Enhancing Biometrics Security

Tiago Santos[†], Gonçalo Lourenço[¥], Luís Ducla Soares[†], Paulo Lobato Correia[¥]
[†] Instituto de Telecomunicações/ISCTE, Av. das Forças Armadas, 1649-026 Lisboa, Portugal
[¥] Instituto de Telecomunicações/IST, Av. Rovisco Pais 1, 1049-001 Lisboa, Portugal
Phone: +351-218418461, Fax: +351-218418472, e-mail: {lds,plc}@lx.it.pt

***Abstract* — The use of biometrics (e.g., fingerprints, irises, faces) for recognizing individuals is becoming increasingly popular and many applications are already available. Biometrics are intrinsically associated with individuals and cannot be forgotten or shared with others. However, one of the most relevant vulnerabilities of biometrics is that once a biometric template is compromised, it cannot be reissued, updated or destroyed. An attacker could then gain access to all the accounts/services/applications using that same biometric trait. This paper proposes a biometric verification system using distributed source coding principles, with enhanced security with respect to traditional biometric verification systems. The generation of different templates from the same biometric data is supported, as well as cancelable templates. Furthermore, it will not be possible to recover the original biometric data from the stored data, thus guaranteeing its privacy.**

Keywords: Biometrics security, cryptographic hash function, distributed source coding, error correcting codes

I. INTRODUCTION

The use of biometrics (e.g., fingerprints, irises, faces) for recognizing individuals is becoming increasingly popular and many applications are already available. Although these applications can be fundamentally different, they can still be grouped into one of two categories: *verification* and *identification* [1][2][3]. While verification systems authenticate a person's identity by comparing the captured biometric characteristic with that person's own biometric template previously stored in the system, identification systems recognize an individual by searching the entire template database for a match with the captured biometric characteristic. Here, only verification systems will be considered since this corresponds to the case where the proposed security enhancements are more relevant, as will shortly become clear.

In verification systems, such as access control systems, the use of biometrics has several advantages over the use of passwords. The first one is the fact that biometrics are intrinsically associated with individuals and cannot be forgotten or shared with others. In addition to this, adequately chosen biometrics have a much higher entropy than poorly chosen passwords and, therefore, are less susceptible to brute force attacks. Finally, systems that rely on biometric verification require very little user expertise and, therefore, can be easily and widely deployed.

The authors acknowledge the support of Fundação para a Ciência e Tecnologia (FCT), under Instituto de Telecomunicações project P436 - BIONSE.

Despite the numerous advantages of biometrics, some disadvantages also exist when compared to passwords. For instance, it was noted in [4] that one of the most relevant vulnerabilities of biometrics is that once a biometric image or template is stolen, it is stolen forever and cannot be reissued, updated, or destroyed. Another problem associated with the use of biometrics is that once a biometric is chosen, the same biometric will be used to access many different systems. This means that, if it is compromised, the attacker will have access to all the accounts/services/applications. This is the equivalent of using the same password across multiple systems, which can lead to some very serious problems in terms of security, as can be easily understood.

In particular, embedded devices, such as smart cards, are especially vulnerable to eavesdropping and attacks [5]. Thus, protection mechanisms to provide a secure storage for the reference biometric template need to be investigated. Recently, novel cryptographic techniques such as fuzzy commitment and fuzzy vault were proposed [6][7]. These schemes integrate error correcting codes to allow protecting data subject to some noise, as happens with the acquisition of biometric templates. Clancy et al. [8] employed the fuzzy vault scheme on a secure smart card system, where fingerprint authentication is used to protect the user's private key. Yang, et al. [9] further addressed the issue to develop an automatic and adaptive recognition system. Linnartz et al. [10] precisely formulated the requirements for protecting biometric authentication systems, presenting a general algorithm meeting those requirements. The feasibility of template-protected biometric authentication systems was further demonstrated in [11].

Given the identified vulnerabilities of biometric verification systems, it is urgent that these problems are addressed. This paper proposes a biometric verification system, which will have enhanced security with respect to traditional systems, exploring the combined usage of distributed source coding and hash functions. The used biometrics will be the iris, since it has been reported to provide some of the best results for verification systems and it remains fairly unaltered during a person's lifetime.

The rest of this paper is organized as follows. Section II presents the proposed architecture for a secure biometric verification system, while the implementation details are described in Section III. Finally, to conclude the paper, some final remarks about the strengths of this type of approach are presented in Section IV, as well as an outline of the envisioned future developments.

II. PROPOSED SYSTEM ARCHITECTURE

After having identified the advantages and disadvantages of using biometrics in verification systems, it would be interesting to design a verification system that preserves all

the advantages of biometrics described above but manages to eliminate the mentioned disadvantages. To do so, in addition to the typical requirements imposed on biometric systems, the following requirements have to be met:

- **Different templates from the same biometric** – Once a biometric is chosen, it will be used to access many different systems. This means the same biometric template will be stored in several databases, each corresponding to a different system. This may lead to a highly insecure situation, because if a given system is compromised, then all the other systems using the same biometric will also be potentially compromised. This leads to a requirement of being able to generate many different biometric templates from the same biometric trait. This way, even if one of the systems becomes compromised, the other systems will not be.
- **Cancelable templates** – Another problem is that a stolen biometric image or template would be stolen forever. This is unacceptable because the number of biometrics that a user can enroll is very limited. For instance, a user only has ten fingers, two eyes, etc. The corresponding requirement is the possibility to generate cancelable biometric templates [12]. This way, if an enrolled biometric template is somehow compromised, it can be simply deleted and a new one is issued, still based on the same biometric.
- **Private biometrics** – Biometric templates should never allow an attacker to recover the original biometric data from them. This is of the utmost importance, as all biometric templates are generated from the original data, which means that if an attacker has access to it, all the systems where that biometric is used could be potentially compromised. This leads to a requirement that the original biometric data cannot be recovered from the stored data, thus remaining private [13]. This is also called information hiding.

The proposed biometric recognition system, which relies on distributed source coding principles and cryptographic hash functions, is able to meet these requirements and, therefore, achieves the necessary enhanced security. This is reflected in the proposed system architecture, which is presented in Figure 1.

In the enrollment stage of a typical biometric verification system, after the biometric acquisition module, some processing is applied in order to obtain the biometric template, **x**, which is then stored in a database. Here, however, the biometric data is never stored in the database to prevent it from being stolen. Instead, after the biometric has been acquired and the biometric template has been generated, an error correcting code and a cryptographic hash function will be applied to it in parallel. The result of these two operations, **s** and **h**, respectively, will then be stored in the database; this will be referred to in the rest of the paper as the **secure biometric template**. It should be pointed out that it is impossible to recover any biometric data from this secure template as the hash function is not invertible and **s** corresponds only to the parity bits generated by the error correcting code, the information bits being simply discarded.

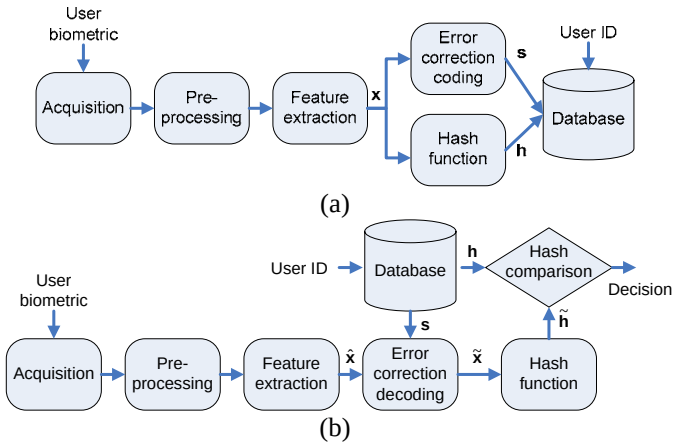


Figure 1 – Proposed system architecture: (a) Enrollment stage; (b) Verification stage.

During the verification stage, the probe biometric is acquired and the corresponding template, $\hat{\mathbf{x}}$, is generated. The error correcting decoder uses $\hat{\mathbf{x}}$ together with the parity bits stored for that user, to recover the original biometric template **x** if the user is who he claims, or something completely different if he is not. The problem here is that **x** itself is not stored in the database, but only a hashed version of it. Therefore, the output of the error correcting decoder needs to be hashed. Only then, is the result compared to the hash that is stored in the database. If the two hashes are equal, then the user is validated to be who he claims to be.

With this system, the three requirements above are verified. In particular, it is possible to generate many different secure biometric templates from the same biometric trait; it is just a matter of using a different set of attributes for the hash function. It is also easy to cancel a secure template by simply deleting the compromised template and generating a new one by using different attributes for the hash function. Finally, since the biometric data is never stored in a database, this guarantees that this information remains private.

III. Implementation Details

The first decision that had to be made before starting the actual implementation of the proposed architecture was to choose the biometric trait to be used. Due to its numerous advantages over other biometric traits for verification systems, the iris [14] was chosen. After this decision, it then becomes possible to decide how each of the modules in the proposed architecture will be implemented.

As shown in Figure 1, the proposed system includes five main modules: biometric data acquisition, pre-processing, feature extraction, cryptographic hash function and error correction coding.

The proposed solutions for each of these modules are described next with more detail.

A. Acquisition

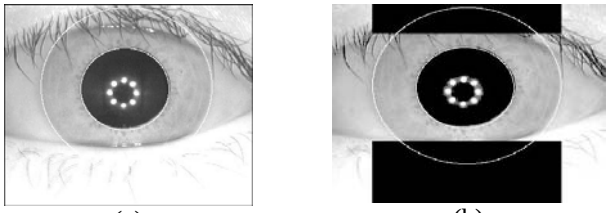
The acquisition module, absolutely necessary in a real biometric verification system, has not been implemented by the authors at the current simulation stage. Instead, it is replaced by a large database of iris images, like the one developed by the Chinese Academy of Sciences' Institute of

Automation (CASIA) [15]. This database consists of 22051 iris images from more than 700 subjects. All iris images are 8 bit gray-level JPEG files, collected under near infrared illumination.

B. Pre-processing

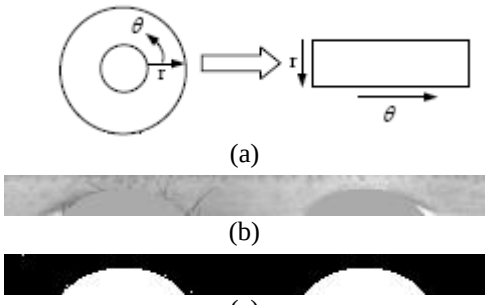
The first step after acquisition is to extract the iris from the input eye images. The iris area is considered as a circular crown limited by two circles. The iris inner (pupillary) and outer (scleric) circles are detected by applying the circular Hough transform [16], relying on edge detection information previously computed using a modified Canny edge detection algorithm [17].The eyelids often occlude part of the iris, thus being removed using a linear Hough transform [18] .The presence of eyelashes is identified using a simple thresholding technique. The output of this segmentation step is illustrated in Figure 2 (b).

Iris segmentation results may appear at different positions and scales and, thus, require a normalization, done with Daugman's rubber sheet model [19]. This method maps the circular iris image into a rectangular representation, as illustrated in Figure 3. The size of the normalized iris image is 20x240 pixels.



(a) (b)

Figure 2 – Iris segmentation in the presence of noise (eyelids and eyelashes): (a) Input image; (b) Iris segmentation result.



(a)

(b)

(c)

Figure 3 – Illustration of the normalization process: (a) Daugman's rubber sheet model [19]; (b) Normalized iris texture image; (c) Noise mask of the normalized image.

C. Feature Extraction

Once the iris texture is available, features are extracted from it to generate a more compact representation, also called the biometric template. To extract this representation, the two-dimensional normalized iris pattern is convolved with a Log-Gabor wavelet. The resulting phase information is quantized, using two bits per pixel. The resulting iris template is composed of 9600 bits, stored as a 20×480 binary matrix.

D. Hash Function

Cryptographic hash functions are a deterministic procedure that takes an arbitrary block of data and returns a fixed-size bit sequence. This bit sequence is very easy to compute and is generated by a dispersion algorithm, the result being usually represented in hexadecimal base. The process is unidirectional, which makes it practically impossible to recover the original content from the hashed bit sequence. Moreover, a very small change in the original content will result in a considerable change in the value of the hash.

Available cryptographic hash functions include: MD2, MD5, SHA-1, SHA-384 and SHA-512. In the present implementation, SHA-512 is selected due to its enhanced security characteristics.

E. Error Correction Coding

In the proposed system, error correction coding is used to correct biometric templates in the verification stage. In this stage, the probe template of a legitimate user is (error) corrected in order to recover the original template, obtained during enrollment; this should be possible because both templates are fairly similar. However, for an illegitimate user, whose probe template is fairly different from the one originally enrolled by the legitimate user, it should not be possible to recover the original from the probe template. Therefore, the selected error correcting code should be strong enough to correct templates of legitimate users, but not so strong as to also correct the templates of illegitimate users. Therefore, the main challenge here is to find the threshold of performance needed for the error correcting codes.

In order to precisely determine the adequate threshold of error correcting performance, tests should be done by varying the performance of the error correcting code with enough granularity, which is not possible with all the existing codes.

Since low-density parity check (LDPC) codes allow their performance to be adjusted with a very fine granularity, they were chosen here for this module. LDPC codes are a class of linear block codes, whose name comes from the fact that their parity-check matrix contains only a few 1's in comparison to the amount of 0's [20]. The code performance can be defined according to the number of columns in the parity-check matrix or number of 1's per column. In addition to their granularity, the error correcting performance curve of LDPC codes is very steep when the limit is approached, which basically means that it will be possible to precisely select which templates can be corrected and which ones cannot.

With these two properties, LDPC codes are ideally suited for this type of application, allowing to choose an error correcting code whose performance closely matches the desired operation threshold. The steepness in performance and the granularity of LDPC codes is illustrated in Figure 4.

F. Graphical User Interface (GUI)

The software development includes two modules, one for users' enrollment, and another corresponding to the verification stage. An illustration of the verification module graphical user interface is included in Figure 5.

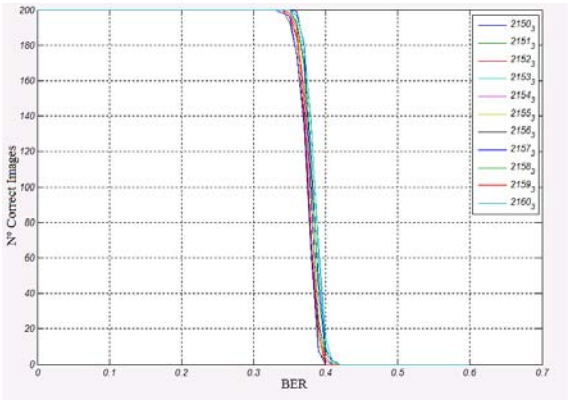


Figure 4 – LDPC codes steepness and granularity illustration (2150-2160 columns and three 1's per column).

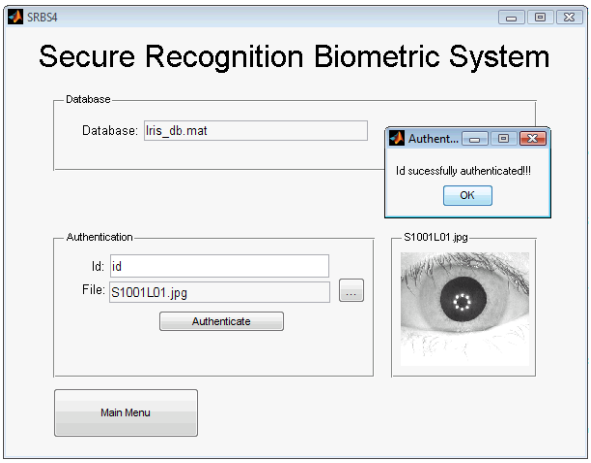


Figure 5 – GUI: Verification Module.

IV. FINAL REMARKS

This paper discusses the problem of biometric verification with enhanced security when compared to traditional systems. The proposal combines cryptographic hash functions and distributed source coding principles to guarantee that different biometric templates can be generated from the same biometric trait, that these templates can be cancelled if needed, and that the original biometric data cannot be recovered from the stored templates.

An implementation using iris as the selected biometric trait is described. In particular, the use of LDPC codes is an asset for this implementation, since these codes, with their granularity and steepness, allow working with a near optimal threshold of performance for secure biometric systems.

The present implementation takes the *Iriscode* software, developed by L. Masek [21], as the basis for the traditional part of our biometric system. At present, the decision threshold is being investigated, in order to subsequently adjust the LDPC coding module.

REFERENCES

[1] A. K. Jain, S. Pankanti, S. Prabhakar, L. Hong, A. Ross, "Biometrics: A Grand Challenge", *Proc. of the International Conference on Pattern Recognition*, Vol. 2, pp. 935–942, August 2004.

[2] J. Wayman, A. Jain, D. Maltoni, D. Maio, *Biometric Systems: Technology, Design and Performance Evaluation*, Springer-Verlag, 2005.

[3] D. Maltoni, D. Maio, A. K. Jain, S. Prabhakar, *Handbook of Fingerprint Recognition*, Springer, 2003.

[4] B. Schneier, "Inside Risks: The Uses and Abuses of Biometrics", *Communications of the ACM*, Vol. 42, No. 8, pp. 136, August 1999.

[5] J. Jeong, M. Chung, H. Choo, "Integrated OTP-based User Authentication Scheme Using Smart Cards in Home Networks", *Proc. of the International Conference on System Sciences*, Waikoloa, HI, USA, January 2008.

[6] A. Juels, M. Sudan, "A Fuzzy Vault Scheme", *Proc. of the International Symposium on Information Theory*, p. 408, Lausanne, Switzerland, June 2002.

[7] A. Juels, M. Wattenberg, "A Fuzzy Commitment Scheme," *Proc. of the 6th ACM Conference on Computer and Communications Security*, pp. 28-36, New York, NY, USA, 1999.

[8] T. C. Clancy, N. Kiyavash, D. J. Lin, "Secure Smartcard-based Fingerprint Authentication," *Proc. of the ACM Workshop on Biometrics: Methods and Applications*, pp. 45-52, Berkeley, CA, USA, 2003.

[9] S. Yang, I. Verbauwhede, "Automatic Secure Fingerprint Verification System Based on Fuzzy Vault Scheme", *Proc. of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. 609-612, Philadelphia, PA, USA, 2005.

[10] J.-P. Linnartz, P. Tuyls, "New Shielding Functions to Enhance Privacy and Prevent Misuse of Biometric Templates", *Proc. of the 4th International Conference on Audio- and Video-Based Personal Authentication*, pp. 393-402, Guildford, U.K., 2003.

[11] P. Tuyls, A. H. M. Akkermans, T. A. M. Kevenaar, G.-J. Schrijen, A. M. Bazen, R. N. J. Veldhuis, "Practical Biometric Authentication with Template Protection," *Proc of the 5th International Conference on Audio- and Video-Based Personal Authentication*, pp. 436-441, Rye Brook, NY, USA, 2005.

[12] N. K. Ratha, J. Connell, R. Bolle, "Enhancing Security and Privacy in Biometrics-based Authentication Systems", *IBM Systems Journal*, Vol. 40, No. 3, pp. 614-634, 2001.

[13] G. I. Davida, Y. Frankel, B. J. Matt, "On Enabling Secure Applications Through Off-Line Biometric Identification", *Proc. of IEEE the Symposium on Privacy and Security*, pp. 148-157, Oakland, CA, USA, May 1998.

[14] S. Yang, I. Verbauwhede. "Secure Iris Verification", *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. II-15-20, Honolulu, HI, April 2007.

[15] CASIA website, http://www.cbsr.ia.ac.cn/IrisDatabase.htm

[16] T. Kawaguchi, D. Hidaka, M. Rizon, "Detection of eyes from human faces by Hough transform and separability filter", *Proc. of the IEEE International Conference on Image Processing*, Vol. 1, pp. 49-52, Vancouver, Canada, 2000.

[17] J. Canny, "A Computational Approach to Edge Detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 8, pp. 679-714, 1986.

[18] R. Duda, P. Hart, "Use of Hough Transformation to Detect Lines and Curves in Pictures: Graphics and Image Processing", *Communications of the ACM*, Vol. 15, pp. 11-15, 1972.

[19] J. G. Daugman, "How Iris Recognition Works", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 14, No. 1, pp. 21–30, January 2004.

[20] R. G. Gallager, *Low Density Parity-Check Codes*, MIT Press, Cambridge, MA, USA, 1963.

[21] L. Masek, P. Kovesi, *MATLAB Source Code for a Biometric Identification System Based on Iris Patterns*, School of Computer Science and Software Engineering, University of Western Australia, Australia, 2003.