y}, \tilde{\boldsymbol{x}})$ to find the closeness of the vector $\boldsymbol{r}'$ and some vector $\boldsymbol{r}$ that could be used as an origin of the vector $\tilde{\boldsymbol{x}}$. If the verifier decides that these vectors are close enough, then components of the vector $\boldsymbol{r}'$ are considered as possible outcomes of biometric measurements of the person whose identifier corresponds to the vector $\tilde{\boldsymbol{x}}$.

The procedures above describe a general structure of the verification scheme under constraints that the transformation $\boldsymbol{r} \to \tilde{\boldsymbol{x}}$ is divided into two steps and that components of the constructed vectors belong to the set F. These procedures are illustrated in Figure 1 where we use some additional notation. Namely, we parameterize the mappings $\boldsymbol{r} \leftrightarrow \boldsymbol{x}$ and $\boldsymbol{r}' \leftrightarrow \boldsymbol{y}$ by a monotone increasing function $F(r), r \in \mathsf{R}$, approaching 0 when $r \to -\infty$ and $r \to +\infty$, respectively. Formally,

$$\frac{d}{dr}F(r) > 0, \quad r \in \mathsf{R} \tag{4}$$

and

$$(F(-\infty), F(+\infty)) = (0, 1) \tag{5}$$

By (4), (5), the function F has a uniquely determined inverse function F^{-1}, and we can simulate the data received at the verification stage, as the result of transmission of the vector $\boldsymbol{x}$ over a channel consisting of the deterministic F^{-1}-transformation $\boldsymbol{x} \to \boldsymbol{r}$, the stochastic mapping $\boldsymbol{r} \to \boldsymbol{r}'$ introduced by a physical V^n channel, and the deterministic F-transformation $\boldsymbol{r}' \to \boldsymbol{y}$ (see Figure 2a). As the verifier also has access to the vector $\tilde{\boldsymbol{x}}$

http://dx.doi.org/10.5772/51800

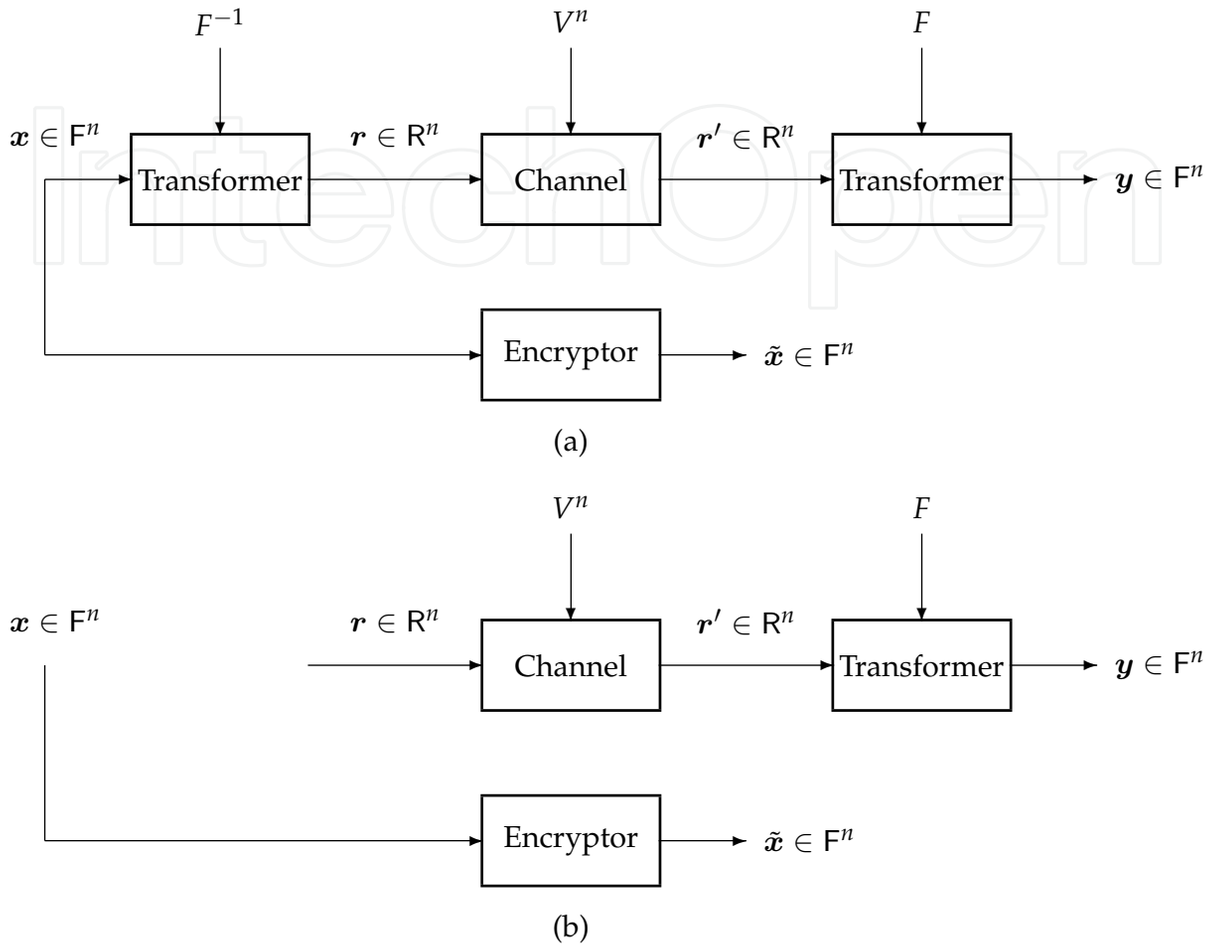

Figure 2. Schemes for transmission of the vector $\boldsymbol{x}$ to the verifier: (a) Legitimate case. (b) Attack.

constructed as an encrypted version of the vector $\boldsymbol{x}$, we get the situation, when the vector $\boldsymbol{x}$ is sent to the verifier over two parallel channels. If the level of noise in the physical channel is not high, then the verifier is supposed to make the acceptance decision. As an alternative, there is a possibility that the vector $\boldsymbol{r}$ is generated independently of the vector $\boldsymbol{x}$ (see Figure 2b where we delete the F^{-1}-transformation). In this case, the verifier is supposed to make the rejection decision.

2. Notation

Let R, R', X, Y denote random variables and r, r', x, y denote their values. This notation is also extended to the vectors of random variables $R^n, (R')^n, X^n, Y^n$ and their values $\boldsymbol{r}, \boldsymbol{r}', \boldsymbol{x}, \boldsymbol{y}$.

In the basic model, we assume that the vector $\boldsymbol{r}$ is generated by a stationary memoryless source having the probability distribution (PD) F^* and the probability density function (PDF) P^*,

$$F^* = \Big(F^*(r) = \Pr_{\text{data}} \{ R < r \}, \, r \in \mathsf{R} \Big) \tag{6}$$

$$P^* = \Big(P^*(r) = \frac{d}{dr} F^*(r), \; r \in \mathsf{R} \Big) \tag{7}$$

The value of the PDF, associated with the vector $\boldsymbol{r}$, is expressed as

$$\mathrm{PDF}(\boldsymbol{r}) = \prod_{t=1}^{n} P^*(r_t) \tag{8}$$

We will simulate the stochastic mapping $\boldsymbol{r} \to \boldsymbol{r}'$ as transmission of the vector $\boldsymbol{r}$ over an observation channel, which is introduced as a stationary memoryless channel specified by the collection of the conditional PDs Φ_r and the collection of the conditional PDFs V_r,

$$\Phi_r = \Big(\Phi(r'|r) = \Pr_{\text{noise}} \{ R' < r' \mid R = r \}, \, r' \in \mathsf{R} \Big) \tag{9}$$

$$V_r = \Big(V(r'|r) = \frac{d}{dr'} \Phi(r'|r), \, r' \in \mathsf{R} \Big) \tag{10}$$

for all $r \in \mathsf{R}$. The value of the conditional PDF, associated with the output vector $\boldsymbol{r}'$, given the input vector $\boldsymbol{r}$, is expressed as

$$\mathrm{PDF}(\boldsymbol{r}'|\boldsymbol{r}) = \prod_{t=1}^{n} V(r'_t|r_t) \tag{11}$$

Let

$$F\mathrm{G}_{m,\gamma} = \Big(F\mathrm{G}_{m,\gamma}(r) = \frac{1}{2} + \frac{1}{2} \mathrm{erf}\Big(\frac{r-m}{\gamma\sqrt{2}} \Big), \, r \in \mathsf{R} \Big) \tag{12}$$

$$\mathrm{G}_{m,\gamma} = \Big(\mathrm{G}_{m,\gamma}(r) = \frac{1}{\gamma\sqrt{2\pi}} \exp\Big\{ -\frac{(r-m)^2}{2\gamma^2} \Big\}, \, r \in \mathsf{R} \Big) \tag{13}$$

where

$$\mathrm{erf}(r) = \frac{2}{\sqrt{\pi}} \int_0^r \exp\{-\tilde{r}^2\} \, d\tilde{r}, \;\; r \in \mathsf{R} \tag{14}$$

is the erf-function, denote the Gaussian PD and PDF, when m is the mean and γ^2 is the variance. We will also use the function

$$\psi_c(x) = \frac{1}{2} + \frac{1}{2} \mathrm{erf}\Big(c \cdot \mathrm{erf}^{-1}(x) \Big) \tag{15}$$

http://dx.doi.org/10.5772/51800

3. The F-transformation of the input data

The function F, satisfying (4), (5), is the PD of some random variable R, and we write

$$F = \Big(F(r) = \Pr\{ R < r \},\, r \in \mathsf{R} \Big) \tag{16}$$

The corresponding PDF is defined as

$$P = \Big(P(r) = \frac{d}{dr} F(r),\, r \in \mathsf{R} \Big) \tag{17}$$

Notice that $(F^*, P^*) \neq (F, P)$ in general case and denote

$$f^* = \Big(f^*(x) = \Pr_{\text{data}} \{ 2F(R) - 1 < x \},\, x \in \mathsf{F} \Big) \tag{18}$$

$$p^* = \Big(p^*(x) = \frac{d}{dx} f^*(x),\, x \in \mathsf{F} \Big) \tag{19}$$

Let us fix the function F, satisfying (4), (5), and map an $r \in \mathsf{R}$ to the $x \in \mathsf{F}$ using the rule

$$x = 2F(r) - 1 \tag{20}$$

The function F has a uniquely determined inverse function

$$F^{-1} = \Big(F^{-1}(z),\, z \in (0,1) \Big) \tag{21}$$

and the equality (20) implies $r = r(x)$, where

$$r(x) = F^{-1}\Big(\frac{x+1}{2} \Big) \tag{22}$$

The F-transformation of the vector $\boldsymbol{r} \in \mathsf{R}^n$ will be understood as the result of n component-wise applications of the mapping (20).

Some values of $r(x)$ for Gaussian data are given in Table 1. Notice that

$$F = FG_{0,\rho} \;\Rightarrow\; \frac{1}{2} + \frac{1}{2}\mathrm{erf}\Big(\frac{r(x)}{\rho\sqrt{2}} \Big) = \frac{x+1}{2} \;\Rightarrow\; r(x) = \rho\sqrt{2} \cdot \mathrm{erf}^{-1}(x) \tag{23}$$

The data in Table 1 illustrate the point that (20) is a non-linear mapping. In particular, if $\rho = 2$, then the values belonging to the interval $(0, 0.64)$ are mapped to the values between 0 and 0.25, ..., the values greater than 3.92 are mapped to the values between 0.95 and 1. The mapping $r \leftrightarrow x$ is also illustrated in Figure 3.

$\mathbf{x} =$	**0**	**0.25**	**0.50**	**0.75**	**0.80**	**0.85**	**0.90**	**0.95**	**1**
$\rho = 2$	0	0.64	1.35	2.30	2.56	2.88	3.29	3.92	$+\infty$
$\rho = 1$	0	0.32	0.67	1.15	1.28	1.44	1.64	1.96	$+\infty$
$\rho = 1/2$	0	0.16	0.34	0.58	0.64	0.72	0.82	0.98	$+\infty$

Table 1. Some values of $r(x)$ for the function $F = FG_{0,\rho}$.

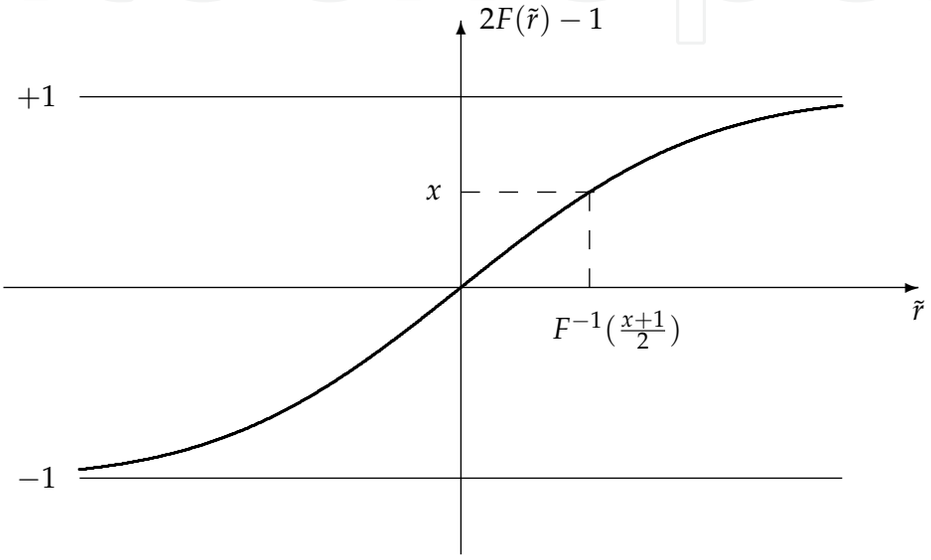

Figure 3. Illustration of the one-to-one mapping $x \leftrightarrow r = r(x)$.

In the following considerations we will restrict ourselves to the class of functions F, satisfying (4), (5) and having additional symmetric properties:

$$r(-x) = -r(+x) \tag{24}$$

and

$$P(r(-x)) = P(r(+x)), \quad x \in \mathsf{F} \tag{25}$$

The F-transformation of input data can be viewed as an application of the inverse to the well-known construction of pseudorandom generators having a fixed probability distribution F. In this case, an algorithm generates the value z of a random variable "uniformly distributed over the $(0,1)$ interval" and outputs $F^{-1}(z) \in (0,1)$. Our scheme receives an $r \in \mathsf{R}$ and outputs $2F(r) - 1$ where the multiplication by 2 and the subtraction of 1 are introduced because of further processing of the signs and the magnitudes. Notice that a similar approach can be used to generate Gaussian variables having the mean 0 and the fixed variance ρ^2; the transformer has to output $FG^{-1}_{0,\rho}(F(r))$ in this case.

http://dx.doi.org/10.5772/51800

As $2F(R)-1$ is a deterministic function of the random variable R, the value of x is the observation of the random variable $X = 2F(R)-1$ having the PD f^* and the PDF p^*. We write

$$f^*(x) = \Pr_{\text{data}}\{\, 2F(R)-1 < x \,\} = \Pr_{\text{data}}\Big\{\, F(R) < \frac{x+1}{2} \,\Big\} = \Pr_{\text{data}}\Big\{\, R < F^{-1}\Big(\frac{x+1}{2}\Big) \,\Big\}$$
$$= F^*\Big(F^{-1}\Big(\frac{x+1}{2}\Big)\Big) = F^*(r(x)) \tag{26}$$

and

$$p^*(x) = \frac{d}{dx}f^*(x) = \frac{d}{dx}F^*(r(x))$$
$$= \Big(\frac{d}{d\tilde{r}}F^*(\tilde{r})\Big|_{\tilde{r}=r(x)}\Big)\cdot\frac{d}{dx}r(x) = P^*(r(x))\cdot\frac{d}{dx}F^{-1}\Big(\frac{x+1}{2}\Big)$$
$$= P^*(r(x))\cdot\Big(\frac{d}{d\tilde{r}}F(\tilde{r})\Big|_{\tilde{r}=r(x)}\Big)^{-1}\cdot\frac{d}{dx}\frac{x+1}{2} = \frac{1}{2}\cdot\frac{P^*(r(x))}{P(r(x))} \tag{27}$$

Therefore

$$f^* = \Big(\, F^*(r(x)),\ x\in\mathsf{F}\,\Big),\quad p^* = \Big(\,\frac{P^*(r(x))}{2P(r(x))},\ x\in\mathsf{F}\,\Big) \tag{28}$$

By (8) and the fact that the F-transformation is a component-wise mapping, the value of the PDF, associated with the vector $\boldsymbol{x}$, is expressed as

$$\mathrm{PDF}(\boldsymbol{x}) = \prod_{t=1}^{n}\frac{P^*(r(x_t))}{2P(r(x_t))} \tag{29}$$

for all $\boldsymbol{x}\in\mathsf{F}^n$.

If $(F^*,F) = (F\mathrm{G}_{0,\rho^*}, F\mathrm{G}_{0,\rho})$, then we use (23) and write

$$f^*(x) = F\mathrm{G}_{0,\rho^*}(\rho\sqrt{2}\cdot\mathrm{erf}^{-1}(x)) = \frac{1}{2}+\frac{1}{2}\mathrm{erf}\Big(\frac{\rho\sqrt{2}\cdot\mathrm{erf}^{-1}(x)}{\rho^*\sqrt{2}}\Big) \tag{30}$$

$$p^*(x) = \frac{\mathrm{G}_{0,\rho^*}(r(x))}{2\mathrm{G}_{0,\rho}(r(x))} = \frac{\rho\sqrt{2\pi}}{2\rho^*\sqrt{2\pi}}\exp\Big\{-\frac{r^2(x)}{2(\rho^*)^2}+\frac{r^2(x)}{2\rho^2}\Big\} \tag{31}$$

Hence,

$$f^* = \Big(\,\psi_{\rho/\rho^*}(x),\ x\in\mathsf{F}\,\Big),\quad p^* = \Big(\,\frac{\rho}{2\rho^*}\exp\Big\{-\frac{\rho^2-(\rho^*)^2}{2\rho^2(\rho^*)^2}r^2(x)\Big\},\ x\in\mathsf{F}\,\Big) \tag{32}$$

where $\psi_{\rho/\rho^*}(x)$ is defined by (15) with $c=\rho/\rho^*$. Examples of the functions $f^*(x)$ and $p^*(x)$ are given in Figures 4, 5.

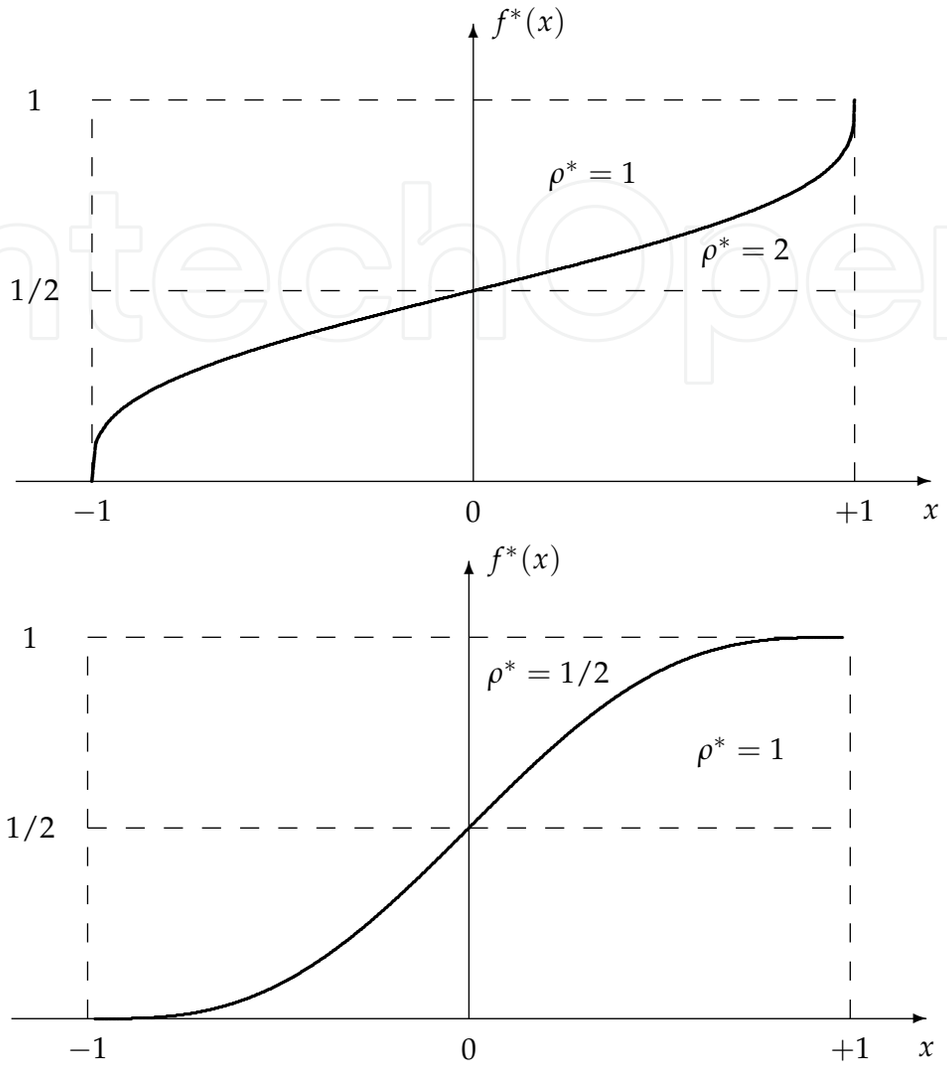

Figure 4. Examples of the PDs f^*, when $(F, F^*) = (FG_{0,1}, FG_{0,\rho^*})$.

Suppose that $F^* = F$. Then

$$f^*(x) = F\Big(F^{-1}\Big(\frac{x+1}{2}\Big)\Big) = \frac{x+1}{2}, \;\; p^*(x) = \frac{d}{dx}\frac{x+1}{2} = \frac{1}{2} \tag{33}$$

and

$$f^* = \Big(\frac{x+1}{2},\; x \in \mathsf{F}\Big), \;\; p^* = \Big(\frac{1}{2},\; x \in \mathsf{F}\Big) \tag{34}$$

i.e., X is a random variable, uniformly distributed over the set F. If the vector $\boldsymbol{x}$ is stored, then the database is perfectly protected against attackers, who want to guess its content. If $F^* \neq F$, then we get an approximate uniform distribution. However, $\boldsymbol{r} \to \boldsymbol{x}$ is a one-to-one mapping, and the attackers, who have access to the database, can reconstruct the vector $\boldsymbol{r}$.

http://dx.doi.org/10.5772/51800

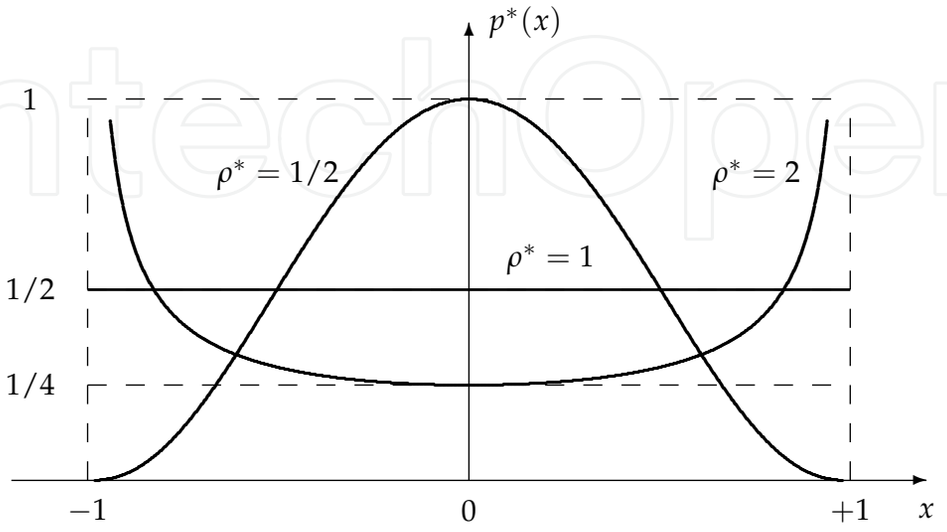

Figure 5. Examples of the PDFs p^*, when $(F, F^*) = (F\mathrm{G}_{0,1}, F\mathrm{G}_{0,\rho^*})$.

Our assumption that the vector $\boldsymbol{r}$ is generated by a known stationary memoryless source, and the value of the PDF, associated with this vector, is expressed by (8), can contradict to the practice. If the source is non–stationary, and $P_1^*(r_1), \ldots, P_n^*(r_n)$ replace the multipliers at the right-hand side of (8), then considerations above are directly extended by replacing the function F with the functions

$$F_t = \Big(F_t(r) = \Pr\{\, R_t < r \,\},\, r \in \mathsf{R} \Big),\ \ t = 1, \ldots, n \tag{35}$$

that approximate the PDs of components of the input vector. A more general assignment corresponds to the sources with memory, when the PDs and the PDFs are given as

$$F^*(\boldsymbol{r}_{t-1}) = \Big(F(r|\boldsymbol{r}_{t-1}) = \Pr_{\text{data}}\{\, R_t < r \,|\, (R_1, \ldots, R_{t-1}) = \boldsymbol{r}_{t-1} \,\},\, r \in \mathsf{R} \Big) \tag{36}$$

$$P^*(\boldsymbol{r}_{t-1}) = \Big(P(r|\boldsymbol{r}_{t-1}) = \frac{d}{dr} F(r|\boldsymbol{r}_{t-1}),\, r \in \mathsf{R} \Big) \tag{37}$$

where $\boldsymbol{r}_{t-1} = (r_1, \ldots, r_{t-1})$ and $t = 1, \ldots, n$. Then

$$\mathrm{PDF}(\boldsymbol{r}) = \prod_{t=1}^{n} P^*(r_t|\boldsymbol{r}_{t-1}) \tag{38}$$

Let

$$x_t = 2F_t(r_t) - 1, \;\; r_t(x_t) = F_t^{-1}\Big(\frac{x_t+1}{2}\Big) \tag{39}$$

and $\boldsymbol{x}_{t-1} = (x_1, \ldots, x_{t-1})$, $r(\boldsymbol{x}_{t-1}) = (r_1(x_1), \ldots, r_{t-1}(x_{t-1}))$. We also denote

$$f^*(\boldsymbol{x}_{t-1}) = \Big(f(x|\boldsymbol{x}_{t-1}) = \Pr_{\text{data}}\{\, X_t < x \,|\, (X_1, \ldots, X_{t-1}) = \boldsymbol{x}_{t-1} \,\},\; x \in \mathsf{F} \Big) \tag{40}$$

$$p^*(\boldsymbol{x}_{t-1}) = \Big(p(x|\boldsymbol{x}_{t-1}) = \frac{d}{dx} f(x|\boldsymbol{x}_{t-1}),\; x \in \mathsf{F} \Big) \tag{41}$$

Then

$$f^*(\boldsymbol{x}_{t-1}) = \Big(F_t^*(r_t(x)|r(\boldsymbol{x}_{t-1})),\; x \in \mathsf{F} \Big) \tag{42}$$

$$p^*(\boldsymbol{x}_{t-1}) = \Big(\frac{P^*(r_t(x)|r(\boldsymbol{x}_{t-1}))}{2P_t(r_t(x))},\; x \in \mathsf{F} \Big) \tag{43}$$

These formulas allow us to extend statistical properties of the F-transformation of the input data to the general case.

4. The F-transformation of noisy observations of the input data

Let us denote

$$\varphi_x = \Big(\varphi(y|x) = \Pr_{\text{noise}}\{\, 2F(R') - 1 < y \,|\, 2F(R) - 1 = x \}, \, y \in \mathsf{F} \Big) \tag{44}$$

$$v_x = \Big(v(y|x) = \frac{d}{dy}\varphi(y|x), \, y \in \mathsf{F} \Big) \tag{45}$$

for all $x \in \mathsf{F}$. If $(\Phi_r, V_r) = (F\mathrm{G}_{r,\sigma}, \mathrm{G}_{r,\sigma})$ for all $r \in \mathsf{R}$, then the observation channel is an additive white Gaussian noise channel having the variance of the noise equal to σ^2.

Let us map an $r' \in \mathsf{R}$ to the $y \in \mathsf{F}$ in the same way as an $r \in \mathsf{R}$ was mapped to the $x \in \mathsf{F}$, i.e., $y = 2F(r') - 1$ and $r' = r(y)$, where $r(y) = F^{-1}\Big(\frac{y+1}{2}\Big)$. Notice that the value of y is the observation of the random variable $Y = 2F(R') - 1$ having the conditional PD φ_x and the conditional PDF v_x, given $X = x$, since $2F(R') - 1$ is a deterministic function of the random variable R'. The result of the F-transformation of the vector $\boldsymbol{r}' \in \mathsf{R}^n$, will be understood is the vector $(2F(r'_1) - 1, \ldots, 2F(r'_n) - 1)$.

One can see that the stochastic dependence between random variables R and R' is translated to the stochastic dependence between random variables X and Y in such a way that

$$\begin{aligned}
\varphi(y|x) &= \Pr_{\text{noise}}\{\, Y < y \,|\, X = x \,\} = \Pr_{\text{noise}}\Big\{\, 2F(R') - 1 < y \,|\, 2F(R) - 1 = x \,\Big\} \\
&= \Pr_{\text{noise}}\Big\{\, R' < F^{-1}\Big(\frac{y+1}{2}\Big) \,\Big|\, R = F^{-1}\Big(\frac{x+1}{2}\Big) \,\Big\} \\
&= \Pr_{\text{noise}}\{\, R' < r(y) \,|\, R = r(x) \,\} = \Phi(r(y)|r(x))
\end{aligned} \tag{46}$$

http://dx.doi.org/10.5772/51800

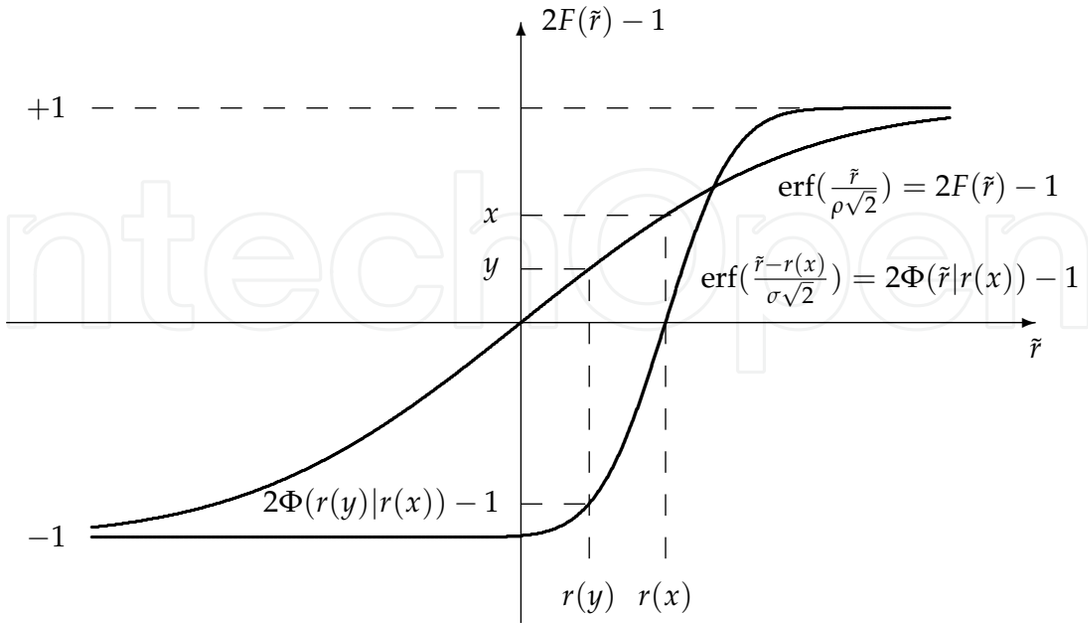

Figure 6. Illustration of the computation of the probability $\Phi(r(y)|r(x))$ for a given pair $(x,y)\in \mathsf{F}\times\mathsf{F}$, when $F=F\mathrm{G}_{0,1}$.

and

$$\begin{aligned} v(y|x) &= \frac{d}{dy}\Pr_{\text{noise}}\{\,Y<y\,|\,X=x\,\}=\frac{d}{dy}\Phi(r(y)|r(x)) \\ &= \frac{d}{d\tilde{r}}\Phi(\tilde{r}|r(x))\Big|_{\tilde{r}=r(y)}\cdot\frac{d}{dy}r(y)=V(r(y)|r(x))\cdot\frac{d}{dy}F^{-1}\Big(\frac{y+1}{2}\Big) \\ &= V(r(y)|r(x))\cdot\Big(\frac{d}{d\tilde{r}}F(\tilde{r})\Big|_{\tilde{r}=r(y)}\Big)^{-1}\cdot\frac{d}{dy}\frac{y+1}{2}=\frac{V(r(y)|r(x))}{2P(r(y))} \end{aligned} \tag{47}$$

Therefore the conditional PDs and the conditional PDFs are specified as

$$\varphi_x=\Big(\Phi(r(y)|r(x)),\,y\in\mathsf{F}\Big),\quad v_x=\Big(\frac{V(r(y)|r(x))}{2P(r(y))},\,y\in\mathsf{F}\Big) \tag{48}$$

for all $x\in\mathsf{F}$. By (11) and the fact that the F-transformation is a component-wise mapping, the value of the PDF, associated with the vector $\boldsymbol{y}$, given the vector $\boldsymbol{x}$, is expressed as

$$\mathrm{PDF}(\boldsymbol{y}|\boldsymbol{x})=\prod_{t=1}^{n}\frac{V(r(y_t)|r(x_t))}{2P(r(y_t))} \tag{49}$$

for all $\boldsymbol{x},\boldsymbol{y}\in\mathsf{F}^n$.

The computation of the PDs φ_x is illustrated in Figure 6 for Gaussian data, and examples of functions constructed using this procedure are given in Figures 7, 8.

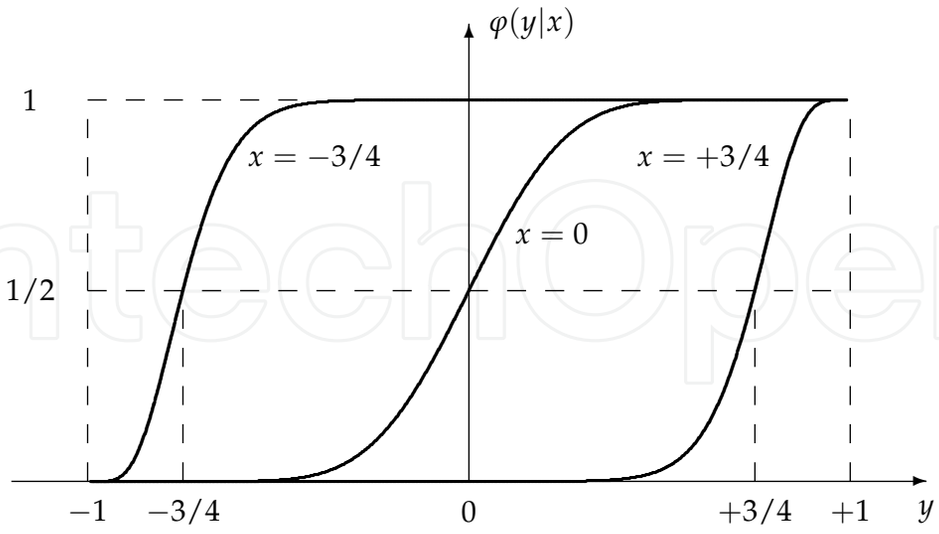

Figure 7. Examples of the conditional PDs φ_x, when $(F, \Phi_r) = (F\mathrm{G}_{0,1}, F\mathrm{G}_{r,1/4})$.

If the observation channel is an additive channel, then $V(r'|r)$ depends only on the absolute value of the difference $r' - r$. It is also usually assumed that

$$V(r'|r) \ \text{ is a monotone decreasing function of } |r' - r| \tag{50}$$

and

$$\max_{r' \in \mathsf{R}} V(r'|r) = V(r|r) = \text{Const} \tag{51}$$

In particular, $\text{Const} = 1/(\sigma\sqrt{2\pi})$ for additive white Gaussian noise channels having the variance of the noise equal to σ^2.

We include transformations at the input/output of the channel and create another channel $\boldsymbol{x} \to \boldsymbol{y}$ whose conditional PDFs essentially depend on the magnitude of the input symbols. This point is illustrated in Figures 7, 8: the slope of the function $\varphi(x|x)$ increases with $|x|$, and $v(x|x)$ tends to the δ-function, as $x \to \pm 1$. Notice that the behavior of functions under considerations is completely determined by the description of physical channel and the chosen function F, and it is not affected by the PD F^*, which can be unknown.

By (51),

$$\frac{V(r(y)|r(x))}{2P(r(y))}\Big|_{y=x} = \frac{V(r(x)|r(x))}{2P(r(x))} = \frac{\text{Const}}{2P(r(x))} \tag{52}$$

Suppose that, for any $\varepsilon > 0$, there is an $r_\varepsilon \in \mathsf{R}$ such that $|r| > r_\varepsilon$ implies $P(r) \le \varepsilon$. Then

$$|r(x)| > r_\varepsilon \ \Rightarrow \ \frac{V(r(y)|r(x))}{2P(r(y))}\Big|_{y=x} \ge \frac{\text{Const}}{2\varepsilon} \tag{53}$$

http://dx.doi.org/10.5772/51800

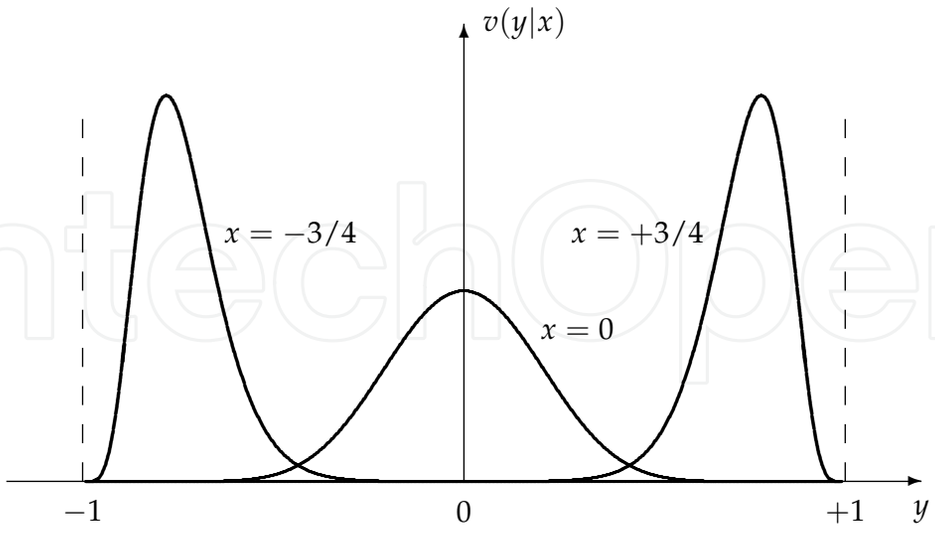

Figure 8. Examples of the conditional PDFs v_x, when $(F, \Phi_r) = (F\mathrm{G}_{0,1}, F\mathrm{G}_{r,1/4})$.

and the expression at the right-hand side tends to infinity, as $\varepsilon \to 0$, i.e., the created channel becomes noiseless. In other words, we preserve significant components of the vector $\boldsymbol{r}$ in noisy observations for a wide class of observation channels.

If $\Phi_r = F\mathrm{G}_{r,\sigma}$, then

$$\varphi(y|x) = \frac{1}{2} + \frac{1}{2}\mathrm{erf}\Big(\frac{r(y) - r(x)}{\sigma\sqrt{2}}\Big), \;\; v(y|x) = \frac{1}{2P(r(y))}\mathrm{G}_{r(x),\sigma}(r(y)) \tag{54}$$

and if $F = F\mathrm{G}_{0,\rho}$, then these equalities can be continued as

$$\varphi(y|x) = \frac{1}{2} + \frac{1}{2}\mathrm{erf}\Big(\frac{\rho}{\sigma}\Big(\,\mathrm{erf}^{-1}(y) - \mathrm{erf}^{-1}(x)\,\Big)\Big) \tag{55}$$

$$v(y|x) = \frac{\rho}{2\sigma}\exp\Big\{-\frac{\rho^2}{\sigma^2}\Big(\mathrm{erf}^{-1}(y) - \mathrm{erf}^{-1}(x)\Big)^2 + \Big(\mathrm{erf}^{-1}(y)\Big)^2\Big\} \tag{56}$$

as it follows from (23). We also write

$$(F, \Phi_r) = (F\mathrm{G}_{0,\rho}, F\mathrm{G}_{r,\sigma}) \;\Rightarrow\; \frac{\mathrm{Const}}{2P(r(x))} = \frac{\rho}{2\sigma}E(x) \tag{57}$$

where

$$E(x) = \exp\Big\{\,\frac{r^2(x)}{2\rho^2}\,\Big\} = \exp\Big\{\,\Big(\mathrm{erf}^{-1}(x)\Big)^2\,\Big\} \tag{58}$$

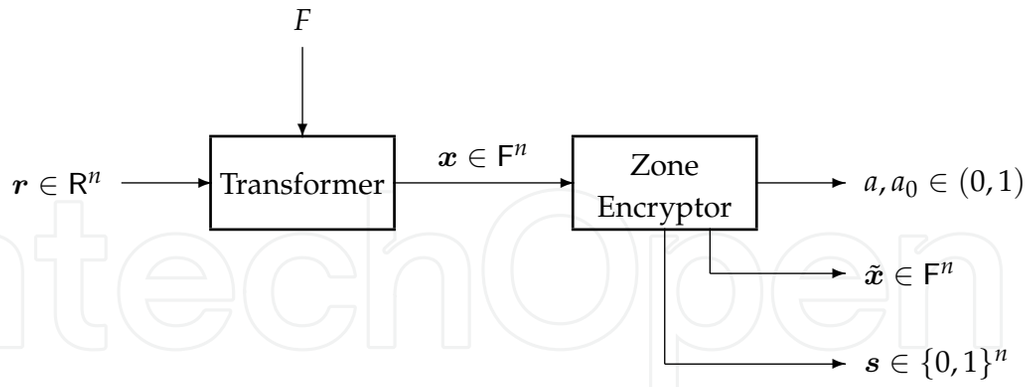

Figure 9. Processing the vector $\boldsymbol{r}$ at the enrollment stage with zone encryption.

does not depend on ρ. By (52), the expression at the right-hand side of (57) specifies the value of $v(x|x)$. The function $E(x)$ is minimized at $x = 0$ and tends to infinity, as $x \to \pm 1$ (see Table 2). Thus, the PDF v_x tends to the δ-function, as $x \to \pm 1$. The speed of convergence is proportional to the ratio $\rho/(2\sigma)$.

The derived formulas will be used in the evaluation of verification performance, but some preliminary estimates can be already presented. Namely, let us fix a $\lambda \in (0,1)$ and, for all $x \in (0,1)$, denote

$$\Delta_x(\lambda) = x - y_\lambda \tag{59}$$

where y_λ is the solution to the equation $\varphi(y_\lambda|x) = \lambda$ and the function $\varphi(y|x)$ is expressed in (55). Furthermore, let

$$\Lambda_x = \Pr_{\text{noise}}\{Y < 0 \,|\, X = x\} = \varphi(0|x) = \psi_{-\rho/\sigma}(x) \tag{60}$$

where the function $\psi(x)$ is defined in (15) for $c = -\rho/\sigma$. By the symmetry properties, Λ_x is the probability that the input symbol, having the magnitude x, changes the sign after the symbol is transmitted over the $X \to Y$ channel. The probability that $Y < x - \Delta_x(\lambda)$ is equal to λ, when $+x$ is transmitted. The probability that $Y > -x + \Delta_x(\lambda)$ is also equal to λ, when $-x$ is transmitted. The numerical illustration is included in Table 2 for $(\rho, \sigma) = (1, 1/4)$. For example, if $x = 0.90$, then $Y < 0$ with the probability $2.4 \cdot 10^{-11}$ and components of the vector $(10^{-2}, 10^{-4}, 10^{-8}, 10^{-16})$ are the probabilities of the events $Y < 0.90 - (0.19, 0.37, 0.71, 1.22) = (+0.71, +0.53, +0.19, -0.32)$. Comparison of parameters above for different x allows us to conclude that the increase of the magnitude of the transmitted symbol leads to essential improvement over the channel having the input alphabet (-1,+1) and the output alphabet {-,+}.

5. Constructing the wrapped versions of input data and verification over noisy observations

We believe that there is a large variety of verification schemes that can be constructed on the basis of statistical properties of the probabilistic ensemble (X, Y). We modify the scheme in

http://dx.doi.org/10.5772/51800

$\mathbf{x}$	$\mathbf{E(x)}$	$\mathbf{\Lambda_x}$	$\mathbf{\Delta_x(10^{-2})}$	$\mathbf{\Delta_x(10^{-4})}$	$\mathbf{\Delta_x(10^{-8})}$	$\mathbf{\Delta_x(10^{-16})}$
0	1.00	0.5	0.44	0.65	0.84	0.96
0.25	1.05	0.1	0.46	0.71	0.97	1.17
0.50	1.25	$3.5 \cdot 10^{-3}$	0.43	0.70	1.03	1.33
0.75	1.94	$2.1 \cdot 10^{-6}$	0.32	0.58	0.95	1.38
0.80	2.22	$1.5 \cdot 10^{-7}$	0.28	0.52	0.90	1.36
0.85	2.82	$4.3 \cdot 10^{-9}$	0.24	0.46	0.82	1.31
0.90	3.87	$2.4 \cdot 10^{-11}$	0.19	0.37	0.71	1.22
0.91	4.21	$5.9 \cdot 10^{-12}$	0.18	0.35	0.68	1.19
0.92	4.63	$1.3 \cdot 10^{-12}$	0.16	0.33	0.65	1.16
0.93	5.16	$2.1 \cdot 10^{-13}$	0.15	0.31	0.61	1.12
0.94	5.86	$2.7 \cdot 10^{-14}$	0.13	0.28	0.57	1.08
0.95	6.82	$2.3 \cdot 10^{-15}$	0.12	0.25	0.53	1.03
0.96	8.24	$< 10^{-15}$	0.10	0.22	0.48	0.96
0.97	10.53	$< 10^{-15}$	0.08	0.18	0.41	0.88
0.98	14.97	$< 10^{-15}$	0.06	0.14	0.34	0.77
0.99	27.59	$< 10^{-15}$	0.04	0.09	0.23	0.59

Table 2. Some values of $E(x)$, Λ_x, and $\Delta_x(\lambda)$ for $(\rho, \sigma) = (1, 1/4)$.

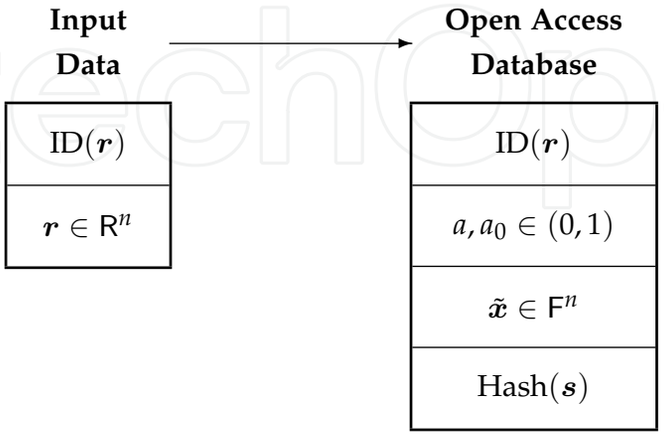

Figure 10. Mapping of the input data to the data stored in an open access database.

Figure 1a and introduce the so–called zone encryption (see Figure 9),

$$\boldsymbol{x} \;\rightarrow\; \Big((a, a_0), \tilde{\boldsymbol{x}}, \boldsymbol{s} \Big) \tag{61}$$

where $a, a_0 \in (0,1)$, $a > a_0$; $\tilde{\boldsymbol{x}} = (\tilde{x}_1, \ldots, \tilde{x}_n) \in \mathsf{F}^n$; $\boldsymbol{s} = (s_1, \ldots, s_n) \in \{0,1\}^n$. If $\mathrm{ID}(\boldsymbol{r})$ is the identifier of the person whose data are given by the vector $\boldsymbol{r}$, then the enrollment is represented as the mapping

$$\Big(\mathrm{ID}(\boldsymbol{r}), \boldsymbol{r} \Big) \;\rightarrow\; \Big(\mathrm{ID}(\boldsymbol{r}), (a, a_0), \tilde{\boldsymbol{x}}, \mathrm{Hash}(\boldsymbol{s}) \Big) \tag{62}$$

where the function Hash is a cryptographic "one–way" function having the property that one can easily find the value of the function for the given argument, but the inversion (finding the argument for the known value of the function) is practically impossible. The parameters at the right-hand side are stored in an open access database (see Figure 10).

The use of "the zone encryption", is caused by the point that we partition the $(-1, +1)$ interval into 5 zones: Significant$^{+/-}$ zones $(\mathrm{Sg}^+, \mathrm{Sg}^-)$, Buffer $^{+/-}$ zones $(\mathrm{Bf}^+, \mathrm{Bf}^-)$, Zero zone (Zr), where

$$\begin{cases} \mathrm{Sg}^+ = (+a, +1), \mathrm{Sg}^- = (-1, -a) \\ \mathrm{Bf}^+ = (+a_0, +a), \mathrm{Bf}^- = (-a, -a_0) \\ \mathrm{Zr} = (-a_0, +a_0) \end{cases} \tag{63}$$

Let the notation $\tilde{x}_t \sim \mathrm{U}(+a, +1)$ be understood in such a way that the value of $\tilde{x}_t$ is chosen at random using a uniform PD over the $(+a, +1)$ interval. Similarly, if $\tilde{x}_t \sim \mathrm{U}(-1, -a)$, then the value of $\tilde{x}_t$ is chosen at random using a uniform PD over the $(-1, -a)$ interval. If $\boldsymbol{x}$ is the wrapped version of the vector $\boldsymbol{r}$, constructed by the F-transformation, then we set

$$\begin{cases} \tilde{x}_t = x_t, & \text{if } x_t \in \mathrm{Sg}^+ \cup \mathrm{Sg}^- \\ \tilde{x}_t \sim \mathrm{U}(-1, -a), & \text{if } x_t \in \mathrm{Bf}^+ \\ \tilde{x}_t \sim \mathrm{U}(+a, +1), & \text{if } x_t \in \mathrm{Bf}^- \\ \tilde{x}_t = 0, & \text{if } x_t \in \mathrm{Zr} \end{cases} \tag{64}$$

Therefore components of the vector $\boldsymbol{x}$, belonging to the Significant zones, are unchanged and components, belonging to the Zero zone, are set to zero. Components, belonging to the Buffer zones, are changed in such a way that the results belong to the Significant zones with different signs. The presented procedure is illustrated in Figure 11, and the binary vector $\boldsymbol{s}$ having components

$$s_t = \begin{cases} 1, \text{ if } x_t \in \mathrm{Sg}^+ \cup \mathrm{Sg}^- \\ 0, \text{ if } x_t \notin \mathrm{Sg}^+ \cup \mathrm{Sg}^- \end{cases} \tag{65}$$

specifies the Significant zones.

http://dx.doi.org/10.5772/51800

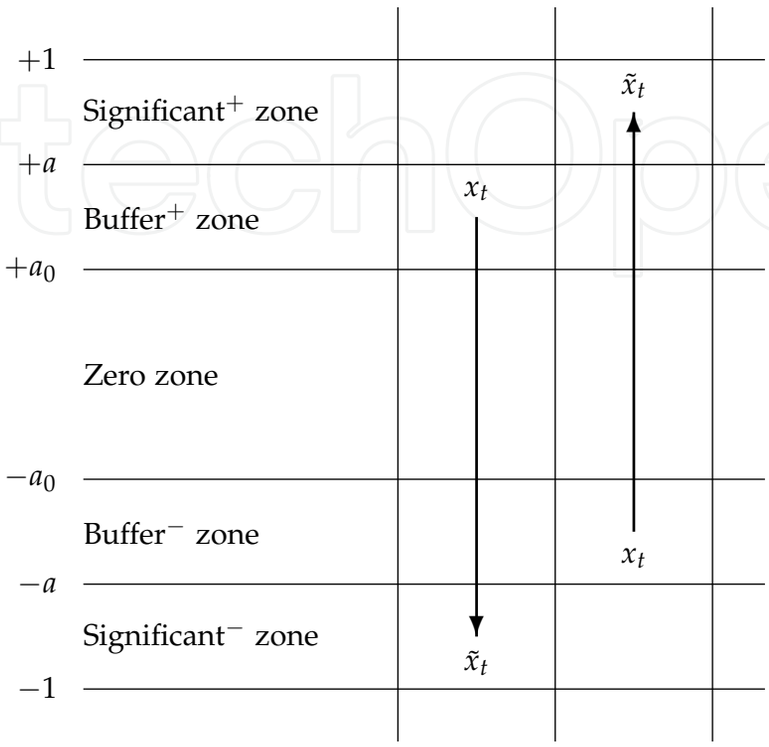

Figure 11. Partitioning of the $(-1,+1)$ interval into zones and illustration of the mapping $x_t \to \tilde{x}_t$, when x_t belongs to the Buffer zones.

The zone encryption can be introduced as representation of the vector $\boldsymbol{x}$ by the sum of three vectors,

$$\boldsymbol{x} = \boldsymbol{x}^{(\mathrm{Sg})} + \boldsymbol{x}^{(\mathrm{Bf})} + \boldsymbol{x}^{(\mathrm{Zr})} \tag{66}$$

constructed as follows: set $x_t^{(\mathrm{Sg})} = x_t^{(\mathrm{Bf})} = x_t^{(\mathrm{Zr})} = 0$ for all $t = 1, \ldots, n$ and use the rules

$$\begin{cases} x_t^{(\mathrm{Sg})} = x_t, \text{ if } x_t \in \mathrm{Sg}^+ \cup \mathrm{Sg}^- \\ x_t^{(\mathrm{Bf})} = x_t, \text{ if } x_t \in \mathrm{Bf}^+ \cup \mathrm{Bf}^- \\ x_t^{(\mathrm{Zr})} = x_t, \text{ if } x_t \in \mathrm{Zr} \end{cases} \tag{67}$$

The vector $\tilde{\boldsymbol{x}}$ can be also represented by the sum of three vectors,

$$\tilde{\boldsymbol{x}} = \tilde{\boldsymbol{x}}^{(\mathrm{Sg})} + \tilde{\boldsymbol{x}}^{(\mathrm{Bf})} + \tilde{\boldsymbol{x}}^{(\mathrm{Zr})} \tag{68}$$

	$(\mathbf{a}, \mathbf{a_0}) = (.88, .76)$							
$\boldsymbol{x}$	$+.68$	$-.77$	$+.91$	$+.24$	$-.98$	$+.08$	$-.52$	$-.81$
$\boldsymbol{x}^{(\mathrm{Sg})}$			$+.91$		$-.98$			
$\boldsymbol{x}^{(\mathrm{Bf})}$		$-.77$						$-.81$
$\boldsymbol{x}^{(\mathrm{Zr})}$	$+.68$			$+.24$		$+.08$	$-.52$	
$\tilde{\boldsymbol{x}}^{(\mathrm{Sg})}$			$+.91$		$-.98$			
$\tilde{\boldsymbol{x}}^{(\mathrm{Bf})}$		$+.90$						$+.95$
$\tilde{\boldsymbol{x}}^{(\mathrm{Zr})}$	0			0		0	0	
$\tilde{\boldsymbol{x}}$	0	$+.90$	$+.91$	0	$-.98$	0	0	$+.95$
$\boldsymbol{s}$	0	0	1	0	1	0	0	0

Table 3. Example of the zone encryption where gaps contain zeroes.

where we first set $\tilde{x}_t^{(\mathrm{Sg})} = \tilde{x}_t^{(\mathrm{Bf})} = \tilde{x}_t^{(\mathrm{Zr})} = 0$ for all $t = 1, \ldots, n$ and then set

$$\begin{cases} \tilde{x}_t^{(\mathrm{Sg})} = x_t^{(\mathrm{Sg})}, & \text{if } x_t \in \mathrm{Sg}^+ \cup \mathrm{Sg}^- \\ \tilde{x}_t^{(\mathrm{Bf})} \sim \mathrm{U}(-1, -a), & \text{if } x_t \in \mathrm{Bf}^+ \\ \tilde{x}_t^{(\mathrm{Bf})} \sim \mathrm{U}(+a, +1), & \text{if } x_t \in \mathrm{Bf}^- \\ \tilde{x}_t^{(\mathrm{Zr})} = 0, & \text{if } x_t \in \mathrm{Zr} \end{cases} \tag{69}$$

Thus, the t-th component of the vector $\tilde{\boldsymbol{x}}$ is either equal to 0, or belongs to the set $\mathrm{Sg}^+ \cup \mathrm{Sg}^-$. Furthermore, $\tilde{x}_t > 0$ implies $\tilde{x}_t \in \mathrm{Sg}^+$, and $\tilde{x}_t < 0$ implies $\tilde{x}_t \in \mathrm{Sg}^-$. Thus, $n = n^{(\mathrm{Sg})}(\tilde{\boldsymbol{x}}) + n^{(\mathrm{Zr})}(\tilde{\boldsymbol{x}})$, where

$$n^{(\mathrm{Sg})}(\tilde{\boldsymbol{x}}) = \Big\{ t \in \{1, \ldots, n\} : \tilde{x}_t \in \mathrm{Sg}^+ \cup \mathrm{Sg}^- \Big\} \tag{70}$$

$$n^{(\mathrm{Zr})}(\tilde{\boldsymbol{x}}) = \Big\{ t \in \{1, \ldots, n\} : \tilde{x}_t = 0 \Big\} \tag{71}$$

Moreover, $n^{(\mathrm{Sg})}(\tilde{\boldsymbol{x}}) = n^{(\mathrm{Sg})}(\boldsymbol{x}) + n^{(\mathrm{Bf})}(\boldsymbol{x})$, where $n^{(\mathrm{Sg})}(\boldsymbol{x}) = |\mathcal{T}^{(\mathrm{Sg})}(\boldsymbol{x})|$, $n^{(\mathrm{Bf})}(\boldsymbol{x}) = |\mathcal{T}^{(\mathrm{Bf})}(\boldsymbol{x})|$, and

$$\mathcal{T}^{(\mathrm{Sg})}(\boldsymbol{x}) = \Big\{ t \in \{1, \ldots, n\} : x_t \in \mathrm{Sg}^+ \cup \mathrm{Sg}^- \Big\} \tag{72}$$

$$\mathcal{T}^{(\mathrm{Bf})}(\boldsymbol{x}) = \Big\{ t \in \{1, \ldots, n\} : x_t \in \mathrm{Bf}^+ \cup \mathrm{Bf}^- \Big\} \tag{73}$$

The numerical example of the zone encryption is given in Table 3. Since $|-.77|, |-.81| \in (.76, .88) = (a_0, a)$, the 2-nd and the 8-th components of the vector $\boldsymbol{x}$ belong to the Buffer

http://dx.doi.org/10.5772/51800

zones. The encryptor replaces these components by numbers chosen at random from the $(+.88, +1)$ interval. For example, $-.77 \to +.90$ and $-.81 \to +.95$. The vector

$$\tilde{\boldsymbol{x}} = (0, +.90, +.91, 0, -.98, 0, 0, +.95) \tag{74}$$

and the hidden version of the vector $\boldsymbol{s} = (0, 0, 1, 0, 1, 0, 0, 0)$ are stored in the database.

Similarly to (63), let us introduce the sets

$$\begin{cases} \mathrm{rSg}^{+} = (r(+a), r(+1)),\ \mathrm{rSg}^{-} = (r(-1), r(-a)) \\ \mathrm{rBf}^{+} = (r(+a_0), r(+a)),\ \mathrm{rBf}^{-} = (r(-a), r(-a_0)) \\ \mathrm{rZr} = (r(-a_0), r(+a_0)) \end{cases} \tag{75}$$

where the function $r(x), x \in \mathsf{F}$, is defined in (22). It is important to notice that the same vector $\tilde{\boldsymbol{x}}$ encrypts many vectors $\boldsymbol{r}$ and

$$\tilde{x}_t \in \mathrm{Sg}^{+} \Rightarrow r_t \in \mathrm{rSg}^{+} \cup \mathrm{rBf}^{-},\ \tilde{x}_t \in \mathrm{Sg}^{-} \Rightarrow r_t \in \mathrm{rSg}^{-} \cup \mathrm{rBf}^{+},\ \tilde{x}_t = 0 \Rightarrow r_t \in \mathrm{rZr} \tag{76}$$

If $\tilde{x}_t \in \mathrm{Sg}^{+}$, then the conclusion, whether r_t belongs to the sets rSg^{+} or rBf^{-}, is specified by the t-th component of the vector $\boldsymbol{s}$, which is hidden from the attacker. Similarly, if $\tilde{x}_t \in \mathrm{Sg}^{-}$, then the conclusion, whether r_t belongs to the sets rSg^{-} or rBf^{+}, is also specified by the t-th component of the vector $\boldsymbol{s}$. The total number of variants is equal to $2^{n^{(\mathrm{Sg})}(\tilde{\boldsymbol{x}})}$. All of them are presented in Table 4 for the vector $\tilde{\boldsymbol{x}}$, defined in (74). For example, if $F = F\mathrm{G}_{0,1}$, then $(r(.88), r(.76)) = (1.55, 1.17)$ and $r_2, r_3, r_8 \in (+1.55, +\infty) \cup (-1.55, -1.17)$, $r_5 \in (-\infty, -1.55) \cup (+1.17, +1.55)$, $r_1, r_4, r_6, r_7 \in (-1.17, +1.17)$.

If we construct a uniform probability distribution at the enrolment stage by using the F-transformation, then all variants are equivalent, and the probability of the correct guess of the vector $\boldsymbol{s}$ by an attacker, who knows the vector $\tilde{\boldsymbol{x}}$, is equal to

$$P_{\mathrm{att}}(\tilde{\boldsymbol{x}}) = 2^{-n^{(\mathrm{Sg})}(\tilde{\boldsymbol{x}})} = 2^{-(n^{(\mathrm{Sg})}(\boldsymbol{x}) + n^{(\mathrm{Bf})}(\boldsymbol{x}))} \tag{77}$$

and the value of the sum $n^{(\mathrm{Sg})}(\boldsymbol{x}) + n^{(\mathrm{Bf})}(\boldsymbol{x})$ is a function of the vector $\boldsymbol{x}$ and the pair (a, a_0).

Notice that the randomization at the enrollment stage can be replaced by deterministic mappings $\mathrm{Bf}^{+} \to \mathrm{Sg}^{-}$ and $\mathrm{Bf}^{-} \to \mathrm{Sg}^{+}$. For example, if the Significant and the Buffer zones have equal sizes, i.e., if $1 - a = a - a_0$, then one can follow the rules: if $x_t \in \mathrm{Bf}^{+}$, then $\tilde{x}_t = -a - (x_t - a_0)$; if $x_t \in \mathrm{Bf}^{-}$, then $\tilde{x}_t = +a + (a_0 - x_t)$.

Let us introduce the decoding as the mapping

$$\left(\boldsymbol{y}, (a, a_0), \tilde{\boldsymbol{x}} \right) \ \to\ \hat{\boldsymbol{s}} \in \{0, 1\}^n \tag{78}$$

				$(\mathbf{a}, \mathbf{a_0}) = (\mathbf{.88}, \mathbf{.76})$			
s_2	s_3	s_5	s_8	$\tilde{x}_2 = +.90$	$\tilde{x}_3 = +.91$	$\tilde{x}_5 = -.98$	$\tilde{x}_8 = +.95$
0	0	0	0	$r_2 \in \mathrm{rBf}^-$	$r_3 \in \mathrm{rBf}^-$	$r_5 \in \mathrm{rBf}^+$	$r_8 \in \mathrm{rBf}^-$
0	0	0	1	$r_2 \in \mathrm{rBf}^-$	$r_3 \in \mathrm{rBf}^-$	$r_5 \in \mathrm{rBf}^+$	$r_8 \in \mathrm{rSg}^+$
0	0	1	0	$r_2 \in \mathrm{rBf}^-$	$r_3 \in \mathrm{rBf}^-$	$r_5 \in \mathrm{rSg}^-$	$r_8 \in \mathrm{rBf}^-$
0	0	1	1	$r_2 \in \mathrm{rBf}^-$	$r_3 \in \mathrm{rBf}^-$	$r_5 \in \mathrm{rSg}^-$	$r_8 \in \mathrm{rSg}^+$
0	1	0	0	$r_2 \in \mathrm{rBf}^-$	$r_3 \in \mathrm{rSg}^+$	$r_5 \in \mathrm{rBf}^+$	$r_8 \in \mathrm{rBf}^-$
0	1	0	1	$r_2 \in \mathrm{rBf}^-$	$r_3 \in \mathrm{rSg}^+$	$r_5 \in \mathrm{rBf}^+$	$r_8 \in \mathrm{rSg}^+$
0	1	1	0	$r_2 \in \mathrm{rBf}^-$	$r_3 \in \mathrm{rSg}^+$	$r_5 \in \mathrm{rSg}^-$	$r_8 \in \mathrm{rBf}^-$
0	1	1	1	$r_2 \in \mathrm{rBf}^-$	$r_3 \in \mathrm{rSg}^+$	$r_5 \in \mathrm{rSg}^-$	$r_8 \in \mathrm{rSg}^+$
1	0	0	0	$r_2 \in \mathrm{rSg}^+$	$r_3 \in \mathrm{rBf}^-$	$r_5 \in \mathrm{rBf}^+$	$r_8 \in \mathrm{rBf}^-$
1	0	0	1	$r_2 \in \mathrm{rSg}^+$	$r_3 \in \mathrm{rBf}^-$	$r_5 \in \mathrm{rBf}^+$	$r_8 \in \mathrm{rSg}^+$
1	0	1	0	$r_2 \in \mathrm{rSg}^+$	$r_3 \in \mathrm{rBf}^-$	$r_5 \in \mathrm{rSg}^-$	$r_8 \in \mathrm{rBf}^-$
1	0	1	1	$r_2 \in \mathrm{rSg}^+$	$r_3 \in \mathrm{rBf}^-$	$r_5 \in \mathrm{rSg}^-$	$r_8 \in \mathrm{rSg}^+$
1	1	0	0	$r_2 \in \mathrm{rSg}^+$	$r_3 \in \mathrm{rSg}^+$	$r_5 \in \mathrm{rBf}^+$	$r_8 \in \mathrm{rBf}^-$
1	1	0	1	$r_2 \in \mathrm{rSg}^+$	$r_3 \in \mathrm{rSg}^+$	$r_5 \in \mathrm{rBf}^+$	$r_8 \in \mathrm{rSg}^+$
1	1	1	0	$r_2 \in \mathrm{rSg}^+$	$r_3 \in \mathrm{rSg}^+$	$r_5 \in \mathrm{rSg}^-$	$r_8 \in \mathrm{rBf}^-$
1	1	1	1	$r_2 \in \mathrm{rSg}^+$	$r_3 \in \mathrm{rSg}^+$	$r_5 \in \mathrm{rSg}^-$	$r_8 \in \mathrm{rSg}^+$
					$\tilde{x}_1 = \tilde{x}_4 = \tilde{x}_6 = \tilde{x}_7 = 0$		
					$r_1, r_4, r_6, r_7 \in \mathrm{rZr}$		

Table 4. The list of vectors $\boldsymbol{r}$ that are encrypted by the vector $\tilde{\boldsymbol{x}}$, defined in (74).

Let the verifier set $\hat{s}_1 = \cdots = \hat{s}_n = 0$ and, for all $t = 1, \ldots, n$, use the rule:

$$\left.\begin{aligned} \tilde{x}_t &\neq 0 \\ |y_t| &> T \\ \mathrm{sgn}(\tilde{x}_t) &= \mathrm{sgn}(y_t) \end{aligned}\right\} \Rightarrow \hat{s}_t = 1 \tag{79}$$

where the value of the threshold T is a function of (a, a_0), which will be specified later. The verifier can then check whether $\mathrm{Hash}(\hat{\boldsymbol{s}})$ is equal to the value of $\mathrm{Hash}(\boldsymbol{s})$, stored in the database, or not. If the answer is positive, then the acceptance decision is made. If the answer is negative, then the rejection decision is made.

Without loss of generality, let us suppose that

http://dx.doi.org/10.5772/51800

$$x_1, \ldots, x_n \in \mathrm{Sg}^+ \cup \mathrm{Bf}^+ \cup \mathrm{Zr} \tag{80}$$

The decoding error in the case, when the vector $\boldsymbol{r}'$ is a noisy version of the vector $\boldsymbol{r}$, occurs in one of two situations (see Figure 12).

E_{10}: $(s_t, \hat{s}_t) = (1, 0)$. Then $x_t \in (+a, +1)$, as it follows from (80) and $s_t = 1$. Furthermore, $y_t < +T$, since $\hat{s}_t = 0$ implies that the conditions at the left-hand side of (79) are not satisfied. Hence,

$$x_t - y_t \geq (+a) - (+T) \tag{81}$$

E_{01}: $(s_t, \hat{s}_t) = (0, 1)$. Then $x_t \in (+a_0, +a)$, as it follows from (80) and $s_t = 0$, $\hat{s}_t \neq 0$. Furthermore, $y_t < -T$, since $\hat{s}_t = 1$ implies that the conditions at the left-hand side of (79) are satisfied. Hence,

$$x_t - y_t \geq (+a_0) - (-T) \tag{82}$$

If the channel $X_t \to Y_t$ would be an additive channel, then its probabilistic description does not depend on the input to the channel, and the differences at the left-hand sides of (81), (82) specify the magnitudes of the noise. The differences at the right-hand sides give lower bounds on these magnitudes. If $T = (a - a_0)/2$, then these differences are equal. However, as the created channel is not an additive channel, we will use another assignment of T.

The decoding error probability, denoted by $\Lambda(T|\boldsymbol{x})$, can be bounded from above as

$$\Lambda(T|\boldsymbol{x}) \leq \Lambda_{10}(T|\boldsymbol{x}) + \Lambda_{01}(T|\boldsymbol{x}) \tag{83}$$

where

$$\begin{aligned}
\Lambda_{10}(T|\boldsymbol{x}) &= \Pr_{\text{noise}} \{Y_t < +T, \text{ for some } t \in \mathcal{T}^{(\mathrm{Sg})}(\boldsymbol{x}) \mid X^n = \boldsymbol{x}\} \\
&= 1 - \Pr_{\text{noise}} \{Y_t \geq +T, \text{ for all } t \in \mathcal{T}^{(\mathrm{Sg})}(\boldsymbol{x}) \mid X^n = \boldsymbol{x}\} \\
&= 1 - \prod_{t \in \mathcal{T}^{(\mathrm{Sg})}(\boldsymbol{x})} \Pr_{\text{noise}} \{Y_t \geq +T \mid X_t = x_t\} \\
&\leq 1 - \prod_{t \in \mathcal{T}^{(\mathrm{Sg})}(\boldsymbol{x})} \Pr_{\text{noise}} \{Y_t \geq +T \mid X_t = +a\} \\
&= 1 - \Big(1 - \varphi(+T \mid +a)\Big)^{n^{(\mathrm{Sg})}(\boldsymbol{x})}
\end{aligned} \tag{84}$$

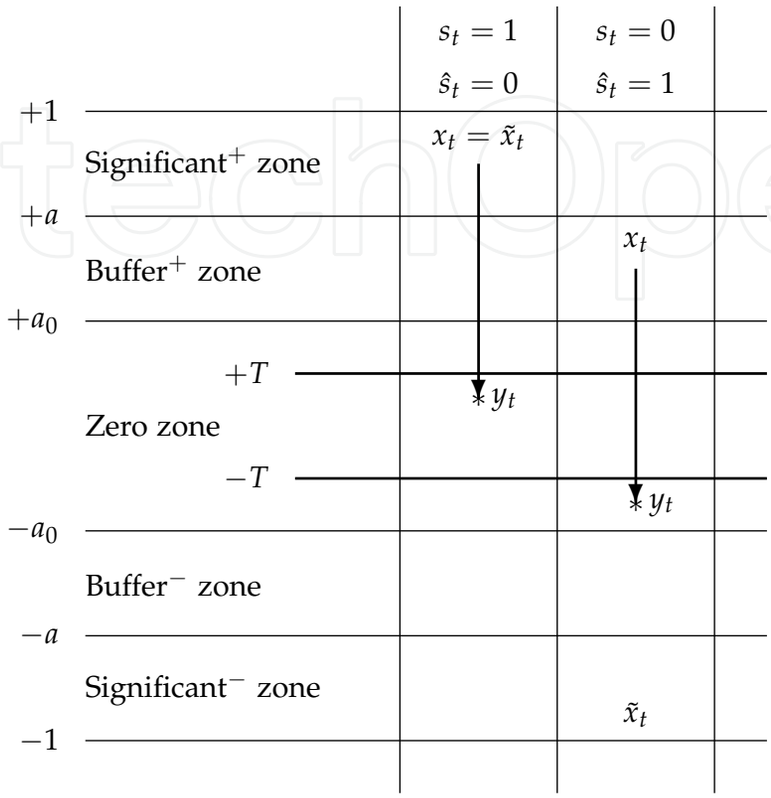

Figure 12. Illustration of the events, when the decoding error occurs and $x_t \in \mathrm{Sg}^+ \cup \mathrm{Bf}^+$.

and

$$\begin{aligned}
\Lambda_{01}(T|\boldsymbol{x}) &= \Pr_{\text{noise}}\{Y_t < -T, \text{ for some } t \in \mathcal{T}^{(\mathrm{Bf})}(\boldsymbol{x}) \,|\, X^n = \boldsymbol{x}\} \\
&= 1 - \Pr_{\text{noise}}\{Y_t \geq -T, \text{ for all } t \in \mathcal{T}^{(\mathrm{Bf})}(\boldsymbol{x}) \,|\, X^n = \boldsymbol{x}\} \\
&= 1 - \prod_{t \in \mathcal{T}^{(\mathrm{Bf})}(\boldsymbol{x})} \Pr_{\text{noise}}\{Y_t \geq -T \,|\, X_t = x_t\} \\
&\leq 1 - \prod_{t \in \mathcal{T}^{(\mathrm{Bf})}(\boldsymbol{x})} \Pr_{\text{noise}}\{Y_t \geq -T \,|\, X_t = +a_0\} \\
&= 1 - \Big(1 - \varphi(-T \,|\, +a_0)\Big)^{n^{(\mathrm{Bf})}(\boldsymbol{x})}
\end{aligned} \tag{85}$$

http://dx.doi.org/10.5772/51800

The products on t at the right-hand sides are written because the observation channel is memoryless and the inequalities follow from the assumption (50). Notice that, by (55),

$$\varphi(+T|+a) = \frac{1}{2} + \frac{1}{2}\mathrm{erf}\Big(\frac{\rho}{\sigma}\Big(\,\mathrm{erf}^{-1}(T) - \mathrm{erf}^{-1}(a)\,\Big)\Big) \tag{86}$$

$$\varphi(-T|+a_0) = \frac{1}{2} - \frac{1}{2}\mathrm{erf}\Big(\frac{\rho}{\sigma}\Big(\,\mathrm{erf}^{-1}(T) + \mathrm{erf}^{-1}(a_0)\,\Big)\Big) \tag{87}$$

and assign T in such a way that

$$\Lambda_{10}(T|\boldsymbol{x}) = \Lambda_{01}(T|\boldsymbol{x}) \tag{88}$$

Suppose that there is a blind attacker, who does not have access to the database and wants to pass through the verification stage with the acceptance decision. We also assume that the attacker can fix a small $\delta > 0$ in such a way that $1 - \delta > a$. Let the attacker submit a vector $\boldsymbol{r}'$, which is mapped to the vector $\boldsymbol{y} \in \{-1+\delta, +1-\delta\}^n$, i.e., the t-th component of the vector $\boldsymbol{y}$ is either equal to $-1+\delta$ or $+1-\delta$. Let the decision, whether the t-th component of the vector $\boldsymbol{r}'$ is equal to $r(-1+\delta)$ or $r(+1-\delta)$, be made with probabilities $1/2$. The probability of the acceptance decision is equal to $2^{-n^{(\mathrm{Sg})}(\tilde{\boldsymbol{x}})}$, since the verifier "punctures" $n - n^{(\mathrm{Sg})}(\tilde{\boldsymbol{x}})$ components $\tilde{x}_t$ equal to 0 and sets the t-th component of the binary vector equal to 1 if and only if $\mathrm{sgn}(\tilde{x}_t) = \mathrm{sgn}(y_t)$. By (77), the obtained probability is exactly the same as the probability of success of an attacker, who knows the vector $\tilde{\boldsymbol{x}}$. Thus, we attain the property of a perfect algorithmic secrecy of the designed verification scheme: although the database is open and an attacker may know the vector $\tilde{\boldsymbol{x}}$, he cannot include this information into the guessing strategy.

Suppose that the vector $\boldsymbol{x} \in \mathsf{F}^n$ is generated by a memoryless source according to a uniform probability distribution over the set F. One can see that the probability that there are $i, i_0, n-i-i_0$ components of the vector $\boldsymbol{x}$ whose magnitudes belong to the $(a,1), (a_0,a), (0,a)$ intervals, respectively, is equal to

$$Q_{a,a_0}(i,i_0) = \frac{n!}{i!i_0!(n-i-i_0)!}(1-a)^i(a-a_0)^{i_0}(1-a-a_0)^{n-i-i_0} \tag{89}$$

and

$$\arg\max_{(i,i_0)\in\{0,\ldots,n\},\, i+i_0\le n} Q_{a,a_0}(i,i_0) = \Big(\,(1-a)n,\,(a-a_0)n\,\Big) \tag{90}$$

Hence, the typical distribution of magnitudes under considerations (see Figure 13) is specified as

$$(a,a_0) = \Big(\frac{n-w}{n}, \frac{n-w_0}{n}\Big) \;\Rightarrow\; \Big(\,(1-a)n,\,(a-a_0)n\,\Big) = (\,w,\,w_0-w) \tag{91}$$

where $w, w_0 \in \{0,\ldots,n\}$ are integers, fixed in such a way that $w \le w_0$.

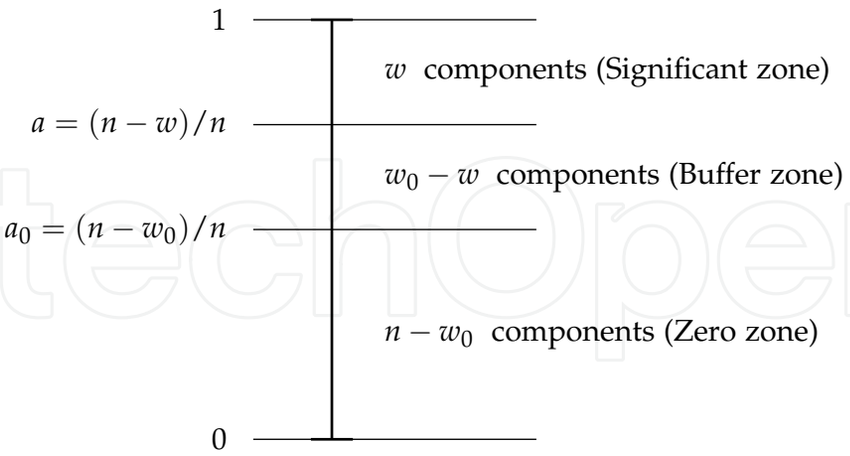

Figure 13. The typical partitioning of the values of magnitudes in three zones.

Let $(w, w_0) = (32, 64)$. Compute (a, a_0) using the expressions at the left-hand side of (91), and consider the typical case when the distribution of magnitudes of components, belonging to the Significant and the Buffer zones, is given by the right-hand side of (91), i.e., there are 32 magnitudes belonging to the Significant zones, $64 - 32 = 32$ magnitudes belonging to the Buffer zones, and $n - 64$ magnitudes belonging to the Zero zone. Thus, the vectors $\boldsymbol{x}$ under consideration are such that $n^{(\mathrm{Sg})}(\boldsymbol{x}) = n^{(\mathrm{Bf})}(\boldsymbol{x}) = 32$. Then (88) is equivalent to the equality $\varphi(+T| + a) = \varphi(-T| + a_0)$. One can see that the equality above implies the assignment

$$\left.\begin{array}{r} (F, \Phi_r) = (F\mathrm{G}_{0,\rho}, F\mathrm{G}_{r,\sigma}) \\ n^{(\mathrm{Sg})}(\boldsymbol{x}) = n^{(\mathrm{Bf})}(\boldsymbol{x}) \end{array}\right\} \;\Rightarrow\; T = \mathrm{erf}\Big(\frac{\mathrm{erf}^{-1}(a) - \mathrm{erf}^{-1}(a_0)}{2}\Big) \tag{92}$$

independently of ρ and σ. The corresponding vectors $\tilde{\boldsymbol{x}}$ contain $32 + 32 = 64$ non–zero components, and the secrecy of the verification scheme, evaluated by the probability of correct guess of the vector $\boldsymbol{s}$ on the basis of the vector $\tilde{\boldsymbol{x}}$ (see (77)), is equal to 2^{-64}. Notice that if the attacker would know that $n^{(\mathrm{Sg})}(\boldsymbol{x}) = n^{(\mathrm{Bf})}(\boldsymbol{x}) = 32$, then this probability is equal to $\binom{64}{32}^{-1}$. Some numerical results are included in Table 5, and one can see that the decoding error probability is very small even for noisy channels and relatively small lengths.

The presented version of the verification algorithm uses only the signs of components of the vector $\tilde{x}$, and this vector can be transformed further to a ternary vector whose contain either 0 or the sign. An improvement of the performance is attained by using the maximum likelihood decoding rule, when we construct the vector $\hat{\boldsymbol{s}}$ in such a way that

$$\hat{s}_t = \begin{cases} 1, \text{ if } \tilde{x}_t \neq 0,\ v(y_t|\tilde{x}_t) \geq v_{a,a_0}(y_t|\mathrm{sgn}(\tilde{x}_t)) \\ 0, \text{ if } \tilde{x}_t = 0 \text{ or } \tilde{x}_t \neq 0,\ v(y_t|\tilde{x}_t) < v_{a,a_0}(y_t|\mathrm{sgn}(\tilde{x}_t)) \end{cases} \tag{93}$$

http://dx.doi.org/10.5772/51800

				σ				
n	a	a_0	T	$1/2$	$1/3$	$1/4$	$1/5$	$1/6$
128	.750	.500	.188	> 1	$1.9 \cdot 10^{-1}$	$8.4 \cdot 10^{-3}$	$1.6 \cdot 10^{-4}$	$1.4 \cdot 10^{-6}$
256	.875	.750	.152	$2.2 \cdot 10^{-1}$	$1.8 \cdot 10^{-3}$	$2.5 \cdot 10^{-6}$	$6.2 \cdot 10^{-10}$	$2.8 \cdot 10^{-14}$
512	.938	.875	.131	$2.2 \cdot 10^{-2}$	$1.1 \cdot 10^{-5}$	$3.5 \cdot 10^{-10}$	$< 10^{-15}$	$< 10^{-15}$
1024	.969	.938	.116	$1.9 \cdot 10^{-3}$	$5.4 \cdot 10^{-8}$	$2.8 \cdot 10^{-14}$	$< 10^{-15}$	$< 10^{-15}$

Table 5. Values of the upper bound on the decoding error probability, when $(w, w_0) = (32, 64)$, (a, a_0) are expressed by the left-hand side of (91), and $(F, \Phi_r) = (F\mathrm{G}_{0,1}, F\mathrm{G}_{r,\sigma})$.

where

$$(v_{a,a_0}(y_t|1), v_{a,a_0}(y_t|0)) = \Big(\frac{1}{a - a_0} \int_{-a}^{-a_0} v(y_t|x)\, dx,\ \frac{1}{a - a_0} \int_{+a_0}^{+a} v(y_t|x)\, dx \Big) \tag{94}$$

6. Conclusion

We believe that there is a request for general theory of processing biometric data, caused by a large variety of parameters that can be taken into account and their different descriptions. It is usually difficult to find a probabilistic models for biometric observations received both at the enrollment and the verification stages that agree with practical situations. One of the most important features is privacy protection, which makes the reconstruction of outcomes of biometric measurements on the basis of the corresponding record record, stored in the database difficult. Another part of privacy protection should make the generation of artificial outcomes of the measurements that allow an attacker to pass through the verification with the acceptance decision difficult. The algorithms presented above can be considered as candidates for the inclusion into such a theory. We introduce the F-transformation of input data, where F specifies some probability distribution. As a result, the data are mapped into the $(-1, +1)$ interval and, if the actual probability distribution is memoryless and it coincides with the multiplicative extension of F, then we attain a uniform probability distribution over the $(-1, +1)$ interval. Otherwise, a uniform distribution can be approached using a generalized version of the F-transformation. The use of the proposed technique for noisy observations at the verification stage leads to an interesting effect that rare outcomes over the artificial ensemble, defined by the probability distribution F and the corresponding probability density function P, are reliably transmitted over any additive observation channel, since $P(r(y))$ appears in the denominator of the constructed probability density function (see 48). This property allows us to transmit large magnitudes over the constructed $X \to Y$ channel with high reliability. Notice that this claim is not affected by the match of the actual probability distribution of input data and the introduced probability distribution F.

The points above can be translated to different verification strategies. Our verification algorithm can be viewed as a secret sharing scheme where the input vector $\boldsymbol{r}$ is converted to a pair of vectors $(\tilde{\boldsymbol{x}}, \boldsymbol{s})$. The vector $\tilde{\boldsymbol{x}}$ is published, while the vector $\boldsymbol{s}$ is supposed to be

decoded on the basis of the vector $\tilde{\boldsymbol{x}}$ and a noisy version of the vector $\boldsymbol{r}$. An important ingredient of the presented algorithms is the dependence of the threshold T on the pair (a, a_0). This dependence assumes that the verifier assigns this pair depending on the vector $\boldsymbol{x}$ received at the enrollment stage. Some other verification schemes are described in [20].

Author details

Vladimir B. Balakirsky[1] and A. J. Han Vinck[2]

1 Data Security Association "Confident", St-Petersburg, Russia
American University of Armenia, Yerevan, Armenia
2 Institute for Experimental Mathematics, University of Duisburg-Essen, Essen, Germany

References

[1] Bolle, R. M., Connell, J. H., Pankanti S., Ratha, N. K., & Senior A. W. (2004). *Guide to Biometrics*.

[2] Ross, A., Jain, A. K., & Zhang, D. (2006). *Handbook on Multibiometrics*.

[3] Tuyls, P., Scoric, B., & Kavenaar, T. (2007). *Security with Noisy Data: Private Biometrics, Secure Key Storage and Anti–Counterfeiting*.

[4] Bhattacharyya, D., Ranjian, R., Alisherov, F., & Choi, M. (2009). Biometric authentication: A review, *International Journal of u- and e-Service, Science and Technology*, vol. 2, no. 3, pp. 13–28.

[5] Balakirsky, V. B., Ghazaryan, A. R., & Han Vinck, A. J. (2007). Testing the independence of two non–stationary random processes with applications to biometric authentication, *Proceedings of the International Symposium on Information Theory*, France, pp. 2671–2675.

[6] Balakirsky, V. B., & Han Vinck, A. J. (2011). Biometric authentication based on significant parameters, *Lecture Notes in Computer Science: Biometrics and ID management*, vol. 6583, pp. 13–22.

[7] Voloshynovskiy, S., Koval, O., Beekhof, F., Farhadzadeh, F., & Holotyak, T. (2011). Private content identification based on soft fingerprinting, *Proceedings of SPIE Photonics West, Electronic Imaging, Media Forensics and Security XIII*. San Fransisco, U.S.A.

[8] Balakirsky, V. B., Voloshynovskiy, S., Koval, O., & Holotyak, T. (2011). Information–theoretic analysis of privacy protection for noisy identification based on soft fingerprinting, *Proceedings of International Conference "Computer Science and Information Technologies"*, Yerevan, Armenia, pp. 107–110. http://www.csit.am/2011/

[9] Kullback, S. (1968). *Information Theory and Statistics*.

[10] Ahlswede, R., & Csiszár, I. (1993) Common randomness in information theory and cryptography, Part I: Secret sharing, *IEEE Transactions on Information Theory*, vol. 39, pp. 1121–1132.

[11] Han, T. S. & Verdu, S. (1993). Approximation theory of output statistics, *IEEE Transactions on Information Theory*, vol. 39, pp. 752–772.

[12] Ignatenko, T., & Willems, F. M. J. (2008). Privacy leakage in biometric secrecy systems, *Proceedings of the 46th Annual Alerton Conference on Communication, Control and Computing*, U.S.A., pp. 850–857.

[13] Levin, L. A. (2003). The tale of one–way functions, *Problems of Information Transmission*, vol. 39, no. 1, pp. 92–103.

[14] Venugopalan, S., & Savvides, M. (2011). How to generate spoofed irises from an iris code template, *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 2, pp. 385–395.

[15] Puente, L., Poza, M. J., Ruiz, B., & Carrero, D. (2011). Biometrical fusion – input statistical distribution, *Advanced Biometric Technologies*, InTech, pp. 87–108. doi:10.5772/18092.

[16] Boumbarov, O., Velchev, Y., Tonchev K., & Paliy, I. (2011). Face and ECG based multi–model biometric authentication, *Advanced Biometric Technologies*, InTech, pp. 67–86. doi:10.5772/21842.

[17] Inuma, M., Otsuka, A., & Imai, H. (2009). Theoretical framework for constructing matching algorithms in biometric authentication systems, *Lecture Notes in Computer Science: Advances in Biometrics*, vol. 5558, pp. 806–815.

[18] Balakirsky, V. B., Ghazaryan, A. R., & Han Vinck, A. J. (2009b). Mathematical model for constructing passwords from biometrical data, *Security and Communication Networks*, vol. 2, no. 1, pp. 1–9.

[19] Balakirsky, V. B., & Han Vinck, A. J. (2010). A simple scheme for constructing fault–tolerant passwords from biometric data, *EURASIP Journal on Information Security*, vol. 2010. doi:10.1155/2010/819376.

[20] Balakirsky, V. B. (2012). Binary multimedia wrap approaches to verification over noisy data. Submitted to *The Computer Journal*.

IntechOpen