Security for Biometric Data

Claus Vielhauer[a], Ton Kalker[b]

[a]Technical University Darmstadt and Magdeburg University, Germany
[b]Philips Research, Prof. Holstlaan 4, 5656 AA Eindhoven, The Netherlands

ABSTRACT

Biometric authentication, i.e. verifying the claimed identity of a person based on physiological characteristics or behavioral traits, has the potential to contribute to both privacy protection and user convenience. From a security point of view, biometric authentication offers the possibility to establish physical presence and unequivocal identification. However from a privacy point of view, the use of biometric authentication also introduces new problems and user concerns. Namely, when used for privacy-sensitive applications, biometric data are a highly valuable asset. When such data are available to unauthorized persons, these data can potentially be used for impersonation purposes, defeating the security aspects that are supposed to be associated with biometric authentication. In this paper, we will systematically unveil critical sections based on the two generic biometric flow models for enrolment and authentication respectively. Based on these critical sections for biometric data in authentication systems, we will point out measures to protect biometric systems. It will be shown that especially techniques using non–reversible representations are needed to disallow malicious use of template information and we will discuss a variant of the Linnartz-Tuyls model for securing biometric templates in detail.

Keywords: Biometrics, Security, Privacy

1. INTRODUCTION

Biometric authentication, i.e. verifying the claimed identity of a person based on physiological characteristics or behavioral traits has the potential to contribute to both privacy protection and user convenience [1][2]. From an user-convenience point of view, biometric authentication has the advantage that it does not make use of pin codes or passwords that can be forgotten or tokens that can be lost. Similarly, biometric authentication offers the possibility of personalization, because a device or service can recognize a user and adapt its settings to the user's preferences. From a security point of view, biometric authentication offers the possibility to establish physical presence and unequivocal identification. However, with respect to the latter, it must be noted that biometric authentication seems to intrinsically characterize by 'larger than wished for' false acceptance and false rejection errors. Therefore, not necessarily all biometric recognition methods will be able to achieve the same level of security as for instance a pin code.

From a privacy and security point of view, the use of biometric authentication also introduces new problems and user concerns [3]. Namely, when used for privacy-sensitive applications, biometric data are highly valuable assets. In particular they are assets that are not easily renewed such as is the case with pin codes or tokens. For example, a stolen fingerprint template is not easily upgraded! When such data are available to unauthorized persons, these data can potentially be used for impersonation purposes, defeating the security aspects that are supposed to be associated with biometric authentication.

2. A GENERAL SIGNAL-BASED MODEL FOR AUTHENTICATION SYSTEMS

Bishop gives a general definition for authentication [4]: „Authentication is the binding of an identity to a subject". Consequently, any authentication process operates on information of two categories; an identification string logically and uniquely assigned to a subject and secondly, information related to the subject allowing a decision on authenticity, i.e. is the person the one she or he claims to be. This information may come from one or more if the following properties related to the person directly:

- **Knowledge** of the subject (e.g. password or other secret information)
- **Possession** of the subject (e.g. key token, identification card)
- **Biometric traits** of the subject (e.g. structure of fingerprints, handwriting style)

Additional environmental conditions such us temporal information (e.g. authentication for access control only at specific dates/times) or spatial position of the subject (e.g. authentication only for specific terminals) may be taken into account for authentication, but are not intrinsically linked to the subject's nature. Bishop presents a formal description for an authentication system consisting of five component sets as presented in the following table.

Information Component	**Designator**	**Description**
Authentication Information	*A*	Set of specific information with which entities prove their identities
Complementary Information	*C*	Set of information that the system stores and uses to validate the authentication information
Complementary Functions	*F*	Set of functions that generate the complementary information from the authentication information, i.e. $f \hat{I} F, f: A \to C$
Authentication Functions	*L*	Set of functions that verify identity, i.e. $l \hat{I} L, l: A \acute{} C \to \{true, false\}$
Selection Functions	*S*	Set of functions that enable an entity to create or alter *A* or *C*

Table 1 - Five components of Bishop's authentication system [4]

This generic modeling approach has been extended in recent work towards a generic signal based model for authentication systems [5]. Here, the authentication process is divided into three components (Human subject H, Authentication System U and Reference Storage R) that communicate via channels as seen from the following figure.

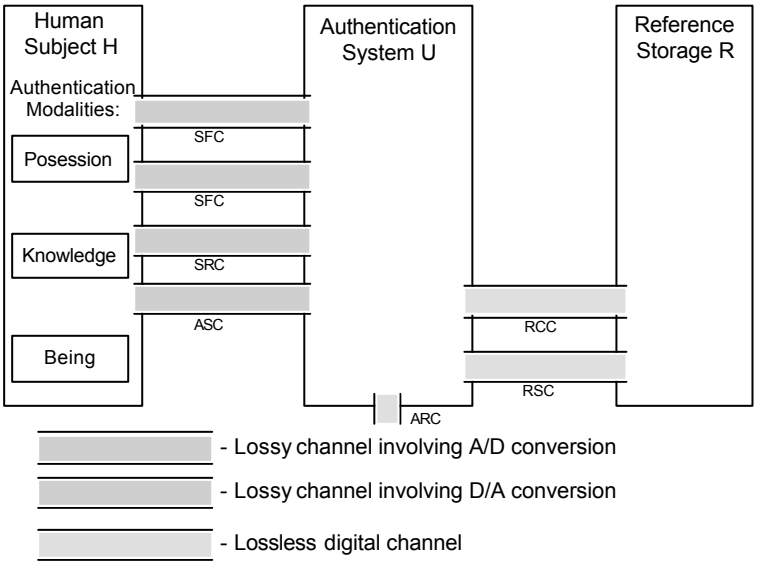

Figure 1: General signal-based model for authentication systems from [5]

In [5], six different channels have been determined between the model's components, which are summarized in the following table. Note that the Authentication Submission Channel is mandatory in all cases. Also note that in case of authentication by identification it is even sufficient. The Synchronization Forward and Reverse Channels (SFC and SRC, respectively) are required only for verification or challenge-response based authentication systems.

Short	Name	Purpose
ASC	Authentication Submission Channel	Transmission of Authentication Information from H to U
SFC	Synchronization Forward Channel	Accompanying Information from H to U (e.g. declared identity or response upon challenge)
SRC	Synchronization Reverse Channel	Information Request from U to H (e.g. challenge)
RCC	Reference Information Control Channel	Request for Reference Information from U to R (e.g. request biometric template of a specific user)
RSC	Reference Information Submission Channel	Transmission of Reference Information from R to U (e.g. biometric template data)
ARC	Authentication Result Channel	Communication of Authentication Result

Table 2 – Description of six channels of a general authentication system model [5]

For our further discussions, we will refer to this model and discuss the protection requirements for the system components for one specific authentication modality of biometrics.

3. BIOMETRIC PROCESSES

Two processes are intrinsic to all biometric authentication systems: one or more enrolment processes for each user of such system in order to register and logically link the biometric features to an individual. This process is prerequisite to the second category of processes, the authentication phase, where an actual sample of the biometric trait is compared to previously enrolled information and the system decides upon authenticity. These two processes are illustrated along with the involved data channels according to [5] in the following figure.

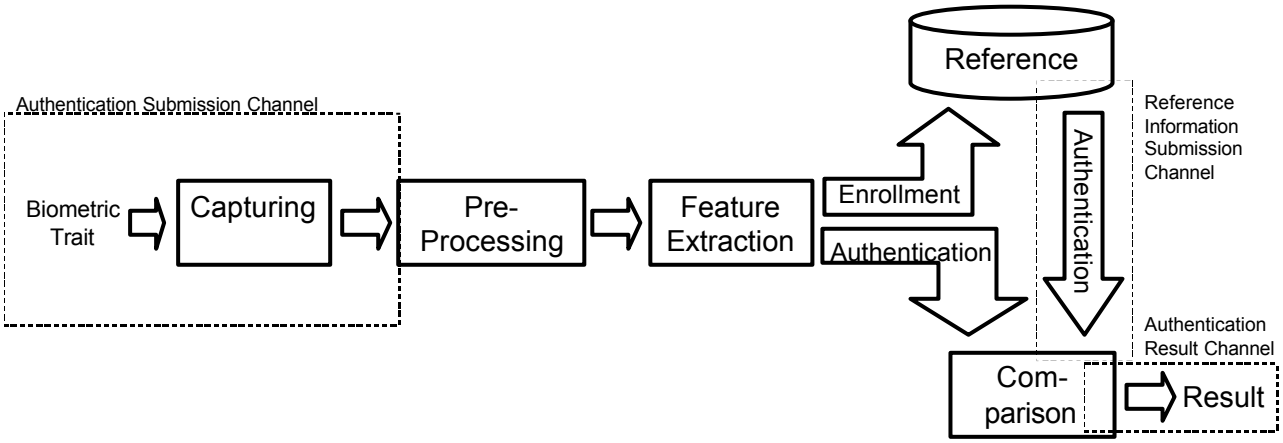

Figure 2 - Biometric authentication processes: Enrollment and Authentication

We will not discuss security measures to ensure authenticity during enrollment, as these should be handled by organizational measurements. Furthermore, we do not analyze means for physical protection of reference storage *R*. For our further discussion, we concentrate our discussions on the specific target of an attacker to gain authentication in place of another person. Views on additional possible aims of attackers, e.g. denial of service, would exceed the scope

of this article. Furthermore, we will focus on technical attacks attempting access to the system components U (Authentication System) and R (Reference Storage). Attacks to the human subject H will be neglected in this article, as these mainly aim at social or physical vulnerability of users. Methods here include criminal approaches like theft, extortion or social engineering like described by Mitnick and Simon [6] and are of a non-technical nature.

Attacks to the remaining system components U and R can be structured according to [5] into three classes:

- Physical Access to U
- Logical Access to U
- Logical Access to R

Note that physical access to U is included, as it comprises physical access to the biometric sensor devices from the analogue world. Here, methods for attacks can target both sensor and system levels by some physical manipulation, whereas logical access to U allows reading and changing information flow on the data channels linked to U. Furthermore, methods of spying and social engineering [6] can be considered as physical access to U, especially for behavioral biometrics, where an attacker's goal can be to study and train the authentication information of subjects.

4. SECURITY ASPECTS IN BIOMETRIC AUTHENTICATION SYSTEMS

In this section, we will systematically unveil critical sections of data flow and storage in respect to the security of the sensitive biometric. Based on this security analysis, we will then discuss one novel approach to solve one of the problems, i.e. how to secure biometric data storage (e.g. by physical, cryptographic and/or other multimedia signal processing means) with respect to the confidentiality of biometric information in the following section.

As stated earlier, we will not address the problems of providing authenticity of the users during enrollment and we also will not look into methods to physically protect the reference storage *R*. Rather than that, we will look into possibilities for a malicious subject attempting to get authentication. In order to do so, the attacker can either try to obtain some information during biometric processing, which he/she feeds into the authentication system at some later point in time, or system circumvention may be attempted. With these basic observations, the following potential attack scenarios can be determined:

Physical Access to U
In this scenario an attacker may physically access the sensor-system and/or the communication channel ASC. With respect to the sensor, physical manipulations of hardware include manipulations of the original sensor, replacement of the sensor or integration of an additional sensor controlled by the attacker with the goal of spoofing the biometric authentication information. All these attacks aim to replay attacks to the authentication system. Instead of physically manipulating the sensor, an attacker may attempt to tap physical signal wires, which implement the channels SRC, SFC and ASC from the model presented in Figure 1. If tapping of these signal lines is successful, the attacker again may use this access for replay attacks.

Logical Access to U
Similarly to the physical access to the signal lines that implement the information channels between the authentication system components, an attacker may attempt to digitally access *U*. If he/she gets read access to the channel, this approach may unveil authentication information *A* to the attacker. Furthermore, if write access is gained, this access can be used to feed authentication information into the system. Besides replay attacks, such logical approach may be used for brute-force attacks, e.g. by systematic generation of synthetic authentication data, similarly to brute-force password attacks.

Logical Access to R
In this attack scenario, an attacker gets access to the complementary information, which is stored for authentication purposes in the reference storage *R*. For the case of read access, the attacker may use the information to estimate some original authentication information *A* to access the system. This is especially a straightforward task, if the reference is stored in its original representation, the biometric raw signals. The other possibility, if an attacker gets write access to the reference storage *R* is the possibility to replace *C* with some complimentary information *C'*, which could match the attacker's biometric features.

From the general authentication system model, we thus can determine the following main critical sections, which could be vulnerable for attacks described in the scenario above and therefore require protection.

Channels:

ASC (Authentication Submission Channel): This channel is vulnerable to sniffing and replay attacks, implying logical access to *U*.

RCC/RSC (Reference Information Control Channel / Reference Information Submission Channel): This channel is vulnerable to sniffing of the complementary information *C* and reconstruction of the authentication information *A*, implying logical access to *R*.

ARC (Authentication Result Channel): This is channel used for transmitting the authentication result to the external world. If taken over by an attacker, physical access to U is achieved. In this case, the authentication process is completely circumvented as the attacker may have complete control over the authentication result, regardless of the information exchanged between the system components.

Securing of channels may be accomplished by physical means in case of wired communication, which may complicate a potential attack. Moreover, the information exchanged over the channels can by cryptographically secured in such way that only authorized system components are able to access the information. For the later protection approach, functions like symmetric/asymmetric encryption or digital signatures are well known methods from cryptography.

Components:

Major security problems arise once an attacker gets physical access to any of the system components *H*, *U* or *R*. Depending on the design of each of the subsystem, an attacker may replace or circumvent hardware and/or software components of modules. Also, functions for accessing information relevant for the authentication process (e.g. spyware) may be implemented. Measures to prevent these types of attacks include physical protection of the devices, such as effusion of sensor devices), mechanisms to disallow modifications to software code (e.g. PROM implementation) and also methods for Intrusion Detection.

Besides approaches to protect the system components of the authentication system, in order to increase the difficulty for an attacker to access information relevant for authentication, measurements can be taken to logically protect the information itself. Here, we have identified that particularly the reference information needs a reasonable degree of protection, because this data needs to be stored persistently and thus is potentially exposed to write or read attacks to *R*, as described earlier in this section.

To protect a biometric authentication system against malicious substitution of complementary information C, methods for data authentication may be applied. Especially for biometric modalities that are based on image or audio data, such authentication may be achieved by well-established methods of digital watermarking. By including a reference data authentication step for the complementary information *C* in the verification process, the system may be able to detect modifications in the reference data and react accordingly. A recent example for this approach is the embedding of fingerprint minutiae information in images [7]. However, this measure does not prevent attacks that are based on knowledge about *C*. A protection approach here is, in analogy to password-based authentication, where *C* represents a hash value of the original authentication information, not to store any data that would allow reconstruction of the authentication information. More precisely, a set of mapping functions $f \in F, f: A \rightarrow C$ is to seek in such way, that it is computationally impossible to restore the original authentication information *A* solely by knowledge of the complementary information *C*, or formally speaking, no easily computable functions f^{-1} exists such that $f^{1}: C \rightarrow A$.

Compared to other protection mechanisms (cryptography for secured channels, watermarking for data authentication) for biometric systems presented in this paper, the later considerations are very recent. Some initial implementations have been published, which aim to implement such functions. An implementation example is the biometric hash for handwriting dynamics [8], which takes the approach of storing statistical intervals and interval mapping of statistical features extracted from handwritten input. It is safe to assume that this aspect of securing biometric systems will be of

increasing interest in the future. However, only very recently, approaches for non-reversible biometric templates have been explored theoretically and we would therefore like to expand our discussion on this subject based on a model published by Linnartz and Tuyls [9] in the following section.

5. THE LINNARTZ-TUYLS MODEL

Linnartz and Tuyls [9] describe a security model for biometric data for the purpose of authentication. This model (referred to as the *password model*) takes the model of passwords and PIN codes as its basis, and updates this model to cope with the inherent fuzzy nature of biometric signals. The model introduced in [9] is written with continuous biometric templates in mind, and the notions of δ-contracting and ε-revealing refer to MSE-like distortion measures. In this section we describe a variant of the Linnartz-Tuyls model that is more geared towards discrete biometric templates (in particular those based on binary alphabets) and is therefore strongly related to [10] as well. In the remainder of this section we will show that it is possible to construct reference data C such that biometric authentication is possible (i.e. is able to deal with fuzzy data) and at the same time reveals only a specified amount of information about the original biometric templates A.

Following tradition in cryptography there are minimally two parties involved the password model: a *prover* Peggy who wants to prove that she is who she claims to be and a *verifier* Victor who wants establish the identity of Peggy. The password model not only assumes that Peggy may be unreliable but also that Victor may be unreliable. The password model is a knowledge based authentication model, where the knowledge of a password (or PIN code) is assumed to be proof of identity[1]. The unreliability assumption on Victor makes that Victor is not allowed to have a clear copy of the password: this would allow Victor to abuse the password for fraudulent purposes, allowing others (including himself) to impose as Peggy. In particular, Victor is only allowed access to a cryptographically hashed version of the password. In the enrollment phase of the password protocol, a trusted third party TP is needed to independently establish the identity of Peggy and associate a hashed password $q=H(p)$ of a password p to Peggy's identity, where H is public and fixed cryptographic hash function. Neither the TP not Victor (need to) have knowledge of the password p. By the definition of cryptographic hash functions it is computationally infeasible to learn the password p from its hashed version q. In the authentication phase Peggy presents her claimed identity to Victor ('Hello, I am Peggy') and similarly presents her password p to a trusted device that computes the hashed version q. Victor then compares q with the hashed version q' in its database associated to the identity Peggy and, depending upon the result of the comparison, verifies the identity of Peggy or not. This protocol is secure because knowledge of q is not sufficient to derive p. Both Victor and Peggy cannot infer passwords from the stored cryptographically hashed versions. In short, if Peggy passes the authentication test, Victor can conclude she really must be Peggy. Similarly, Peggy can be assured that her password (p) is only known to herself, and that Victor cannot abuse the data (q) given to him to recreate passwords.

A similar approach is obviously highly desirable for biometric templates. Unfortunately, a direct application of the ingredients of password model is inappropriate. The main problem is that cryptographic hash functions are bit-sensitive and that biometric templates have inherently a fuzzy nature. In particular if A_e is the biometric template of Peggy at the time of enrollment and A_a a corresponding biometric template at authentication time, then we may reasonably assume that A_e and A_a are similar in some appropriate distance measure. If this similarity is large enough (specified by some appropriately chosen threshold T), Victor can conclude with some specified probability of error that Peggy is who she claims to be. However, applying a cryptographic hash function H to the biometric templates will result in general in maximally different reference data C_e and C_a. Unless A_e and A_a are equal (which is in general highly unlikely), the verifier Victor has no computationally feasible method at his disposal to verify that C_e and C_a are derived from similar templates. The challenge is therefore to find a method that will allow scrambling of fuzzy biometric templates (to prevent Victor from creating fake biometric templates from reference data) that is robust to the fuzzy nature of biometric data.

In order to ease the following discussion we assume from the remainder of this section that biometric templates are represented by (long) binary strings and that Hamming distance is used as similarity measure. This is a mild assumption that fits many biometric modalities, like for the 2048-bit binary Iris Code of Daugman [11].

---

[1] As stated before, we do not consider cases where security is breached by non-technical attacks whereby Peggy reveals her password to unauthorized users by means of social engineering.

It seems natural that the application of error correcting codes (ECC) is a good way to impose robustness on biometric templates. In other words, for given a biometric template A, first apply an appropriately chosen ECC E to arrive at a *more stable version* R before applying a cryptographic hash function H. That is, the reference data stored by Victor are of the form H(E(A)). However, this *natural assumption* turns out to be false. An easy counter example can be given for biometric templates represented as bit strings of length 3, with a simple repeat code as ECC. We moreover assume that the enrollment phase is modeled as i.i.d. source with bit probability 0.5 and the authentication process is modeled by independent bit errors with error probability t. Then the following is easily derived (see also Figure 2):

a. The probability that at least one bit is in error for the 3-bit template during authentication is equal to $3t-3t^2+t^3$, which is approximately equal to 3t for small t; in other words the *false rejection rate* P_{fr} is approximately equal to 3t.

b. The *false acceptance rate* P_{fa} for the 3-bit templates is equal to (1/8);

c. The probability that the single bit of the reduced 1-bit template is in error during authentication is approximately equal to (3/2)t for small t; that is, $P_{fr} = (3/2)t$.

d. The false acceptance rate P_{fa} for the reduced 1-bit template is equal to (1/2).

From this it might seem that the reduced template is more robust (the false rejection rate is reduced by a factor 2). Note however this gain comes at the cost of an increased false acceptance rate, leading to the point (0.5,(3/2)t) on the ROC curve. This result can easily be improved by selecting any of the components of the original 3-bit templates: the error rate for a component-bit is equal to t (i.e. smaller than (3/2)t) at a fixed false acceptance rate of 0.5.

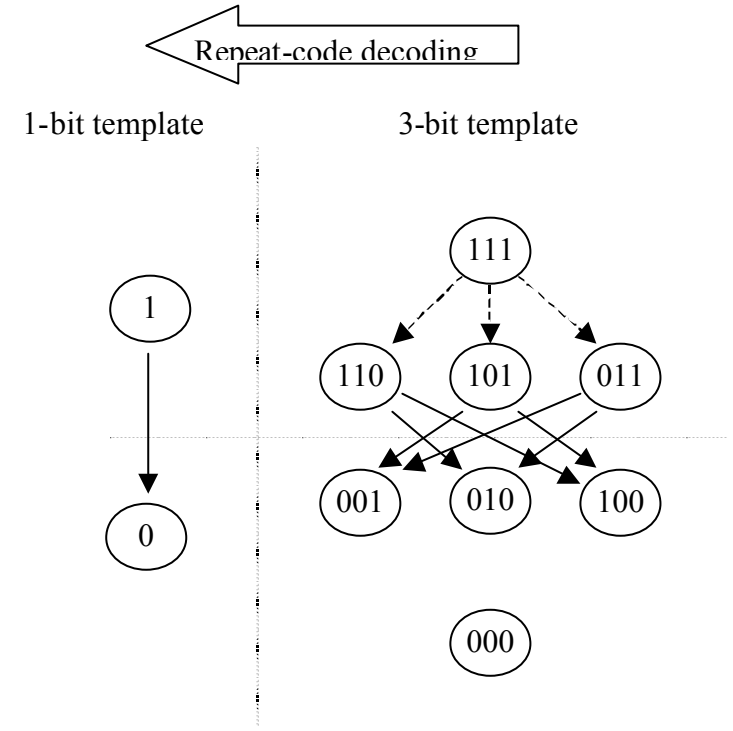

Figure 3 - Reduction to a 1-bit template. On the right hand side the transitions are shown that are linear in t (single bit flip) that cause a bit flip in the decoded domain (solid) or retain the bit value (dashed). The resulting transition probability (linear in t) in decoded domain is given 3*(1/4)*2*t = (3/2)t.

This example is typical for general error correcting codes as a means to stabilize biometric templates. In fact, the higher the dimensionality of the error code, the worse the performance will be. The explanation of this behavior relies on the fact that for (high-dimensional) error correcting codes and *random sampling* of the code space most of the code points

will be on the boundaries of the Voronoi regions [12] defining the error correcting code. In normal usage of error correcting codes the code points are selected at the center of the Voronoi regions (error correcting encoding). However in a typical biometric application the code points are selected randomly with respect to a fixed and pre-determined error correcting code. As a result there is a high sensitivity to noise, moving a code point from one Voronoi region to another one. In the example above this is reflected by the fact that for most code point (6 out of 8) a single bit flip is sufficient to change the sum of the 3-bit template from larger or equal than 2 to smaller than 2 (see also Figure 3).

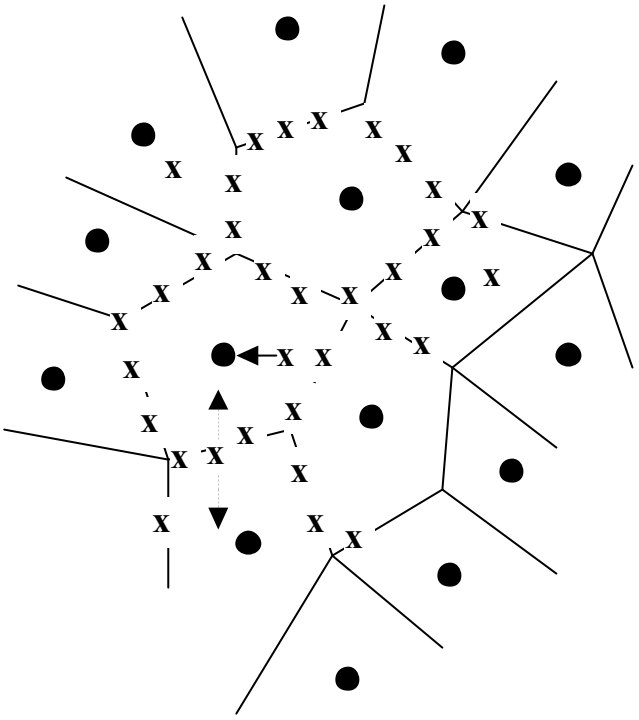

Figure 4 - Abstract representation of a high-dimensional error correcting code. The code partitions the ambient space into Voronoi regions (solid lines). A code point ('X') inside a Voronoi region is corrected (solid arrow) to the center of the Voronoi region (black dot). However, for high dimensional spaces and random sampling of the code space, most points ('X') are near the boundaries of the Voronoi regions. Only a small amount of noise is needed to move a code point from one region to another region.

The essential idea of Linnartz and Tuyls [9] is to overcome the *high-dimensionality effect* by the use of a *helper signal W* that essentially associates with every entry A in the template reference database but also an offset W such that A is a point in the coset W+E, where E is the code set of the chosen error correcting code. By the use of the helper signal w the biometric template A is essentially moved away from the boundary of the Voronoi regions, making it stable signal for cryptographic hashing. More precisely, the enrollment process is now modified as follows for a given error correcting code $E = \{e_i\}$. Without loss of generality we may assume that it easy to compute the index of a code word e, but that it is computational infeasible to derive e from its index[2] (see also Figure 4).

- A biometric template A is measured.
- A vector W is determined such that A-W is an element e_i of E, i.e. A-W has no errors when decode by E.
- The pair C = (W,i) is stored in the reference database.

The authentication process is a below.

[2] This can always be achieved by applying a cryptographic hash function to the any given index function.

- Peggy identifies herself to Victor ('Hi, I am Peggy').
- Peggy allows Victor to measure her biometric trait A_a , assumed to be a noisy version A+N of A.
- Victor retrieves Peggy's helper signal w from the reference database and computes A_a-W.
- Victor applies error correcting E to A_a-W and obtains a code word e.
- Victor computes the index j of e and compares j with the index i from the reference database. If j is equal to i, then Victor authenticates Peggy.

If the noise signal N is compatible with the error correcting code E (or, vice versa, if E is chosen to be compatible with the given noise power), then this scheme will successfully identify Peggy. In the original paper of Linnartz and Tuyls this robustness property is referred to as the *d-contracting* property, where the value δ refers to the allowed power of the noise signal N. More generally, the false rejection rate is given by the probability that the noise signal is too large to allow successful decoding by E. This rate is obviously decreased by increasing the volume of the Voronoi regions, i.e. more powerful error correcting codes. Furthermore, the false acceptance rate is determined by the probability that a random code falls in the Voronoi region of e.

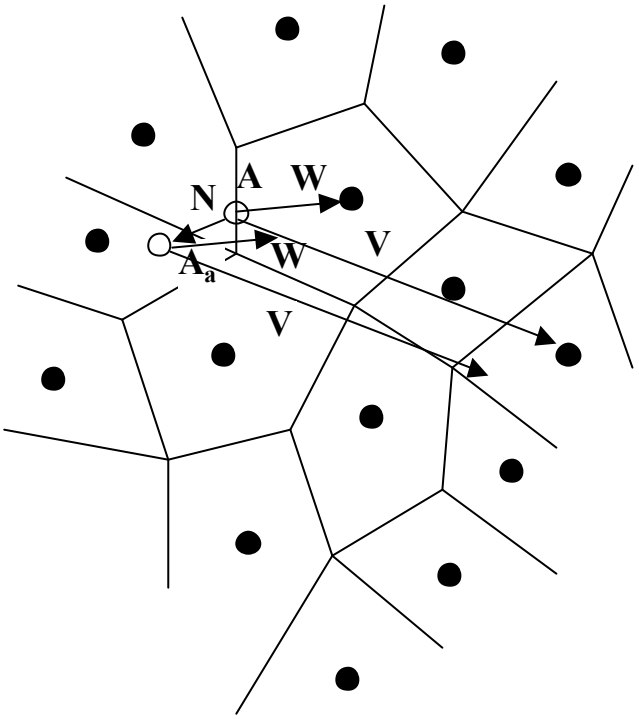

Figure 5 - Using a helper signal to move away from Voronoi boundaries. At enrollment time, a vector W is determined that moves the measured template A to a code point e. There is a degree of freedom in choosing W as the code point e need not be one that is closest to A. In particular V is also a valid helper signal.

Recall that the purpose of introducing cryptographic hash functions in authentication was the need to hide the original biometric templates A. In the model described above, this translates to the requirement that the template p cannot be easily retrieved from the reference data C=(W,i). We will consider this requirement in the following simple example.

Consider the biometric template A for Peggy represented as binary strings of length 3M. Let the error code E be the M-fold Cartesian product of repeat codes of length 3. That is, the code words of E are of the form (a_1, a_1, a_1, a_2, a_2, a_2, …, a_M, a_M, a_M), where a_i is either 0 or 1. Error correcting decoding for E is given by partitioning the templates in contiguous triples, and for each triple computing the majority symbol. In this way the index set for E is given in a natural manner as a vector of length M. A suitable hash function H is chosen to scramble the index set of E, making it computationally

infeasible to retrieve the original index i of a scrambled index H(i). Given an original biometric template $A = (A_1, \ldots, A_M)$, where each A_i is a triple of binary values, the enrollment process proceeds by selecting a code word e_i from the error correcting code E. The helper data W is constructed as the difference $W = A - e_i$, which is therefore a vector of triples of the form A_i or $1 - A_i$. The data stored in reference data base is equal to the pair C=(W, H(i)). One easily verifies that at the time of authentication, following the protocol described earlier, Peggy can successfully be authenticated from her biometric A_a if there is at most 1 bit error in each constituent triple. Moreover, the false acceptance rate for this authentication scheme is equal to 2^{-M} (assuming as usual an i.i.d. source model for biometric templates).

The security analysis of this scheme relates to the question how much can be learned about the original biometric template A from the reference data C=(W, H(i)). Assuming a properly chosen hash function H and a size M that is large enough, we may assume that it is infeasible to retrieve i form H(i). So, although in an information theoretic sense H(i) completely reveals i, in a practical and computational sense i may be assumed to be unknown. However, the helper data A does reveal information about the original template A. In this particular case every triple in W reveals two bits of information on the corresponding triple in A and leaves 1 bit of uncertainty. Formulated more precisely, the mutual information $I(A;W)$ between templates and helper data is equal to 2/3 (a value that is considered rather large). In the terminology of Linnartz and Tuyls the example above is referred to as *(2/3)-revealing*. It is without saying that in general we strive for as little exchange of information between original templates and helper data as possible. That is, we would like the scheme to be ε-revealing, $I(A;W)<\varepsilon$, where ε is as small as possible.

6. CONCLUSIONS AND FUTURE WORK

We have analyzed security aspects of biometric authentication systems based on a new general signal-based authentication system model[5], developed from the information based authentication model of Bishop[4]. Our evaluation has shown, that particularly biometric channel and storage protection are essential in biometric security applications. Main insights are that channel protection can be achieved by cryptographic measures, whereas protection of biometric reference data demands for additional protection schemes. Besides securing reference data by methods of data authentication, especially techniques using non–reversible representations are needed to disallow malicious use of template information. To this end, we have presented a variant of the Linnartz-Tuyls secure biometric template method [9]. We have concentrated on modeling secure biometric templates in terms of error correcting codes and cryptographic hashing functions. Moreover, we have sketched how false acceptance rates, false rejection rates and security are related. Most importantly, we have shown that it is possible to construct biometric hashing function by including helper data in the reference data. The model presented is still mainly of a theoretical nature and much work still needs to be done for constructing practical codes and fitting existing biometric modalities in the model presented. Our future work will elaborate to which degree practical applications like the Bio-Hash [8] implement non-reversible templates as described in the theoretical work of Linnartz-Tuyls and which level of accuracy can be achieved. Finally, we may observe an interesting analogy with Dirty Paper coding techniques in the watermarking community [13]. In this analogy we may view the helper data as representing quantization. It is well known in the watermark community that improved performance can be obtained by *distortion compensation*, i.e. by only fractionally translating an input symbol in the direction of the center of a quantization interval. One may wonder if a similar approach pays of in a biometric context where there may be an improved performance in terms of security.

ACKNOWLEDGMENTS

The authors would like to thank Jean-Paul Linnartz and Pim Tuyls for their contribution by the Delta-Contracting and Epsilon-Revealing Biometric Authentication model and the fruitful discussions in context with this work.

REFERENCES

1. E. Aarts. Ambient Intelligence in HomeLab. Philips Research, The Netherlands, 2002
2. S. Brostoff and A. Sasse. Are Passfaces more usable than passwords? A field trial investigation. In McDonald S. et al (Eds), 'People and Computers XIV - Usability or Else', Proceedings of HCI 2000, Sunderland, UK, pp 405-424, Springer, September 2000.

3. Adams and M. A. Sasse (2001). Privacy in Multimedia Communications: Protecting Users, Not Just Data. In A. Blandford, J. Vanderdonkt & P. Gray (Eds.), 'People and Computers XV - Interaction without frontiers', Joint Proceedings of HCI2001 and ICM2001, Lille, France, pp 49-64, Springer, Sept. 2001.

4. M. Bishop. Computer Security, pp. 309-337, Addison-Wesley, Boston, U.S.A, 2003

5. C. Vielhauer. Signal-based Modeling of Biometric Authentication Systems, Technical Report at Otto-von-Guericke University Magdeburg, http:// http://wwwiti.cs.uni-magdeburg.de/~vielhaue/, 2003

6. K. D. Mitnick and W. L. Simon. The Art of Deception: Controlling the Human Element of Security, Wiley, Hoboken, NJ, U.S.A, ISBN: 0-471-23712-4

7. A. K. Jain and U. Uludag. Hiding Fingerprint Minutiae In Images, Proc. Automatic Identification Advanced Technologies (AutoID), pp. 97-102, New York, March 14-15, 2002

8. C. Vielhauer, R. Steinmetz and A. Mayerhöfer. Biometric Hash based on Statistical Features of Online Signature, Proceedings of the International Conference on Pattern Recognition (ICPR) , pp. 1:123-126, 2002.

9. J.-P. Linnartz and P. Tuyls. New Shielding Functions to Enhance Privacy and Prevent Misuse of BiometricTemplates, Proceedings of the 4th International Conference on Audio and Video Based Biometric Person Authentication, LNCS 2688, pp. 393-402, Guilford, UK, June 9-11, 2003.

10. A. Juels and M. Sudan. A Fuzzy Vault Scheme, Proceedings of IEEE International Symposium on Information Theory, 2002.

11. J. Daugman. High confidence visual recognition of persons by a test of statistical independence, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 15(11), pp. 1148-1161, 1993.

12. H.M.C. Coxeter. Regular Polytopes. Dover Publications, New York, 1973.

13. M. Costa. Writing on dirty paper. IEEE Transactions on Information Theory, 29(3), pp. 439–441, 1983.