Biometrics & Security: Combining Fingerprints, Smart Cards and Cryptography

THÈSE N^{O} 4748 (2010)

PRÉSENTÉE LE 25 AOÛT 2010

À LA FACULTÉ INFORMATIQUE ET COMMUNICATIONS
LABORATOIRE DE SÉCURITÉ ET DE CRYPTOGRAPHIE
PROGRAMME DOCTORAL EN INFORMATIQUE, COMMUNICATIONS ET INFORMATION

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Claude BARRAL

acceptée sur proposition du jury:

Prof. A. Lenstra, président du jury
Prof. S. Vaudenay, Dr A. Tria, directeurs de thèse
Prof. B. Dorizzi, rapporteur
Dr A. Drygajlo, rapporteur
Prof. A. Ross, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2010

Acknowledgments

First of all, I would like to thank **David Naccache** for his crazy idea to give me the opportunity to start a PhD thesis, on the late, and **Serge Vaudenay** for his welcoming in **LASEC,** for teaching me cryptography, and his incredible patience with my slow advancement in this PhD work.

Many thanks to **Jean-Pierre Gloton, David Naccache** and **Pierre Paradinas** for supporting my Doctoral School application.

I must thank all my colleagues in **Gemplus,** then **Gemalto,** for all their support. Especially **Pierre Paradinas** for hiring me, more than ten years ago, in his GRL team - the Gemplus Research Lab - **Denis Praca,** my very first mentor at Gemplus and **Michel Agoyan** for his precious support whatever was the subject (e.g. Hardware, Software, Chip Design, Trainees management). Then **Eric Brier** and **Cédric Cardonnel,** first persons to work with me on Biometrics. **Pascal Paillier** and **Louis Goubin** for their support in cryptography. **Jean-Louis Lanet** for giving me the opportunity to give my very first courses at universities.

Precisely, I would like to thank every person having trusted me for my teaching skills on Biometrics, Smart Cards and Cryptography: **Traïan Muntean** at Ecole Supérieure d'Ingénieurs de Luminy, Marseille, France. **Caroline Fossati** at Ecole Centrale Marseille, France. **Marie-Laure Potet** at ENSIMAG, Grenoble, France. **Assia Tria** and **Laurent Freund** at Ecole des Mines de St Etienne - Centre Microelectronique de Provence George Charpak, Gardanne, France. **Thierry Fournel** at Institut Supérieur d'Optique, St Etienne, France.

Thanks a lot to all my past and present colleagues at LASEC for helping me with practical work during cryptography learning: **Pascal Junod, Gildas Avoine, Jean Monnerat, Thomas Baignières, Sylvain Pasini, Martin Vuagnoux, Julien Brouchier.**
And **Philippe Oechslin** for teaching me IT security at EPFL.

My professional relations were very helpful: especially **Bernadette Dorizzi** from INT Evry, **Nicolas Delvaux** and **Loïc Bournon** from Sagem, **Jean-François Mainguet** from Atmel.

The rock band which I'm proud to be the bass guitar player and helped me, for more than five years now, to counter-measure with professional and thesis stress. Thanks a lot to past and present members: **Eric Brier, Julien Brouchier, Cédric Cardonnel, David Hallé, Laurent Lagosanto** and **Daniela Laurans.** And thanks God for my very first marathon (during this dissertation's redaction), and finally a second one while I was in late with the redaction...

Ecole des Mines de St Etienne and CEA-Leti welcomed me in their lab and gave me the opportunity to pursue my research. Thanks a lot to **Assia Tria** and **Alain Merle**. My colleagues were very helpful with Linux, LaTeX, and other stuffs: **Michel Agoyan, Sylvain Bouquet, Pascal Manet, Jean-Baptiste Rigaud, Bruno Robisson**. For their collaboration on biometrics, many thanks to **Stéphanie Anceau, Philippe De Choudens, Abdel Yakoub**.

David Naccache and **Assia Tria** finally convinced me not to give up in particular hard times, thanks!

Many thanks to one external reader and spellchecker: **Bart Bombay**, and to the jury: the President, the Director, the co-Director and the three Examiners.

And last but not least, MANY THANKS to my family. Especially my wife, **Agathe,** and children **Julien, Antoine, Axel,** for supporting me. And my mother, mother-in-law, sisters, nieces and everyone having helped me to save time for my studies and helped my wife while I was in Lausanne few days a week for several months.

— À ceux que j’aime.

“Tout le monde savait que c’était impossible. Il est venu un imbécile qui ne le savait pas et qui l’a fait.”

— Marcel Pagnol.

Résumé

Depuis le début de ce siècle et plus particulièrement depuis les évenements du 11 Septembre 2001 aux Etats-Unis, plusieurs technologies biométriques sont considérées comme assez matures pour être un nouvel outil de sécurité. Généralement associées à un support personnel afin de respecter la vie privée, les données biométriques de référence sont enregistrées dans des équipements électroniques securisés, tel les cartes à puce et les systèmes utilisent des outils de cryptographie pour entrer en communication avec la carte à puce et échanger les données biométriques de manière sécurisée. Après une introduction générale sur la biométrie, les cartes à puce et la cryptographie, une second partie adressera nos travaux sur les fausses empreintes et les failles des systèmes de capture d'empreintes digitales. La troisième partie présentera notre approche d'un algorithme léger de reconnaissance d'empreinte digitale dédié aux cartes à puce. La quatrième partie détaillera les protocoles de sécurité dans les applications telles que la carte PIV, et présentera notre implémentation dans la carte à puce du protocole proposé par le NIST. Finalement, une cinquième partie adressera l'interaction entre Cryptographie et Biométrie, leur antagonisme, leur complémentarité, et présentera notre application d'un protocole de challenge-response aux données biométriques afin de faciliter la reconnaissance d'empreintes digitales.

Mots-clés: Biométrie, Cartes à Puce, Cryptographie, Reconnaissance d'Empreinte Digitale, Match-On-Card, Algorithmes Légers, Electronique Embarquée, Protocole Sécurisé de Vérification d'Identité par Biométrie, Cryptographie-Biométrie Interaction.

Abstract

Since the beginning of this brand new century, and especially since the 2001 Sept 11 events in the U.S, several biometric technologies are considered mature enough to be a new tool for security. Generally associated to a personal device for privacy protection, biometric references are stored in secured electronic devices such as smart cards, and systems are using cryptographic tools to communicate with the smart card and securely exchange biometric data. After a general introduction about biometrics, smart cards and cryptography, a second part will introduce our work with fake finger attacks on fingerprint sensors and tests done with different materials. The third part will present our approach for a lightweight fingerprint recognition algorithm for smart cards. The fourth part will detail security protocols used in different applications such as Personal Identity Verification cards. We will discuss our implementation such as the one we developed for the NIST to be used in PIV smart cards. Finally, a fifth part will address Cryptography-Biometrics interaction. We will highlight the antagonism between Cryptography - *determinism, stable data* - and Biometrics - *statistical, error-prone* -. Then we will present our application of challenge-response protocol to biometric data for easing the fingerprint recognition process.

Keywords: Biometrics, Smart Cards, Cryptography, Fingerprint Recognition, Match-On-Card, Light Weight Algorithms, Embedded Electronics, Cryptography-Biometrics Interaction, Secure Biometric Identity Verification Protocol.

Contents

List of Figures

List of Tables

Glossary

BIOMETRICS

authentication (or verification)	the process of proving an identity by comparing a candidate biometric sample against a known reference sample, also known as *one-to-one* recognition.
biometric system	system for the purpose of the automated recognition of individuals based on their behavioural and biological characteristics.
candidate	fresh biometric sample captured at recognition session to be compared to a reference sample for further authentication or identification.
decision	the process of deciding whether the comparison between a candidate and a reference is positive (pass) or not (negative, fail) .
enrollment	the registration of the user in the system by capturing a representative biometric sample to be stored as the reference.
extraction	the process of converting a captured biometric sample into biometric data so that it can be compared to a reference template.
failure to acquire (FTA)	error rate relative to the impossibility of capturing a usable image, FTA is considered as part of FTE.
failure to enroll (FTE)	error rate relative to the impossibility to extract usable features from an image.
false acceptance rate (FAR) or false match rate (FMR)	the probability to falsely accept a wrong guy in the biometric system, error rate often referred as the security level (also known as *false positive*).
false rejection rate (FRR) or false non-match rate (FNMR)	the probability to falsely reject an authorized user in the biometric system, error rate often referred as the convenience level (also known as *false negative*).

friction ridge	the ridges present on the skin of the fingers and toes, the palms and soles of the feet, which makes contact with an incident surface under normal touch. On the fingers, the unique patterns formed by the friction ridges make up fingerprints.
identification	the process of identifying an unknown biometric sample against a database of reference samples, also known as *one-to-n* or *one-to-many* recognition.
match / matching	the process of comparing a biometric sample against a previously stored template and scoring the level of similarity.
minutia (single) minutiae (plural)	friction ridge characteristics that are used to individualize a fingerprint. Minutiae occur at points where a single friction ridge deviates from an uninterrupted flow. Deviation may take the form of ending, bifurcation, or a more complicated type.
population	the set of end-users for the application.
reference	user's representative biometric sample captured at enrollment, to be stored in a database or personal token, and used for further authentication or identification against a candidate biometric sample.
template	representative compressed data extracted from a biometric sample.
verification	see "authentication".

SMART CARDS

automated teller machine (ATM)	a banking terminal that dispenses cash once the account identified with a smart card
chip card	another name for a smart card; refers to a plastic card with an embedded integrated circuit, which offers memory and micro-processing capabilities.
combi card	a smart card with both “contact” and “contactless” interfaces.
common criteria	framework for the independant evaluation of the security level of smart cards, defined by ITSEC organization.
contact smart card	a smart card that requires physical contact with a card reading device to exchange data.
contactless smart card	a smart card that transmits and receives data using radio frequency (RF) technology; does not require physical contact with a card reading device.
form factor	the physical device that contains the smart card chip. Smart chip-based devices can come in a variety of form factors, including plastic cards, key fobs, wristbands, wristwatches, PDAs, and mobile phones.
ICC	Integrated Circuit Card. ICC typically refers to a plastic (or other material) card containing an integrated circuit which is compatible to ISO/IEC 7816.
module	The contact and active part of the card, embedding the eletronic chip and reported onto the plastic card body.
multi-application smart card	a microprocessor smart card - typically with lots of memory and computing power - with more than one application residing on it.
single-application smart card	a smart card issued by a single organization for a singular purpose.

CRYPTOGRAPHY

authentication	the process of determining the precise identity of someone who conducts an online transaction or who sends an online communication.
certificate authorities	online enterprises that distribute and manage digital certificates, which are used to authenticate identity in an online environment.
cryptography	mechanisms and practices used to encode data for security purposes.
digital signature	a digital code attached to an online message that distinctly identifies the sender and confirms that a message has not been altered during transmission.
encryption	the scrambling of data for confidentiality purposes; a practice that allows only intended recipients to decode information.
hash algorithm	a software algorithm that computes a value (hash) from a particular data unit in a manner that enables detection of intentional/unauthorized or unintentional/accidental data modification by the recipient of the data.
message authentication code (MAC)	a short piece of information used to support authentication of a message. A MAC algorithm accepts as input a secret key and an arbitrary-length message to be authenticated, and outputs a MAC (sometimes known as a tag or checksum). The MAC value protects both a message´s integrity as well as its authenticity, by allowing verifiers (who also possess the secret key) to detect any changes to the message content. MACs are computed and verified with the same key, unlike digital signatures.
public key infrastructure (PKI)	a system that uses digital certification and certificate authorities to positively identify people and ensure trust in online transactions.

Notations

BIOMETRICS

IM	a biometric data image
IM_{ref}	the reference image
IM_{cand}	the candidate image
$TP = \{m_1, m_2, m_3, ..., m_n\}$	a template
TP_{ref}	the reference template
TP_{cand}	the candidate template
$Extract()$	the extraction algorithm
$Match()$	the matching algorithm
$m\ \{x,\ y,\ \theta,\ t\}$	a minutia point
T_h	a matching threshold

SMART CARDS

$MoC\ \{TP_{ref}\ ,\ TP_{cand}\ \}$	the Match-On-Card algorithm

CRYPTOGRAPHY

p, q	prime numbers
n	composite number (typically a *RSA* modulus)
$x \leftarrow_R X$	the element x pick at random in the set X
K	a Key
K_{priv}	a private Key
K_{pub}	a public Key
K_s	a secret Key
$a \| b$	concatenation of b with a

Abbreviations

1:1	One to One
1:N	One to Many
2TDEA	2-key Triple Data Encryption Algorithm (i.e. 2-key 3DES)
3DES	Triple DES
AES	Advanced Encryption Standard
AFIS	Automated Fingerprint Identification System
AFNOR	Association Française de Normalisation (France)
aka	also known as
ANSI	American National Standards Institute (USA)
ATM	Automated Teller Machine
b	bit(s)
B	Byte(s)
CBC	Cipher Block Chaining
CEN	Comité Européen de Normalisation (European Committee of Normalization)
CESTI	Centre d'Évaluation de la Sécurité des Technologies de l'Information
CLK	Clock
COS	Chip Operating System
CPU	Central Processing Unit

DES	Data Encryption Standard
DNA	DeoxyriboNucleic Acid
DUT	Device Under Test
DVB	Digital Video Broadcasting
EER	Equal Error Rate
EEPROM	Electrically Erasable Programmable Read Only Memory
ECC	European Citizen Card
ECC	Elliptic Curve Cryptography
EMV	Europay, Mastercard, Visa
EPROM	Erasable Programmable Read Only Memory
FAR	False Acceptance Rate
FIPS	Federal Information Processing Standard
FMR	False Match Rate
FNMR	False Non Match Rate
FRR	False Rejection Rate
FTA	Failure To Acquire
FTE	Failure To Enrol
GND	Ground (supply voltage -)
GSM	Global System for Mobile communications
I/O	Input/Output
I^2C	Inter Integrated Circuit bus
IAM	Identity & Access Management
IBE	Identity-Based Encryption
IBS	Identity-Based Signature

IC	Integrated Circuit
ICAO	International Civil Aviation Organization
ICC	Integrated Circuit Card
ID	Identity
ISO	International Standardization Organization
IT	Information Technology
ITSEC	Information Technology Security Evaluation Criteria
kB	kiloBytes
MAC	Message Authentication Code
MB	MegaBytes
MoC	Match-on-Card
MRTD	Machine Readable Travel Documents
NBS	National Bureau of Standards (USA)
NIST	National Institute of Standards and Technology (USA)
NPU	Numeric Processing Unit
OS	Operating System
PCB	Printed Circuit Board
PIN	Personal Identification Number
RAM	Random Access Memory
RDP	Relative Dielectric Permittivity
RF	RadioFrenquency
RFID	RadioFrenquency Identification
RSA	Rivest, Shamir, Adleman
RSA1024	RSA with 1024-bit key size

RSA2048	RSA with 2048-bit key size
RST	Reset
ROM	Read Only Memory
SAM	Security Access Module
SCOS	Smart Card Operating System
SHA	Secure Hash Algorithm
SIM	Subscriber Identity Module
TOE	Target Of Evaluation
TPM	Trusted Platform Module
User I/F	User InterFace
Vcc	Supply voltage (+)
VIP	Very Important Person
VM	Virtual Machine
Vpp	Programming voltage

Thesis Outline & Results Overview

Our Background

I come from the ***Smart Card*** industry. My primary activity has been the prototyping development of novel smart card concepts by the addition of various hardware components in a card plastic body, far beyond the sole integration of one smart card integrated circuit. With the original idea to enhance the user interaction with his smart card, I have integrated flexible displays, microbatteries, push buttons or additional electronic chips such as dedicated microcontrollers or flash memory. My research covers both hardware electronic design with mechanical constraints and very small *die electronic chips* (i.e. the silicon chip itself, without any packaging), and also firmware development in microcontrollers to drive these new input/output embedded devices.

In most systems, a smart card is a security element activated by the user when entering his Personal Identification Number (aka the PIN code). User authentication is a major concern and its convenience is a major issue. Hence with the advent of very small silicon-based fingerprint sensors, I naturally embedded one of the first sensor prototype within a smart card in order to propose a convenient alternative (or complement) to the PIN code. Beyond the technical (actually mechanical) challenge, this raised the computational challenge to our view: comparing digits is obvious, whereas comparing fingerprints is much more challenging.

This led to my interest in software and algorithmic issues: image processing, pattern recognition, data extraction, data matching, recognition error rates and so on. This is the field of ***Biometrics***: machine recognizable individual traits. Ten years ago, the research community in this field (aka *Biometricians*) was focused on robust data capture, image quality, extraction reliability or matching performance, without broader IT-security conscientiousness: privacy concerns, biometric template replay attack, faked biometric inputs and override of matching decisions were out of their scope.

In the smart card industry and any IT-security environment, security and privacy of data are the core business. Such a goal is achieved by using security tools from a toolbox called ***Cryptography***. Confidentiality, Integrity and Authentication of data are well-known subjects within this research community (aka *Cryptographers*) and issues such as man-in-the-middle attacks, impostor attacks, exhaustive searches or tamper-resistance are part of our business. Thus, I came to apply such tools to biometrics - on one hand proposing and testing security evaluation frameworks of biometric systems and, on the other hand, proposing and testing protocols to securely handle sensitive biometric data. My attention was then caught by quite paradoxical situations: (a) living humans versus dead machines, (b) the variability of biometric data versus the determinism of mathematical models in cryptography.

Scope Overview

This dissertation is about Biometrics - mostly Fingerprints - Smart Cards and Cryptography. Precisely, it is about security issues that are raised when we combine at least two of these three security toolboxes. My activities cover hardware electronic design, embedded software, algorithms and protocols.

Once upon a time there was convenience: humans simply needed to communicate, but then came security concerns: humans need privacy. We saw always the same story: it's all about trade-off between convenience and security.

The idea of inserting an electronic chip into a piece of plastic is nearly as old as modern cryptography and computer-based biometrics: about thirty-five years. But practically, massive application deployment started only twenty years ago with the advent of large information technology infrastructures for communications, banking transactions and the public internet.

Smart cards are perhaps some of the most widely used, yet often underestimated, electronic devices in use today. Credit cards, corporate badges, ID cards, ePassports are now popular but the technical background is still unknown to the general public. And this latter is a good point, this is a proof of maturity: only convenience without any technical concern for the end-user. However this lack of awareness is not a security argument, no security by secrecy. This is the reason why smart cards are using cryptography and physical tamper-resistance since the beginning of their young history. The very first (and straigthforward) application was the secure storage of cryptographic keys. Secondly, the ability to run cryptographic algorithms definitely differenciates smart cards from other passive security elements. Then came biometrics, and following the example of cryptography the smart card is not only a secure container but also an active security element able to process biometric data in order to better protect the user.

The main challenge with smart cards is the very limited computational resources in comparison to other embedded electronic platforms such as PDA or mobile phones. We must extract the best from the processor in a minimal amount of code lines, whence emerges the slogan **"tiny is beautiful"**. Many cryptographic primitives were originally designed for hardware implementation on 8-bit processors, at a time when these processors were the heart of high-end computers. This is therefore suitable to smart cards. In opposition, the automation of the biometric recognition came later on with more powerful computers in the nineties. We are used to say smart cards are equivalent to computers of twenty-five years ago: running biometric recognition (even from capture to decision) will be "business as usual" in 2015. Today we are only at the crossroad, this is the reason why my subject is particularly motivating.

I took a look at (the complexity of) biometric sensors, image sizes, image processing, template extraction, template storage, matching algorithms in order to determine suitability to smart cards and to propose eventually a novel approach from our smartcard-centric world. I was then caught up with our natural trend in security paranoia, studying potential flaws in the system, from the image capture to the final decision of the recognition process. By chance, the most deployed and mature biometric technology is fingerprint recognition and this has been also quickly identified to be the most suitable technique to smart cards. I took a look at flaws in fingerprint sensors, fingerprint templates, and data communication in order to alert our authorities and to discuss countermeasures.

Finally my most recent interest goes on one hand to the complex interaction between *error-free* cryptography and *error-prone* biometrics. On the other hand, there is also a complex interaction between humans and bioelectronics, along with extensive promises in biometrics and aliveness detection.

Thesis Outline

This dissertation is divided in five parts.

Part I - General Introduction

This first part is the introduction, covering all the three domains being at the intersection of our present work. I will depict here the basics and state-of-the-art of the different technologies, however focusing on details within my scope.

The first chapter presents biometrics: the different techniques, the system architecture, error rates, evaluation criteria and applications.

The second chapter presents smart cards and related technologies: hardware architecture, operating systems, applications. One section will be dedicated to so-called *multi-component smart cards* and my work in this area. The last section will briefly introduce the interaction between smart cards and biometrics.

The third chapter presents cryptography: general concepts, purposes, basic primitives and protocols. The last section will briefly introduce the interaction between cryptography and the previously described technologies: smart cards and biometrics.

Part II - Security Issues with Biometrics

This second part is dedicated to security issues with biometrics and its use in authentication systems.

The fourth chapter presents general security issues in the system architecture, describes different uses of biometric data within smart cards, and briefly compares biometrics with PIN codes.

The fifth chapter presents fingerprints in detail: classification, discriminant data, representative templates and existing standards.

The sixth chapter presents the state-of-the-art in fingerprint sensor technologies: optical technologies, silicon-based technologies, different form factors and trends for the future.

The seventh chapter presents my work on fake finger issues: fingerprint copy techniques, materials, sensors' weaknesses and reproducible attacks.

The eighth chapter presents the state-of-the-art and my work in digital attacks on fingerprint templates and images: synthetic generation of template and images, recovering original image information from a template and generating a synthetic, fingerprint-looking, image around real and fixed minutiae.

The ninth chapter presents the state-of-the-art in aliveness detection within biometric sensing systems and my work in bypassing them: different approaches, different techniques, intrinsic protection by sensor technology, circumvention experiments or suggestions to bypass upon our experience.

The tenth chapter concludes this second part by proposing evaluation methodologies to build a certification framework for the security level of fingerprint-based recognition systems.

Part III - Biometric Algorithms for Smart Cards

This third part is about implementing biometric algorithms in smart cards, and how to evaluate their reliability and suitability to real world applications.

The eleventh chapter presents my work about Match-on-Card algorithms: previous related works, my approach, implementation and testing.

The twelfth chapter presents the de-facto standard (NIST's MINEX) to evaluate fingerprint matching algorithms and results: PC-based solutions, Match-on-Card solutions and results obtained here with one particular algorithm using my basis as shown in the previous chapter.

Part IV - Security Protocols for Smart Cards with Biometrics

This fourth part covers security protocols for smart cards with biometrics by two examples: my proposal for a simple and cost-effective approach and my work to enhance the security and the convenience of one particular corporate badge, the U.S. government PIV card.

The thirteenth chapter presents a very simple protocol with light cryptography to target cost-effective systems and low-end smart card chips. The aim is to protect from both malicious terminals and also malicious smart cards by avoiding replay attacks, yes-cards, no-cards and exhaustive search.

The fourteenth chapter presents the Personal Identity Verification (PIV) card used by workers for the US government and my related work within the Secure Biometric Match-on-Card (SBMOC) initiative. SBMOC was a testing program by the US NIST to assess the feasibility of a convenient PIV card with the use of contactless smart cards, Match-on-Card and strong cryptography - without compromising security.

Part V - Biometrics & Cryptography Interaction

This fifth part discusses the paradoxical interaction between biometrics and cryptography: how to secure noisy data, how to compare in the encrypted domain, how to revoke templates. I will depict here our approach of a challenge-response protocol based on biometric data and how smart cards could help to build practical crypto-biometric systems.

The fifteenth chapter presents the state-of-the-art in mixing cryptography and biometrics: ideas, stakes, concepts and applications. Concepts such as fuzzy extraction, biometric hashing, and cancelable biometrics are covered.

The sixteenth chapter presents my biometrics-based challenge-response protocol and its implementation to both secure and ease the recognition process within a smart card. This efficient alternative to Match-on-Card is fully detailed and compared to the classical matching approach.

Finally, the seventeenth and last chapter presents my ideas to build practical crypto-biometric systems with the help of smart cards and discusses some simulated solutions that I have evaluated.

Discussion about the Outline

The second part of this dissertation describes the two last years of work within CEA-Leti and Ecoles des Mines on funded projects. These projects gave me the opportunity to obtain resources and access to state-of-the-art materials in fingerprint sensing and electronic circuit manufacturing. All this work was necessary to propose relevant evaluation methodologies for the mandatory, at medium term, certification of the security level of fingerprint recognition systems from the image capture to the matching decision.

All other parts of this work were conducted within the research & security department of Gemplus (and Gemalto, after the merger between Gemplus and Axalto), the world leader in smart cards and digital security. This represents punctual projects within my technology survey activities about the future of smart cards. Here there is no particular link between the different projects, apart from using biometrics and cryptography: multi-component smart cards (especially with fingerprint sensors), Match-on-Card algorithms, secure protocols for biometrics and cryptographic approach of biometrics.

Results Overview

First of all, since 1998 I was among the first to demonstrate the feasibility of screen cards, high memory cards and fingerprint cards during worldwide events such as *Cardtech-Securtech'2001* in the US and the annual show *Cartes'2001* in France. My work presented in chapter 2, section 2.5 regarding multi-component smart cards and technology trends with USB communication, MEMS integration and so on has been published in Computer Network journal under the name "From smart cards to smart objects: the road to new smart technologies" [89].

My work about security issues with fingerprint recognition systems presented in part II has delivered a dozen technical reports and one publication under the name "Fake Fingers in Fingerprint Recognition: Glycerin supersedes Gelatin" [9] where I disclose the power of glycerin to solve durability issues with state-of-the-art gelatin-made fake fingers. In this area I also demonstrated the weaknesses of aliveness detection systems by circumventing available systems and proposing ways to bypass other "published" systems I did not obtain. I proved the weaknesses of fingerprint templates and the ability to conduct brute-force and masquerade attacks to cheat with the recognition process. All my experience in this work results in the definition of a certification framework for the security evaluation of fingerprint recognition systems, as the one already existing for smart cards.

My work with Match-on-Card algorithms, described in part III of this dissertation, serves as basis for an algorithm competing at the annual contest conducted by the US NIST. My support of NIST in defining the evaluation framework for match-on-card and the results I obtained with my algorithm are published in two NIST Interagency reports. My algorithm and its implementation on smart card enters the world of professional solutions, just behind the leaders. There is still a lot of room left for improvements.

Regarding security protocols for smart cards with biometrics, half of my work in part IV has been published under the name "A Protection Scheme for MOC-Enabled Smart Cards" [10]. This paper discusses the ability to build security even with low-end and cost-effective smart cards. The second half of my work is published in a NIST Interagency report and proves the feasibility of a contactless smart card processing match-on-card and demanding cryptography within seconds.

Described in the last part (V) of this dissertation, my cryptographic approach of fingerprint matching within a smart card has been published under the name "Externalized fingerprint matching" [8] in reference to the ability to securely send complex tasks for pre-processing to a powerful, but potentially malicious, terminal. Receiving the result of this pre-processing, the smart card will process the easy and final step of the recognition with the help of an internal secret. I prove here the feasibility to emulate a match-on-card feature even with a simple memory card, by an interaction between cryptographic primitives and biometric data.

Conclusion

The wide scope of my work helped in maintaining high motivation by constantly proposing new technical challenges in different research areas. I also had the chance to always find "customers" to support my ideas and vision within the scope of smart cards, biometrics and cryptography. After many years on the subject and a lot of time spent in writing this dissertation, I still believe in the power of biometrics, but only if added to other security tools.

PART I

General Introduction

CHAPTER 1
Introduction to Biometrics

Contents

1.1 A few words about Biometrics

We may refer the reader to [62, 128, 132] for a complete overview of Biometrics.

1.1.1 A little bit of History

Handprints and *Footprints* are classical petroglyphs found in prehistoric caves. Moreover, these pretroglyphs are found in each regions: African caves, North-American caves, European caves and Australian caves. Later on, fingerprints were used on clay tablets in ancient Babylonia, thumbprints were used on clay seals in ancient China, fingerprints were used on official documents in 14th century Persia.

Modern history of *Anthropometry* began around 1882 with Alphonse Bertillon, a clerk in the prefecture of police in Paris, using measurements of parts of the body. Bertillon's system included measurements such as head length, head width, length of the middle finger, length of the left foot; and length of the forearm from the elbow to the tip of the middle finger. This may have been inspired by the well-known Da Vinci drawing "The Vitruvian Man", which depicts a nude male figure in two superimposed positions with his arms and legs apart and simultaneously inscribed in a circle and square.

Few years later, Sir Francis Galton, a British anthropologist, published a detailed statistical model of fingerprint analysis and identification and encouraged its use in forensic science. Galton identified the characteristics by which fingerprints can be identified. These same characteristics (minutiae) are basically still in use today, and are often referred to as Galton's Details.

Two years later, Sir Edward Henry, Inspector General of the Bengal Police in India became interested in the use of fingerprints for the use of criminal identification. He ordered the Bengali Police to collect prisoners' fingerprints in addition to their anthropometric measurements. Expanding on Sir Galton's classification system, Sir Henry developed the Henry Classification System between the years 1896 to 1897. The Henry Classification System was to find worldwide acceptance within a few years.

In 1918, Edmond Locard, a French pioneer in forensic science, devised the "12 points" rule; the rule stating that only twelve matching Galton's details are enough to surely identify or authenticate a criminal.

More recently, the need for user authentication and user identification in Information Technology (IT) world seems to date back to the late sixties [56], and the idea to use fingerprints was already there [129, 130].

1.1.2 A little bit of Science Fiction

Beyond modern fingerprinting in forensic science, recent media such as literature and cinema expand the notion of biometrics in civil environment. Regarding literature, Georges Simenon, a well-known French detective writer, is known to have attended some Locard's lectures in 1919 or 1920, hence his extensive use of fingerprinting in his novels. Dozens of movies and TV series such as James Bond, Impossible Mission, Star Trek use extensively user authentication by biometric means. In 1971, James Bond's "Diamonds are Forever" shows fingerprint recognition and fake fingerprint technique. In 1997, Andrew Niccol's "Gattaca" uses DNA identification from either blood sample, bodyhair sample or urine sample to trace everyone, like the "Big Brother" of "1984", exploiting a common fear about biometrics in real life, and its nasty exploitation. In 2002, Steven Speilberg's "Minority Report" uses on-the-move iris recognition at distance, a much hyped and futurist technology for this time...

We may refer the reader to [71] for an impressive list of movies with detailed biometrics use in each one.

1.2 Biometric Modalities

1.2.1 Introduction

To prove our identity, we can use three ways:

1. Something we have (e.g. a Smart Card)
2. Something we know (e.g. a PIN code, a Password)
3. Something we are (Biometrics, e.g. Fingerprint, Face, Iris)

In everyday life, we usually give our trust to a combination of *something-we-have* and *something-we-know* (e.g. banking cards, SIM card in mobile phones) but a password can be communicated or guessed and a personal device can be lost or borrowed. Building a three-factor authentication with the addition of one or several biometric techniques brings high confidence in our authenticated interlocutor and provides non-repudiation.

A general definition of Biometrics could be:

"Biometrics allow a person to be authenticated or identified using behavioral or physiological characteristics. These characteristics must be automatically recognizable and verifiable".

The biometric authentication has the advantage of checking the user's personal characteristics. These characteristics can be physical ones such as fingerprints, face, iris or behavioral ones such as voice, handwritten signature, keyboard tapping.
This leads to a possible split in the usually called something-we-are, see figure 1.1:

1. Something we are (physical Biometrics)
2. Something we do (behavioral Biometrics)

Behavioral characteristics are much less stable than physical characteristics because of their poor resistance to user's stress or health troubles. The authentication process is a comparison between a pre-registered reference image, or template (representative data extracted from the raw image, built during an *enrollment* step) and a newly captured candidate image, or template. Depending on the correlation between these two samples, the algorithm will determine if the applicant is accepted or rejected. This statistical process leads to a False Acceptance Rate (FAR, i.e. the probability to accept a non-authorized user) and a False Rejection Rate (FRR, i.e. the probability to reject an authorized user).

1.2.2 Fingerprint Recognition

We may refer the reader to [74] for a complete overview of Fingerprint Recognition.

Fingerprint recognition is based on the imaging of the fingertips. The structure of a fingerprint's ridges and valleys is recorded as an image or digital template (a simplified data format, minutiae-based most of the time) to be further compared with other images or templates for authentication or verification, see figure 1.2. Images of fingertips are captured with specific fingerprint sensors. Among all the biometric

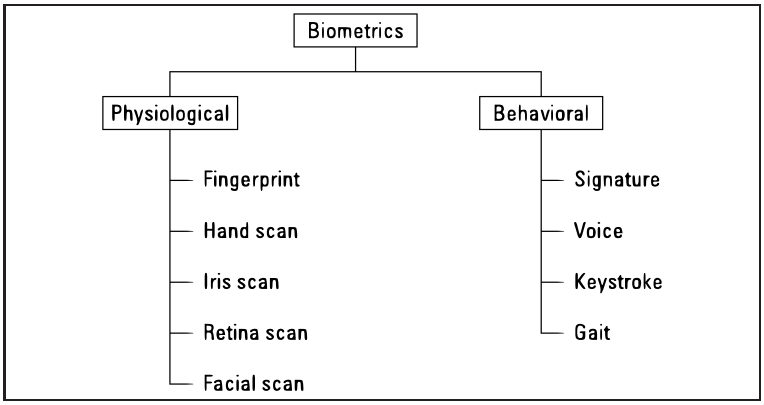


Figure 1.1: Two Biometric Families

techniques, fingerprint-based identification is the oldest method which has been successfully used in numerous applications for over a century, more recently becoming automated due to advancements in computing capabilities. Fingerprint identification is popular because of the inherent ease in acquisition, the numerous sources (ten fingers) available for collection, and their established use and collections by law enforcement and immigration. This is the second and optional biometrics to be used in ePassport, however mandatory in Europe in mid'09.

The reader will find more detailed information about fingerprint structure in chapter 5.

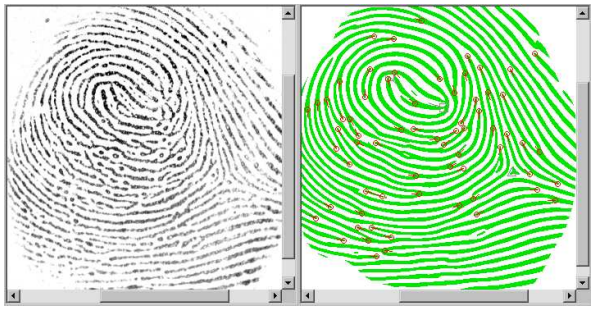


Figure 1.2: Fingerprint Recognition

1.2.3 Face Recognition

We may refer the reader to [69] for a complete overview of Face Recognition.

Face recognition is based on the imaging of the face. Structure of the face is recorded as an image or digital template (there is a plenty, however non-mature, of simplified data formats) for further comparison. Early face recognition algorithms used simple geometric models, see figure 1.3, but the recognition process has now moved into a science of sophiticated mathematical representations and matching processes. Major advancements and initiatives in the past ten years have propelled this technology into the spotlight. This is the most intuitive biometrics since everyone is using it to recognize their human friends, and used for a long time in identity documents. This is the first and mandatory biometrics to be used in ePassports.

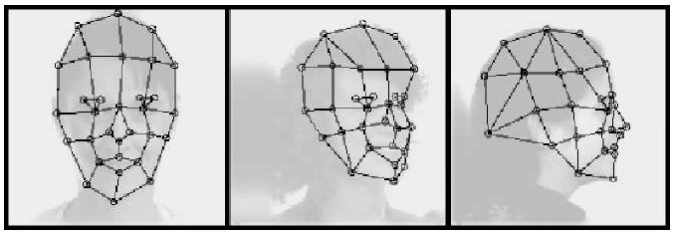


Figure 1.3: Face Recognition

1.2.4 Hand Geometry

Hand geometry recognition is the longest implemented biometric type, debuting in the market in the late Eighties. One of the shortcomings of the hand geometry characteristic is that it is not highly unique, limiting the applications to verification tasks only. The devices use a simple concept of measuring and recording the length, width, thickness, and surface area of an individual's hand while guided on a plate. Hand geometry systems use a camera to capture a silhouette image of the hand. The image captures both top surface of the hand and a side image that is captured using an angled mirror, see figure 1.4. The template is stored in nine bytes of data, an extremely low number compared to the storage needs of other biometric systems.

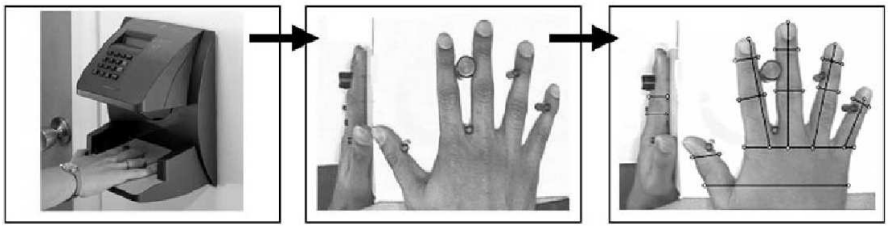


Figure 1.4: Hand Recognition

1.2.5 Iris/Retina Recognition

User authentication based on the eye splits in two families: 1- Iris recognition is based on the extraction of representative data from the externally visible colored ring around the pupil, whereas 2- Retina recognition is based on the analysis of the blood vessel pattern located in the posterior portion of the eye, see figure 1.5. The automated method of iris recognition is relatively young, existing in patent only since 1994. The iris is a muscle within the eye that regulates the size of the pupil, controlling the amount of light that enters the eye. The color is based on the amount of melatonin pigment within the muscle. Iris imaging requires use of a high quality digital camera. Today's commercial iris camera typically use near-infrared light to illuminate the iris without causing harm or discomfort to the subject. Iris recognition is the third and optional biometrics to be used in ePassports.

1.2.6 Vein Pattern Recognition

The prominence and acceptance of biometric technologies such as fingerprints, facial recognition, hand geometry, and iris recognition may leave little demand for other modalities. However, the emerging vein-

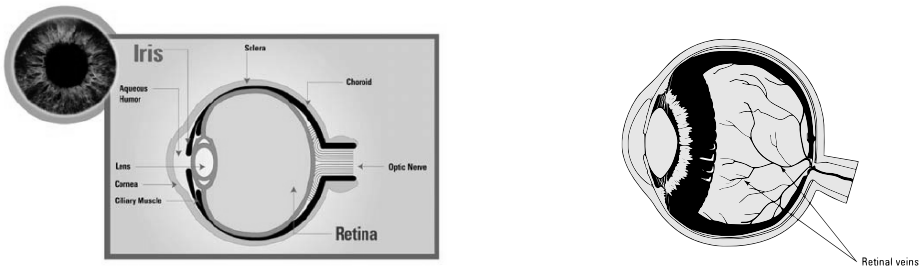


Figure 1.5: Iris & Retina

pattern recognition technology, with its own unique features and advantages, has maintained its position against the others.

Vein pattern recognition is gaining momentum as one of the fastest-growing technologies. It is on course to become the newest entrant to mainstream biometric technologies, moving from the research labs to commercial deployment.

The technology works by identifying the subcutaneous (beneath the skin) vein patterns in an individual's hand, wrist or finger. When a user's hand is placed on a scanner, a near-infrared light maps the location of the veins. The red blood cells present in the veins absorb the rays and show up on the map as black lines, whereas the remaining hand structure shows up as white. After the vein template is extracted, it is compared with previously stored patterns and a match is made, see figure 1.6. Currently being a niche in Japan, pushed by many Japanese industries, with no real scientific basement up to now, vein recognition is however emerging in research programs worldwide and is proving to give acceptable results [67].

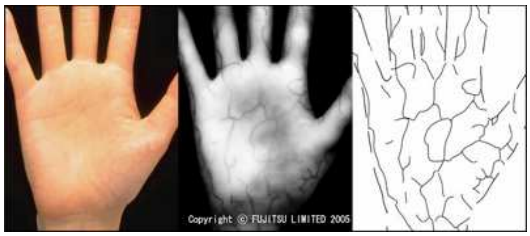


Figure 1.6: Vein Pattern Recognition

1.2.7 Other Techniques

Each human is unique in many ways. Thus, the number of possible physical characteristics or personal trait s of a human is only limited by our imagination and our ability to measure the characteristic or trait. Among other biometric techniques, few may appear classical such as *voice recognition*, *handwritten signature*, see figure 1.7, *DNA*, whereas several may appear esoteric such as *facial thermograms*, *handgrip dynamics*, *gait*, *body odors*, *ear shape*. The most recent techniques cover *biodynamic signature*, *otoacoustic emissions*, *brainwave pattern*.

A large, however non-exhaustive, list of known biometric techniques is listed in table 1.1

Figure 1.7: Other biometrics: signature, handgrip, gait, ear

Physiological	**Behavioural**
Fingerprint	Handwritten signature
Face	Keystroke dynamics
Iris	Gait
Retina	Handgrip dynamics
Voice	Voice
Vein pattern	Lips dynamics
Palmprint	Mouse dynamics
Hand geometry	
DNA	
Facial thermograms	
Body odor	
Fingernail bed	
Brainwave pattern	
Biodynamic signature	
Otoacoustic emissions	
Ear shape	
Skin spectrography	

Table 1.1: Known Biometrics

1.2.8 Multimodal Biometric Systems

Multimodal biometric systems take input from single or multiple sensors that capture two or more different modalities of biometric characteristics. For example, a single system combining face and iris information for biometric recognition would be considered a "multimodal" system regardless of whether face and iris images were captured by different imaging devices or the same device. It is not required that the various measures be mathematically combined in anyway. For example, a system with fingerprint and voice recognition would be considered "multimodal" even if the "OR" rule was being applied, allowing users to be verified using either of the modalities. Basically, the "OR" rule will allow to cover a larger population (e.g. user can't provide fingerprints) and the "AND" rule will allow higher security, users being authenticated by both modalities. Despite a lot a technical advantages, using multimodal biometrics is costly and less user-friendly.

Some combinations appear more natural than others: palmprint + fingerprint + hand geometry, face + voice + lip movement, face + iris.

1.3 Biometric Systems Architecture

1.3.1 General Architecture

The general architecture of a biometric system is depicted in figure 1.8. Basic components of the system are data acquisition (*capture* sensors), transmission channel, signal processing (*extract* and *compare* algorithms), data storage (server database, smartcards) & decision policy.

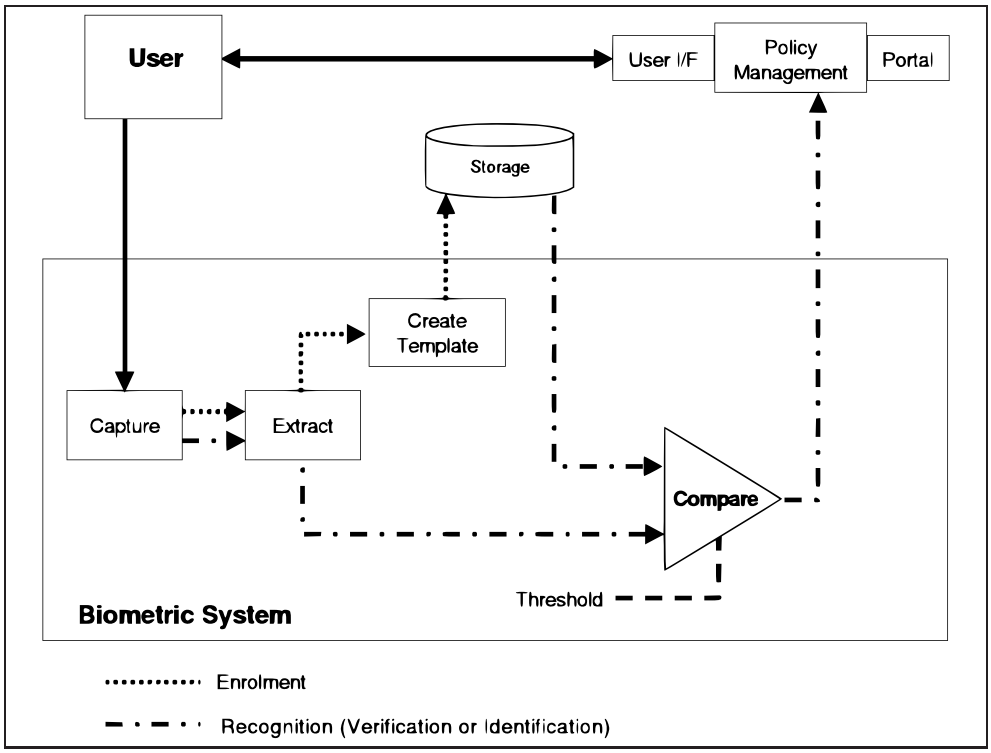


Figure 1.8: General Architecture of a biometric system

Briefly, the system stores a *reference* data of the user, generated at *enrollment*, to be compared to the newly captured *candidate* data at *verification/authentication* or *identification*. Depending on the *decision* (i.e. match/fail or thresholding of the given score), the user will gain access, or not, to the system.

1.3.2 Extraction Algorithm

The so-called *extraction* algorithm processes the original biometric input signal to extract strong repeatable features to build a template. The purpose of using this approach is to save storage space and communication bandwith by a lossy, however efficient, compression. Regarding fingerprints, a bitmap image of about 100 kBytes (about 12kB after lossless compression, used in ePassports) may be represented by a minutiae template of about 250 bytes. Figure 1.9 depicts the classical image processing applied to a fingerprint image to extract minutiae.

Basically, extraction is a function with an image as input and providing a template as output:

$$TP_{ref} = Extract\ (IM_{ref}\ )$$

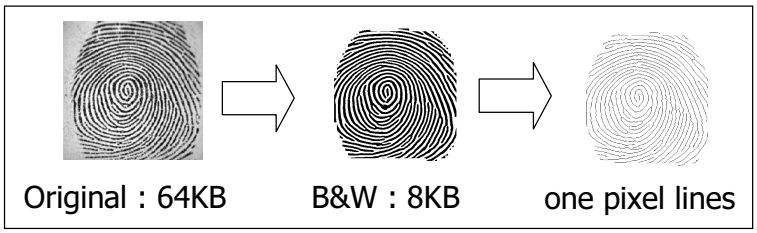


Figure 1.9: Processing Fingerprint Bitmap

1.3.3 Matching Algorithm

The so-called *matching* algorithm compares two templates to determine whether they are from the same biometric source or not. Mathematical transformations are applied to a candidate template to evaluate the *distance* from a reference template. Depending on this distance and a *threshold*, a pass/fail decision is made. Figure 1.10 illustrates the minutiae matching between two fingerprints.

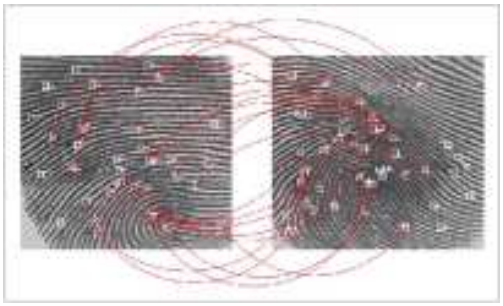


Figure 1.10: Minutiae-based Matching

Basically, matching is a function with two templates as input and providing a decision, or a score, as output:

$$Score = Match\ (TP_{cand}\ ,\ TP_{ref}\ )$$

For security reasons, the typical output is only the decision; the score is used only internally for comparison to the threshold.

1.3.4 Enrollment

Enrollment is the registration of an authorized user into the system. This is a one-shot process, often considered as the critical, and costly, part of the system. The quality of the generated reference template at enrollment will determine the efficiency of the system during its life cycle. In most (security-oriented) applications, the registration needs the presence of the user in an authority's dedicated booth, needs time to train the user to capture a good image and needs some post-processing to confirm the quality of the sample (running an authentication) or verifying non-duplication in a database. Usually, we consider the enrollment to be done in a secure environment, limiting possible attacks in this phase, whereas *self-enrollment* is only considered for convenience-oriented applications. Figure 1.11 depicts the enrollment purpose: generate and store a reference template from each user of the system.

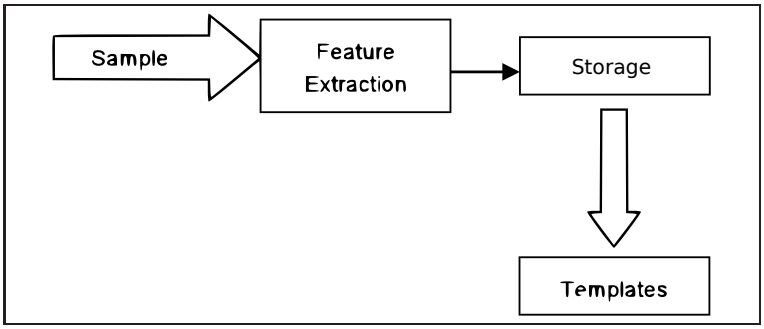


Figure 1.11: System Architecture: Enrollment

1.3.5 Authentication/Verification

Authentication (or Verification) is the process of comparing a known biometric sample to a known reference identified in a database or a personal token to confirm the identity of the claimant. This is a *one-to-one* process (1:1), and is usually used in identity documents such as corporate badges, National ID cards or ePassports, to prove that the carrier is the authorized owner of the document. Figure 1.12 depicts the authentication architecture, whereas figure 1.13 depicts the authentication architecture using multimodal (here *bi*modal) approach.

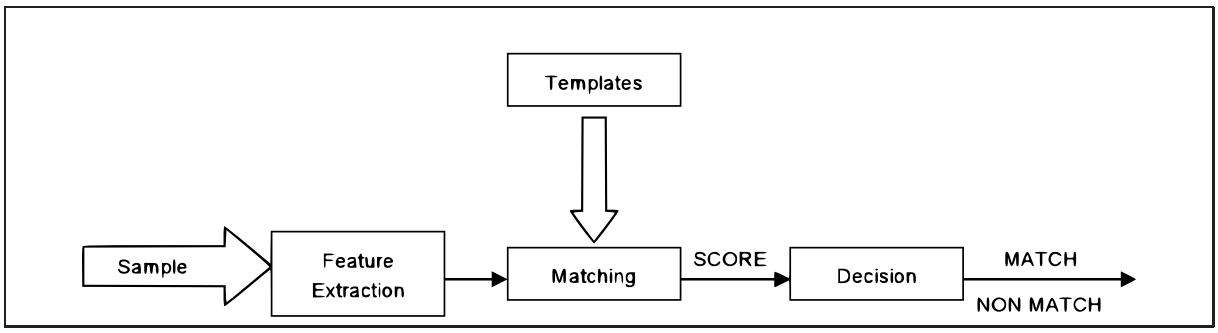


Figure 1.12: System Architecture: Matching

With **authentication**, the biometric system tries to answer the question **"Is this X?"**.

1.3.6 Identification

Identification is the process of comparing an unknown biometric sample to multiple references in a database to find out the identity of the owner of this unknown sample. This is a *one-to-many* process (1:N), usually used in criminal sciences (e.g. AFIS -Automated Fingerprint Identification System-). Civilian AFIS may be used for both positive or negative authentication. Positive identification refers to proving the membership of a group (whitelist, e.g. VIP access to a secure area), whereas negative authentication refers to proving the non-membership of a group (blacklist, e.g. pathological gamblers shortlist used in casinos). Identification is often used at enrollment, even for authentication-oriented systems, to find out duplication of identities before registering the new user.

With **identification**, the biometric system tries to answer the question **"Who is X?"**.

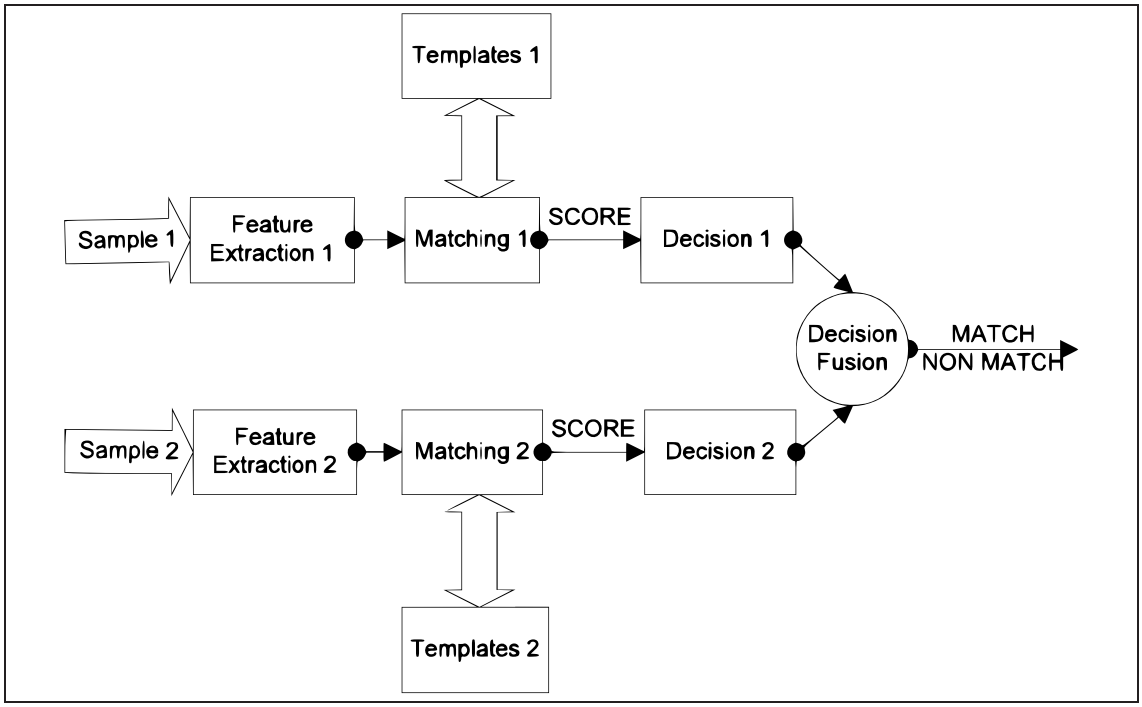


Figure 1.13: Multimodal Decision-level Fusion

1.3.7 Authentication vs Identification

In Biometrics language, *authentication* (or *verification*) refers to a one-to-one process (i.e. the user has to prove who he's claiming to be), whereas *identification* refers to a one-to-many process (i.e. the system has to find out who is the owner of the biometric candidate sample by comparison with a large database of biometric reference samples). We must make here a distinction in the meaning of these words (identification and authentication) between Biometrics and IT: in IT, the identification (login name) always precedes the authentication (password), that is to say we first claim who we are and then we prove our statement. An IT system never identifies a user with his password, whereas in biometrics the same sample may serve the purpose of authentication or identification.

1.4 Biometric Systems Errors

The authentication process is a comparison between a pre-registered reference image, or template[‡], (built during an *enrollment* step) and a newly captured candidate image, or template. Depending on the correlation between these two samples, the algorithm will determine if the applicant is accepted or rejected. This statistical process leads to a False Acceptance Rate (FAR, i.e. the probability to accept a non-authorized user) and a False Rejection Rate (FRR, i.e. the probability to reject an authorized user). Let's say that a low FAR represents security and low FRR represents user convenience: a system with a very low FAR, hence a high FRR, remains perfectly secure since the authorized user himself can't use it!

[‡]Representative data extracted from the raw image

Depending on the application, we have to focus our efforts on FAR or FRR, let's take two opposed examples:

- Very secure access to a restricted area where we do not want to take the risk that a bad guy get in, even if an authorized user will need to apply twice or more: this is low FAR.
- Forensic applications where we need to identify the bad guy, even if in a first pass we will identify multiple suspects and refine our investigations later on: this is low FRR.

Another metric that can be read in the literature is EER (Equal Error Rate, point where FAR=FRR), this is interesting to benchmark different biometric systems, but is definitively not a good choice of FAR vs FRR trade-off in the real world since any well-studied application will for sure need a focus on either FAR or FRR. See figure 1.14.

$$FAR = \frac{FA}{N} \quad where\ FA\ =\ number\ of\ false\ acceptance,\ and\ N\ =\ total\ (large)\ number\ of\ samples$$

$$FRR = \frac{FR}{N} \quad where\ FR\ =\ number\ of\ false\ rejection,\ and\ N\ =\ total\ (large)\ number\ of\ samples$$

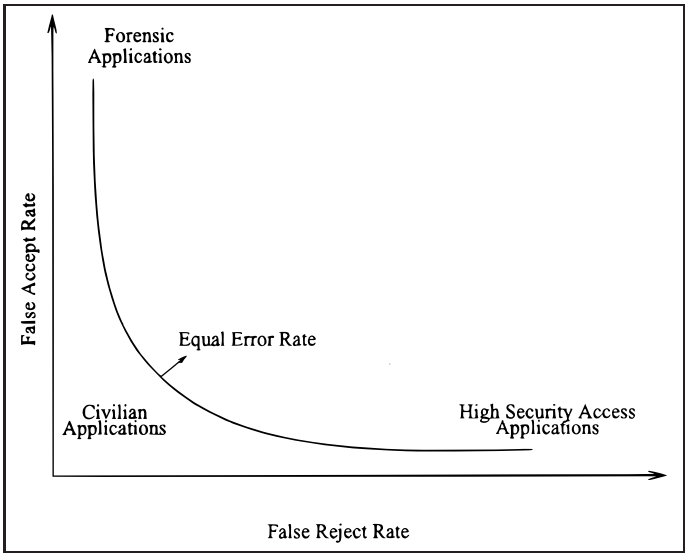


Figure 1.14: Error Rates

Other measurable error rates are Failure to Enrol (FTE) and Failure to Acquire (FTA). FTA is generally considered as a subset of FTE. Depending on the minimun required quality of the image to ensure the good functioning of the biometric system, images could be rejected before trying to extract features from it, this enters in FTA rate.

If a good image is captured, depending on the minimum required quality/number of extractable features to ensure the good functioning of the biometric system, generated templates could be rejected before storage as a reference or submission to a matching module. This is FTE.

Obviously this FTE is closely linked with FAR and FRR. One could design a very good system in terms of FAR/FRR if he rejects each bad image or poor template, hence having a very high FTE. The goal of each biometric system is to be usable by the largest targeted *population*, hence the maximum acceptable value for FTE is generally considered of about 1%. With this value fixed, we are now able to measure relevant values of FAR and FRR for the given biometric system.

1.5 Biometric Systems Evaluation

Main criteria to be taken into account when considering a biometric system are listed in table 1.2.

1	Universality	Can anyone provide the considered biometrics?
2	Uniqueness	How hard is it to find two persons with close characteristics?
3	Permanence	Stability along the lifetime
4	Collectability	How easy/cheap is it to capture the biometric data?
5	Performance	Which FAR/FRR can the system provide?
6	Acceptability	How well will end-users appreciate the system?
7	Circumvention	How hard is it to fool the system?

Table 1.2: Seven pillars of biometrics

1. Universality: a few users would be unable to speak or to provide fingerprints for some reasons, whereas everyone has a face.

2. Uniqueness: Fingerprints and iris have proven good results, whereas facial recognition suffers from twins and look-alikes.

3. Permanence: Fingerprints and iris have proven good stability, whereas voice and face features evolve quite a lot.

4. Collectability: face recognition may need a simple webcam, whereas fingerprints and iris need very specialized devices, not so inexpensive. DNA analysis is another example of complex collectability, whereas it involves a biochemical process that is, as of today, too expensive to be used in IT systems.

5. Performance: Fingerprints and iris have proven good FAR/FRR levels, whereas other techniques are not so mature.

6. Acceptability: THE most important criteria to obtain end-users support. Depending on cultural issues, fingerprints are associated with criminals or touching a public device is not healthy (SRAS disease in Asia, transmitted by physical contact). An iris scanner is too intrusive, diseases and drugs can be detected in eye examination. Using faces for identification is used in every culture, people are accustomed to photographs on ID cards, passports, application forms. This is one of the main reasons why, together with universality and collectability, only facial recognition is currently mandatory in ICAO specifications [53].

7. Circumvention: systems with high FAR will be more easy to attack at matching level, hand geometry with the entropy of a nine bytes template is prone to brute-force attack, and most fingerprint imaging systems do not implement aliveness detection and could be fooled with fake fingers [76].

1.6 Biometric Systems Management

The better quality the reference template, the better chance of correctly matching against a live sample. Depending on the lifecycle of a biometric system, it may be necessary to sometimes update the reference data in order to take into account drifts on the input biometric source. One approach is to update yearly the reference data by replacing it with the last matching sample. However, this must be done with caution, since a false acceptance at this stage would definitely unsettle the system. An approach like *enrollment with generalization* (i.e. capture multiple samples and use statistics to create the reference template) should be used, but would no longer be transparent to the end-user [5].

Automated Fingerprint Identification Systems (AFIS) are examples of very large and complex IT systems. Scaling the system is a major issue: total number of managed references, numbers of enrollments per day that increase the database, number of identification requests per second to handle, and so on must be fully anticipated [5, 128]. For instance, the European Visa Information System (VIS) will handle more than seventy million records.

The system management must also handle:

- fallback procedures in case of repeated false rejection, failure to enroll or users refusing the system.
- safety procedures to avoid influenza, or other diseases spread by touch-based biometric stations.
- an information plan to educate and train the users.
- software, hardware, firmware upgrades
- security procedures to avoid attacks and private information leakage

1.7 Privacy Issues

We usually claim biometric data are public to highlight the fact that they are not secret: everyone leaves his fingerprints everywhere, everyone face is visible to others. Privacy-concerned organizations are sensible to so-called biometrics with no trace (e.g vein pattern: no physiological trace, not visible by eye), but the digital trace exists just after the data capture. Actually we should consider biometric data as being private (and privacy is totally different from secrecy on a security standpoint) since they may leak some non-public information about the user [88]. Health information, racial information, psychologic profile are examples of discussed issues; a few may appear more like “urban legends” than real-life issues. However, even if fictional, they nevertheless affect general public acceptance. This information may be either captured with an (hidden) additional feature during the biometric sensing (e.g. temperature measurement) or extracted at latter stage from the captured image (e.g. analysing a retinal scan). Data protection schemes (e.g. managed databases, personal tokens) and rules (e.g the European Personal Data Directive [123]) exist to define who has access to certain information, and for which appropriate use.

Enrollees may be concerned by the misuse of their personal data: using face information in surveillance systems or comparing fingerprint information, originally dedicated to civil purposes, against forensic databases. This privacy issue is the use of the same biometric data for different applications: if one may obtain the reference biometric data from an insufficiently secure system, he may use it to fool other systems using the same reference by so-called replay attacks. This threat, and countermeasures, will be discussed in parts II and V. Another issue is the lack of aliveness detection in biometric systems. There exist several stories of criminals using severed fingers to steal fingerprint-protected cars or to enter bank gates [71](Myths & Reality section). It is important to note that taking into account privacy concerns of the general public will be decisive for the acceptance of biometric systems.

1.8 Applications

There are several key reasons why biometrics are becoming increasingly popular. Basically, biometrics are used for either security or convenience. However any biometric application is always a trade-off between convenience and security.

Biometrics are used for **convenient authentication** in physical or logical access control, avoiding badges or passwords: there is nothing to lose or forget.

Regarding security, **increased need for strong authentication** paves the way to biometrics: no lost identifiers, can't be stolen as easily as tokens. In combination with smart cards and PIN codes, this proves the "physical link" with the user.

Decreasing cost of the technology, both on sensors side and processing chips (running algorithms), makes biometrics more affordable.

Increased government and industry adoption, such as biometric ePassports, eVisas, national eID cards or banking applications, puts biometrics under the spotlight to the general public. As an outgrowth of the September 11, 2001, terrorist attacks, an increased awareness of physical security and public safety has also helped make biometrics attractive.

Information Technology (IT) apart, biometrics are used for **person identification** applications, such as forensic sciences or legal medicine.

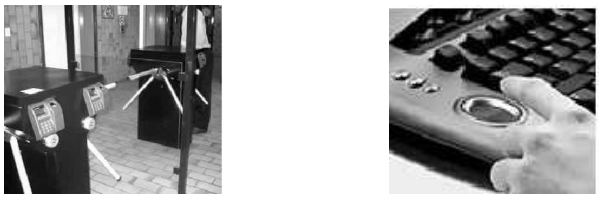


Figure 1.15: Physical and logical access control

Figure 1.16: Government applications

Figure 1.17: Forensic identification applications

CHAPTER 2
Introduction to Smart Cards

Contents

2.1 A few words about Smart Cards

We may refer the reader to [78, 90] for a complete overview of Smart Cards.

2.1.1 A little bit of History

The idea to use a memory electronic chip in a credit card appeared in the late sixties from the United States, Japan, Germany and France, but with no exploitation of these patents.

Roland Moreno, a science reporter fond of electronics, patented his first concept of the memory card in 1974 (the original prototype having the form factor of a ring for payment in shops). In 1977, Michel Ugon, engineer at Bull, invented the first microprocessor smart card.

The first mass use of the memory cards was for payment in French pay phones, starting in 1983 (France Télécom's Télécarte). The second mass use of smart cards (however the first use of microprocessor cards) was with the integration of microchips into all French debit cards (Carte Bleue) completed in 1992.

The major boom in smart card use came in the 1990s, with the introduction of the smart-card-based SIM used in GSM mobile phone equipment in Europe. Due to the ubiquity of mobile phones in Europe, smart cards have become very common.

Smart cards are also being introduced in personal identification and entitlement schemes at regional, national, and international levels. Citizen cards, drivers' licenses, and health card schemes are becoming more prevalent. Contactless technology currently becomes widespread in the form of ePassports.

Regarding the industry, the very first "CP8" based on Bull's patent was produced by Motorola. In 2001, Bull sold its CP8 Division together with all its patents to Schlumberger. Subsequently, Schlumberger combined its smart card department and CP8 and created Axalto. In 2006, Axalto and Gemplus, at the time the world's no.2 and no.1 smart card manufacturers, merged and became Gemalto.

2.1.2 A little bit of Science Fiction

A well-known inspiration for smart card invention is René Barjavel's "The Ice People" (French title: "La Nuit des Temps") published in 1968. This novel starts with a scientific expedition in Antarctica, discovering in ice a suspended animation chamber from which scientists awoke a frozen woman, Eléa. Telling her story, she depicts a very advanced civilization, about 900.000 years ago, with only two continents, Gondawa and Enisoraï, being at war and finally destroying current life on Earth. In that past time, every person was wearing a very useful gold ring on the right middle finger for authentication and physical access (key)...

In 1997, Luc Besson's "The Fifth Element" shows a futurist world with the Multi-Pass, a combination of multiapplication card and its active plastic case (authentication, transactions) with two photographs, one being the face of the owner, the second being the same face in another color domain (Biometrics?)

2.1.3 Authentication Token

A smart card is an authentication token. This is the ultimate descendant of plastic cards and magnetic stripe cards. Plastic cards were originally and widely used in banking applications using a unique ID number in the form of embossed characters in the card body. Then came magnetic stripe cards, able to digitally store some kiloBytes (kB) of information such as owner name, birthdate, account number. A smart card is a small computer, silicon-based, able to *dynamically* store AND process (compute operations) several kB of information. Other exotic cards may be cited such as optical memory cards and variant as CD-ROM type cards, able to *statically* store few MegaBytes (MB) of information. One essential characteristic of a smart card is tamper-resistance (i.e. can't be easily forged or copied).

Figure 2.1: Different types of cards

2.2 Smart Card Architecture

2.2.1 Physical Characteristics

A smart card is no more than a piece of silicon in a piece of plastic. Figure 2.2 depicts the manufacturing of smart cards: a silicon chip connected to a contact module, the whole being reported onto a plastic card body. Figure 2.3 depicts in detail a smart card module.

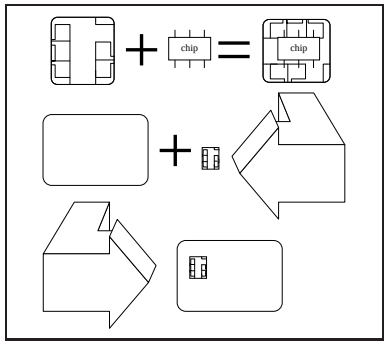


Figure 2.2: Smart Card Manufacturing

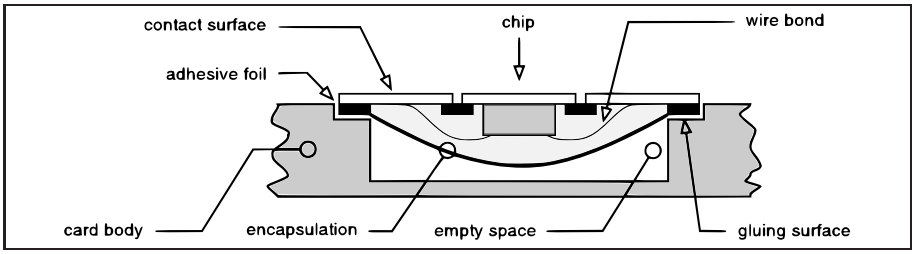


Figure 2.3: Smart Card Module Architecture

Smart Card physical characteristics are defined by ISO/IEC 7816-1 (cardbody size) and ISO/IEC 7816-2 (location of the contacts).

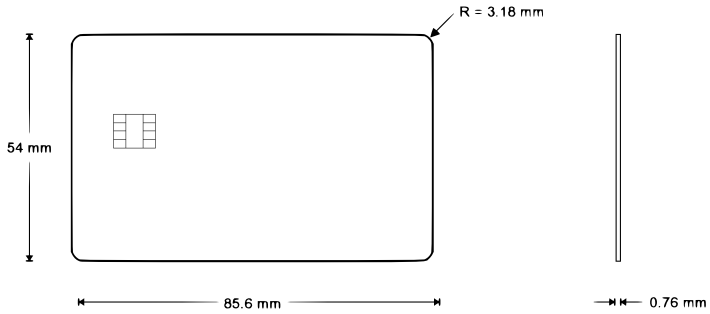


Figure 2.4: Smart card dimension and contact location

2.2.2 Electrical Characteristics

Electrical characteristics are defined by ISO/IEC 7816-3 (electrical interface and transmission protocol). Basically, most smart cards use an asynchronous serial transmission protocol, character oriented. The electrical interface is five contacts: Vcc and GND (power supply), Reset (initialisation), Clock and I/O (serial interface).

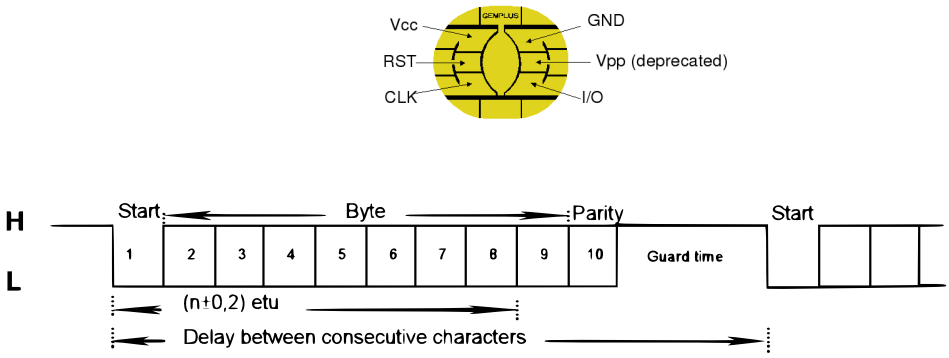


Figure 2.5: Smart card contacts attribution and I/O character frame

2.2.3 Memory Cards

The first smart cards used in large quantities were memory cards for telephone applications. These cards are prepaid, with the value stored electronically in the chip being decreased by the amount of the call charge each time the card is used. The data needed by the application are stored in the memory, which is usually an Erasable Electrically Programmable Read Only Memory (EEPROM). Access to the memory is controlled by the security logic, which in the simplest case consists only of write protection or erase protection for the memory or certain memory regions. This type of smart card can be used not only for telephone calls, but also whenever goods or services are to be sold against prior payment without the use of cash. Examples of possible uses include local public transport, vending machines of all types,

cafeterias, swimming pools, car parks and so on. The advantage of this type of card lies in its simple technology (the surface area of the chip is typically only a few square millimeters), and hence its low cost. The disadvantage is that the card cannot be reused once it is empty and must be discarded as waste.

Most advanced security logic for memory cards can also implement simple encryption. Current memory cards use ISO/IEC 7816-3 serial protocol, however old memory cards were using classical interface of IC serial-access memories, such as I^2C. Figure 2.6 depicts the general architecture of a memory card.

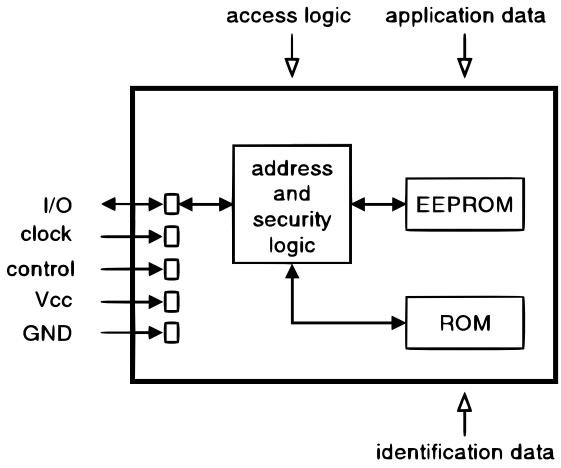


Figure 2.6: Architecture of a memory card

The heart of the chip in a memory card is the application memory (EEPROM), which is usually surrounded by three additional functional blocks: masked Read Only Memory (ROM) -for identification data-, access & security logic, and an I/O port.

2.2.4 Microprocessor Cards

Microprocessor cards were first used in the form of bank cards in France. Their ability to securely store private keys and execute modern cryptographic algorithms made it possible to implement highly secure offline payment systems. Following a drastic reduction in the cost of smart cards in the early 1990s due to mass production, new applications have been introduced, such as the SIM cards in GSM networks. Possible applications for microprocessor cards include identification, access control systems for restricted areas and computers, secure data storage, electronic signatures and electronic purses, as well as multifunctional cards incorporating several applications in a single card. Modern smart-card operating systems also allow new applications to be loaded into a card after it has already been issued to the user, without compromising the security of the various applications. This new flexibility opens up completely new application areas. The essential advantages of microprocessor cards are large storage capacity, the ability to securely store confidential data and the ability to execute cryptographic algorithms.

Figure 2.7 depicts the general architecture of a microprocessor card.

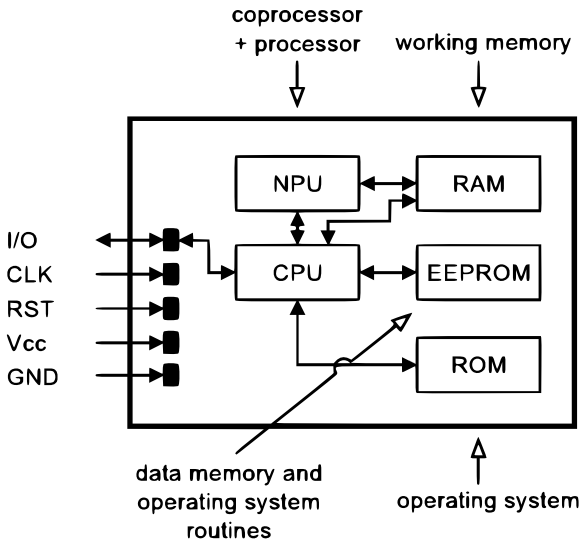


Figure 2.7: Architecture of a microprocessor card

The heart of a microprocessor card, as the name suggests, is a Central Processing Unit (CPU) and eventually a Numeric Processing Unit (NPU, aka cryptoprocessor), which are usually surrounded by four additional functional blocks: masked ROM, EEPROM, Random Access Memory (RAM) -as working memory for the CPU- and an I/O port. These functional blocks are all included in a single silicon chip.

2.2.5 Contactless Cards

A so-called contactless smart card only differs from classical contact cards by the communication interface. Both memory and microprocessor smart card could use a contactless interface, often identifiable by the absence of golden contact plates. Contactless cards, in which energy and data are transferred without any physical contact between the card and the terminal, have achieved the status of commercial products in the last decade, allowing very convenient applications where the card does not necessarily have to be held in the user's hand during use, but can remain in the user's purse or wallet. Very popular applications are access control in public transportation, corporate badges, ski passes, whereas most recent applications are ePassports and banking transactions. Moreover, This type of card also gets rid of wear and manufacturing failure rate of the electrical contact.

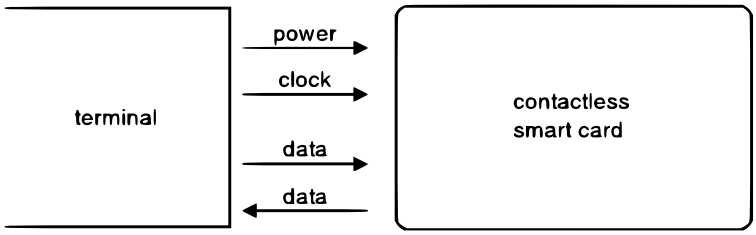


Figure 2.8: Architecture of a contactless card communication interface

Regarding internal architecture, only the I/O part is different compared to above described architectures for memory and microprocessor smart cards. Classical communication range is up to 10cm (proximity). However, many applications, needing the proof of the willingness of the user for the transaction, may reduce this range to only few millimeters. Since the power supply of the chip is coming from the RF interface, we may notice that the processing power of a microprocessor card will depend on the distance to the reader interface, thus making complex computations not so easy to implement in comparison to contact smart cards. Contactless smart cards interface and protocol is defined by ISO/IEC 14443. Contactless interface actually opens the way to any form factor, other than a classical smart card body, by embedding the chip and its antenna: keyfob, wristwatch, ring, even underskin implantation!

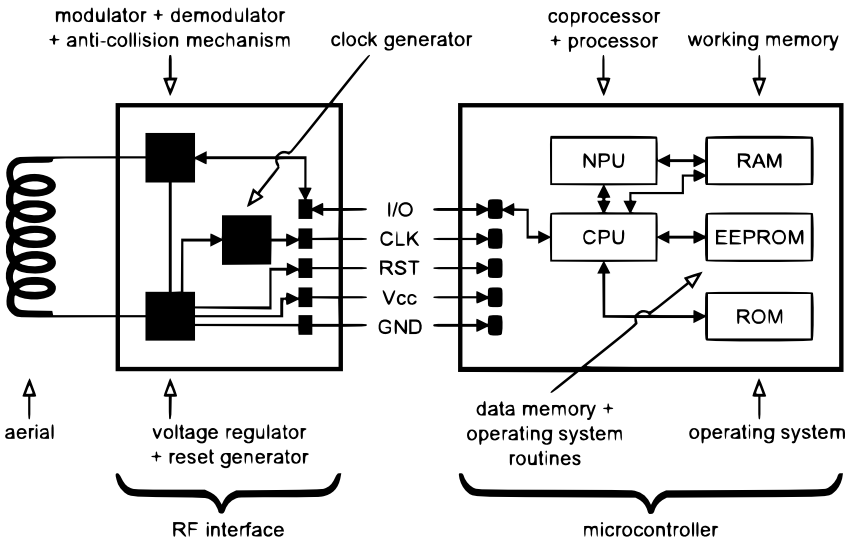


Figure 2.9: Architecture of a contactless card

2.2.6 Other complex architectures

One may talk about *combi-cards* or *dual-cards* (actually *dual-interface* card). Combi-card generally refers to one smart card body embedding two different, and separated, electronic chips and applications, one being contact and the other being contacless. This is only two separate cards. Dual-interface smart card refers to one chip and its application(s) being accessible both by contact and contactless. However, under these two generic nicknames, we may find more complex architecture such as two chips (or possibly two cores, memory and/or microcontroller, on the same silicon piece) with shared memory, but separated applications. See figure 2.10.

Roughly speaking, contact interface is preferred for security, whereas contactless interface is preferred for convenience. Typical applications are a corporate badge using contactless interface for physical access control and contact interface in a PC-card smart card reader for logical access control and emails encryption/signature or a multiapplication, let's say transportation and banking, card using contactless for both public transportation and small amount transactions, and contact interface for significant amount transactions such as cash withdrawals in ATM.

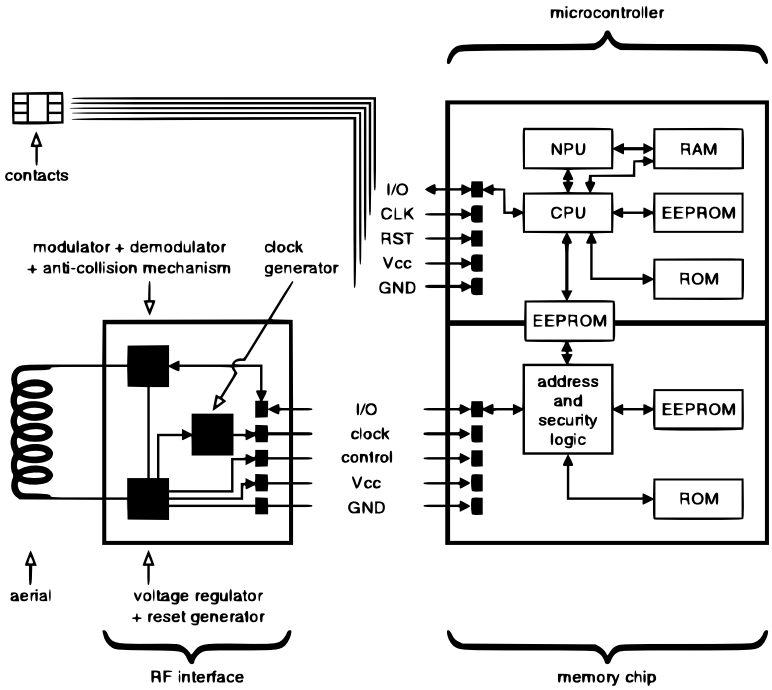


Figure 2.10: Exemple of complex architecture

2.2.7 Smart Cards vs RFID

Unfortunately, someone may refer to a contactless smart card as a RFID or vice-versa, the best example being the biometric ePassport too often referred to as a RFID device. For instance in the US, most "civil libertarians" have serious issues with RFID regarding privacy, they are right but tend to put badges and ePassports in the same category.

A RFID (Radio Frequency IDentification) device simply presents an ID to a reader device using radio frequency, just like an electronic barcode. Most RFID devices are not tamper-resistant, are clonable, and the main driver is low cost, in opposition to contactless smart cards, which are real security devices with computation capabilities and/or secured memory storage. Only the most simple implementation of a memory card is a RFID.

2.3 Operating Systems

The first smart cards were only state machines responding to specific commands such as sending an ID, incrementing or decrementing a counter, storing a data, and were one application dedicated. Following the world of computers, in the late eighties came the first operating systems for smart cards, and virtual machines for smart cards came in the late nineties.

2.3.1 Native

Using an operating system, usually named COS (Chip Operating System) or SCOS (Smart Card Operating System), allows several applications to be stored, used and managed independently in a single smart

card. However, smart card applications had to be developed for specific OS on specific microcontroller. OS is ROMized (masked in ROM), whereas applications may be loaded in EEPROM after card issuance. Each native operating system is proprietary of the card manufacturer.

2.3.2 Open Platforms for Smart Cards

A virtual machine (VM) provides a hardware abstraction layer, allowing any application, loaded after card issuance, to run on any VM compliant smart card, hence achieving the ultimate flexibility. This brings the needed interoperability between smart card manufacturers. However, this flexibility has a cost: operations are slower than on native implementation and the VM is using a lot of memory resources of the chip.

MultOS

Coming from the banking area during the development of the Mondex electronic purse, MultOS (**Mult**i-applications **OS**) was the very first operating system for smart card. In opposition to other VM solutions described below, developing applications for such cards, using a specific MultOS Executable Language (interpreted byte code) was only for a specialist of the industry. As of today, MultOS is still alive, however rare, on the market in banking applications and few national ID programs.

Javacard

Javacard is the most used solution on the market. Javacard comes from the Java programming language, developed in the early nineties by SUN Microsystems as the modern platform-independent language: "Write once, run anywhere", solving issues with code portability in a heterogeneous market. Actually Javacard is a subset of Java, developers have to learn constraints of javacard and use dedicated tools for compilation and byte code generation. A PC Java byte code is not compatible with the Javacard byte code. We implicitely talk here about the current Javacard 2.O specification from SUN Microsystems, the one widely spread on the market. The specification of Javacard 3.0 exists for the future and defines a real Java interpreter in a smart card, compatible with the PC one.

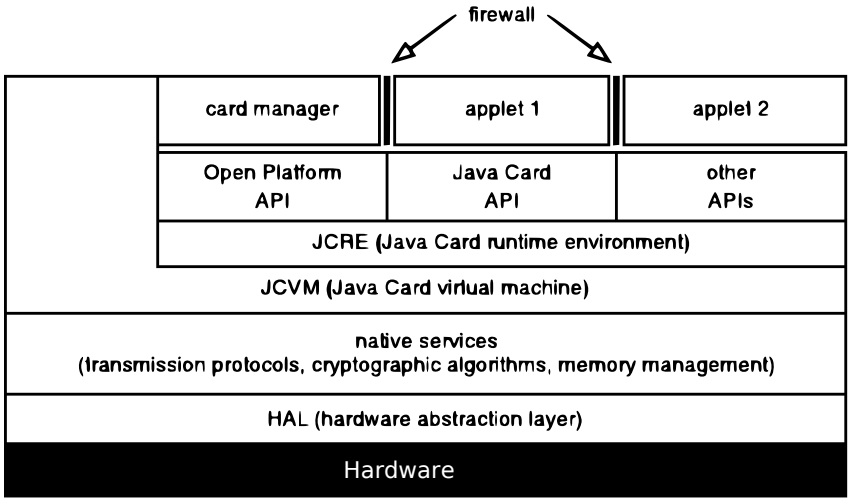


Figure 2.11: Architecture of a Javacard

.Net

Recently, Microsoft and its VM solution named *.Net* gains momentum in the computer industry as an alternative to Java, with script language very easy even for young developers. Naturally the smart card industry goes to .Net for cost-effective new developments, especially for smart card applications dedicated to the PC environment (and Microsoft Windows) such as logical access control. The advantage here is that the .Net assembly code is the same for smart card as the PC one.

Others

In the late nineties, Microsoft launched **Windows for Smart Card** (WfSC) with a large support from smart card manufacturers, arguing the accessibility of Visual Basic or C++ to any software developer, in opposition to native smart card OS and emerging Javacard. However it never took-off and WfSC resigned a couple of years later.

In Germany, in the mid-nineties, came the idea of using the well-known Basic programming language for the development of smart cards applications, the **BasicCard**. Actually this was a dedicated subset of Basic, especially developed by the promoting company ZeitControl Cardsystems, and the company proposed smart cards embedding the ZeitControl Basic interpreter. However, it never entered in the industry and was only used in small niches or for academic purposes.

Coming from the handheld industry, small memory footprint **Linux** OS are now studied as a potential candidate for smart card, allowing smart card manufacturers to get rid of licensing fees due to Java and .Net owners.

2.4 Applications

2.4.1 Telephony

Prepaid memory cards were the very first commercial application in public telephony. With the rising of mobile telephony, public telephony became a very small market, still alive in developping countries. There are well over 2 billion smart cards in use today within mobile communications devices. Most of these devices are Subscriber Identity Module (SIM) used in Global System for Mobile communications (GSM). This card ensures the authentication to the service provider network, transparently through any mobile handheld, and then confidentiality of the communication. The **SIM card** is a particular implementation of so-called SAM modules (Security Access Module) described below. Regarding telephony, SAM cards were found in modems for remote access to the Internet during the nineties. Beside its interesting features for the mobile operator, the SIM card also offers useful features to the end user: secure storage and portability of personal data such as the personal phone book or text messages.

2.4.2 Banking

This was the very first commercial application for microprocessor cards in France. The **payment card** represents the strong link between the customer and its bank, allowing automatic operations such as payment in shops with online verification of the bank account for solvency. The French CB example was followed by large international financial institutions such as Visa and Mastercard and is now a worldwide standard named EMV (Europay, Mastercard, Visa). From the nineties to nowadays, several trial of smart cards as an **electronic purse** failed for wide adoption by the public. The idea of electronic purse is to replace coins for small transactions, the application is close to a secured memory card with debit and credit capabilities. Recently, financial institutions tested contactless payment cards for their convenience.

As a convergence application with mobile telephony, we may see field tests of payment by presenting a mobile phone to an antenna. The SIM card also embed a contactless banking application through NFC (Near-Field Communication) interface.

2.4.3 Identity & Access Management

SAM modules (Security Access Module) are found in electronic equipments needing strong authentication based on cryptographic primitives. For instance, security modules are found in DVB (**Digital Video Broadcasting**, aka PayTV) devices and incorporated in **corporate badges** for PC logon application. As of today, Trusted Platform Modules (TPM) tend to replace SAM cards where portability is not an issue, since TPM are soldered on the mainboard, close to the application CPU, and are thus not removable. As a convergence application with telephony, extended SIM cards serve today in mobile phone as secure module for mobile TV applications. SAM modules may also be found in trusted smart card readers for strong mutual authentication with the end-user smart card, such as in health card infrastructures. Besides all these logical access applications, **public transportation** (e.g. monthly ticket for bus and metro) is a physical access application close to a rechargeable secured memory card.

2.4.4 Identity & Travel Documents

After the tsunami of mobile telephony for the past fifteen years, governmental applications are now considered as the next Eldorado for smart cards. Following international regulations, every issued **passport** is now electronic and embeds a contactless smart card chip (and not a simple RFID!). A lot of **National ID** programs are going to electronic, the same for **drivers licenses**, **health cards**, **visa applications** etc. All these applications can be found under the nickname ***eGov***. Even Corporate badges of government workers, such as PIV card in the US, are closer to an ePassport than a classical (and private) company corporate badge considered as an IAM application.

The first eID cards with fingerprint recognition and Match-on-Card were issued in United Arab Emirates in 2004. Biometric ePassports are now issued since 2006. These applications represent the most advanced technology in terms of physical and digital security. Unlike SIM cards and banking cards, where the relevance of the digital function is dominant, all the features of the whole ID document are important as these documents also serve as visual identification. Beyond protecting against digital counterfeit, these products also propose advanced physical countermeasures, inherited from years of research in banknotes and (paper or plastic) ID security such as UV printing, Optically Variable Inks (OVI), laser engraving. Another big challenge is the durability of such smart card products: five or ten years under physical and environmental stress, in comparison to only two years for banking cards (with the same stress) and quick renewal for SIM cards (with no stress). The dominant standard for document interoperabily is coming from ICAO (International Civil Aviation Organization), mandatory for ePassports and largely influencing eID platforms. The Logical Data Structure (LDS) of ICAO defines access conditions and organization of the memory in so-called data groups: personal data (e.g. name, birthplace, address), face photograph, two optional fingerprint images (mandatory in Europe in Mid'09), two optional iris images, digitalized handwritten signature, digital certificates, etc...

2.5 Multicomponents Smart Cards

Beyond just a single piece of silicon in a piece of plastic, smart card technology has a trend of multiple components. Since the middle of the nineties, smart cards manufacturers have been prototyping different card concepts to enhance usability and security. The industry produces many nice mockups but finally the manufacturing cost for large production is not acceptable. Multiple components, hence multiple connexions, multiplies the failure rate of production. The main constraints are standardized card thickness and flexibility need (ISO "flexion & torsion" test). The idea of such a card concept is to take advantage of the large "free" surface of a conventional smart card. Thin components suitable to smart card are LCD or OLED flexible displays, micro-batteries, piezo-electric power source, quartz clock source, silicon-based fingerprint sensors, keypad, MEMS (Micro ElectroMechanical System) or large memory chips. A typical convenient application for a display is the ability for the user to directly read the balance of his electronic purse, whereas the security application is the generation and display of a One-Time cryptogram to be used for active authentication. A self-powered smart card could implement interesting active security features for tamper-resistance such as temperature or light sensor, or it could offer the possibility to save RAM memory and the possibility for large range contactless communications. A fingerprint-enabled smart card could provide a three-factor authentication and prove the "physical link" with the user.

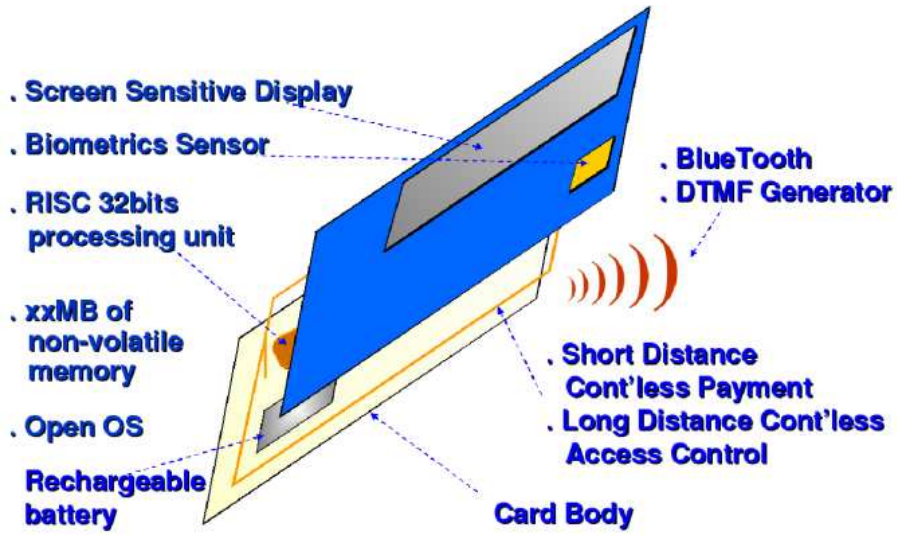


Figure 2.12: Our Concept Card

Our work in this area has been published in [89].

2.5.1 Screen cards

In the Gemplus Research Lab, we originally developed smart cards with different display technologies. Our goal was to demonstrate the feasibility of a high-end product with real interactivity. We used a Ricoh twisted nematic display based on plastic substrate and electronics embedded on a flexible 100 μm glass epoxy circuit board.

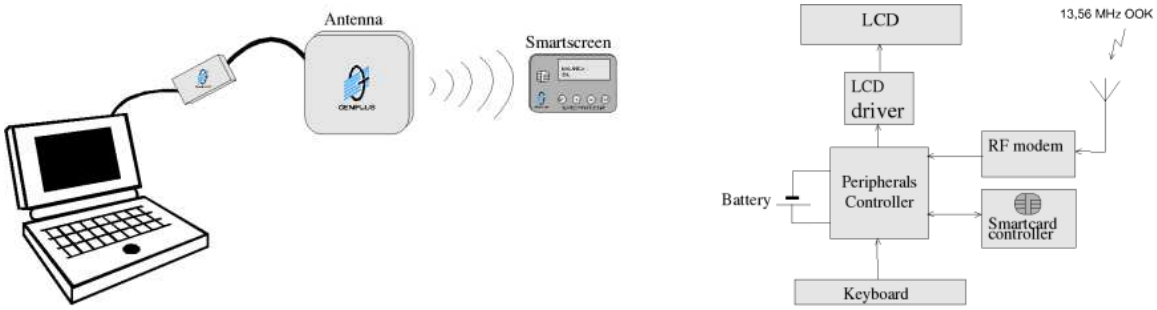


Figure 2.13: ScreenCard application and architecture

Here the challenges were multiple, both electronic and mechanical: very thin ($< 0.5mm$) and low power consumption components, flexion-resistant components placement and interconnection, cardbody integration.

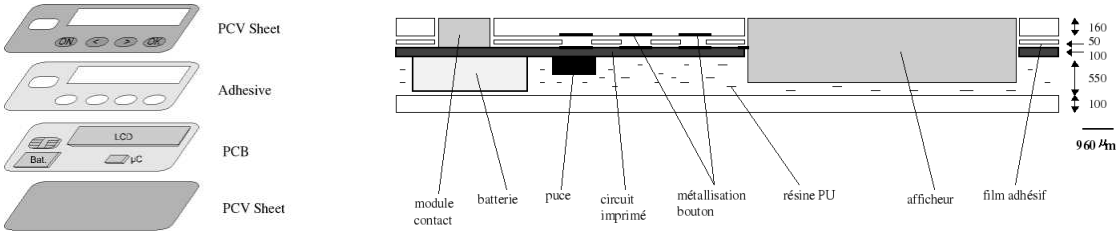


Figure 2.14: ScreenCard physical layout

Figure 2.15: Screencard flexible printed circuit board

This project was fully successful but the industrialization of the product is difficult because of the high number of connections to handle. For this type of dot matrix display, one must connect a 120-pin driver with a 96-connection display. This gives poor yields and poor mechanical robustness that strongly affects the cost of the card.

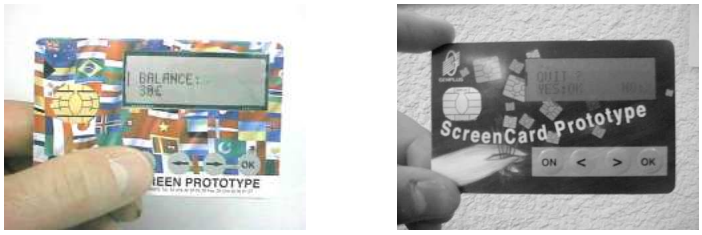


Figure 2.16: Screencard prototypes

2.5.2 Extended memory cards

Another challenge was to extend memory capabilities of smart cards above the classical few tens of kB by using multiple chips technology and conductive glue interconnexions. We were able to demonstrate the feasibility of 2MB SIM cards and 224MB smart cards. These technologies are currently in use today, for instance within the MultiMedia SIM Cards plugged in SmartPhones.

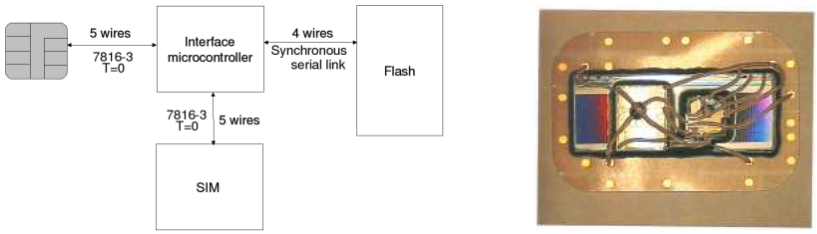


Figure 2.17: 2MB SIM card architecture and module prototype

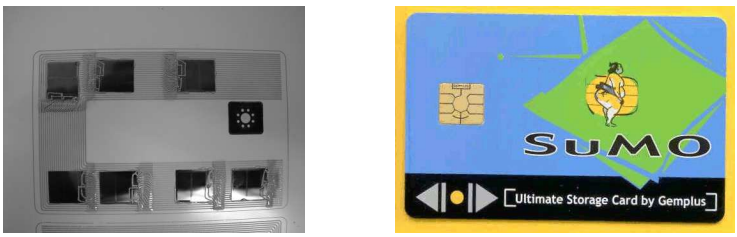


Figure 2.18: 224MB smart card prototype

2.5.3 Fingerprint cards

With the emergence of silicon-based fingerprint sensors and our experience in silicon chip integration, the idea came naturally that fingerprint-enabled smart cards could replace or complement user authentication with PIN code. The mechanical challenge of the integration and resistance to flexions & torsions was a success (see figure 2.19), however issues still remainded regarding the electronic challenge: beyond just capturing an image, power consumption of such chips (for autonomous or contactless cards) and especially image processing needed for fingerprint comparison is out of the capabilities of classical smart cards.

Then the question was **"How to achieve such a product and for doing what?"**.

We will briefly discuss this point in the next section.

And ***this is the entry point of our interest for biometrics and security, leading to this PhD dissertation***.

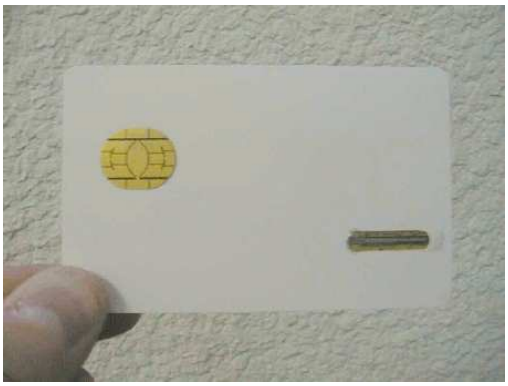


Figure 2.19: Fingerprint-enabled smart card

2.6 Interaction with Biometrics

Gemplus manufactured the very first combo-reader (i.e. fingerprint sensor + smart card reader in one device) in 1997. The original application was convenience for logical access control, fingerprint capture replacing PIN code presentation for corporate logon application. However security was in mind, the reference template being securely stored in the user's smart card instead of the client laptop. Actually not so convenient (no card = no logon), thus sufficiently secure. Beyond storing the reference template, a possible feature for any competing technology (e.g. optical memory card, barcode), a microprocessor smart card can take advantage of its processing capabilities to surpass the competition. Then came the idea to study the complexity of fingerprint software primitives such as extraction and matching, to find suitable ways to develop such tools onto the more limited smart card platform. This led to the Match-on-Card feature: the smart card not only stores the reference template, but is also able to compute the comparison. This feature comes with several issues: performance on contactless cards (i.e. chip powered by the RF field, less energy = less processing power), performance through virtual machines (i.e. javacard or .Net implementations are slower for intensive operations).

Instead of receiving fingerprint information from the insecure external world, the smart card could embed the fingerprint sensor as described in the previous section. The concept of personal sensor is especially appreciated in Asia for sanitary reason. This leads to multiple architectures: autonomous smart cards, secure interaction with an external powerful smart card reader to deport complex operations such as data extraction from the original image, sensor interaction when the contact smart card is plugged into a reader.

2.6.1 The Personal Token

The most important feature of the smart card within the biometric scheme is its role of *personal token*. In most non-governmental applications, regulations from privacy-concerned organization (such as CNIL in France, BnD in Germany or PFPDT in Switzerland) do not allow the creation of centralized databases of biometric data. Theses regulations even lay down the mandatory use of a personal token, often a smart card, to "distribute" the database of users' biometric data. In France, the CNIL even advises the use of Match-on-Card technology, together with smart cards, for confidence of the end user: he is the carrier of his own biometric reference and somewhat controls the comparison engine.

CHAPTER 3
Introduction to Cryptography

Contents

3.1 A few words about Cryptography

We may refer the reader to [79, 115, 125] for a complete overview of Cryptography.

3.1.1 A little bit of History

Since the origin of history, fierce battles have been waged between *codemakers* and *codebreakers*. Prehistoric hieroglyphs, Egyptian hieroglyphs, druidic runes were communication tools reserved to certain high-castes in order to differentiate and protect their community. More intentionally, secret communications were key elements of how wars were won and lost, from the ancient Greece to the world war II and the famous German Enigma machine. All this refers to the so-called *conventional cryptography*. From the middle of the last century, *modern cryptography* is more related to diplomacy, business, espionage and unfortunately terrorism with the recent use of steganography to set up 2001/9/11 terrorist attacks in the US. Modern crytography in IT began with IBM and Horst Feistel developments during the early seventies to finally build the US national standard Data Encrytion System (DES) in 1977. Most current communication systems extensivelly use cryptography: banking networks, mobile telephony, satellite television, internet. Regarding IT, the reader will find more information and crytographic tools description within the following sections.

3.1.2 A little bit of Science Fiction

Beyond excellent books about the history of cryptography, such as David Kahn's *The codebreaker* or French Jacques Stern' *La science du secret*, cryptography feeds several novels more or less influenced by historical facts. The best recent example is Dan Brown's *The Da Vinci code* with, among others enigmas, the description of the so-called *cryptex*, a 5-wheel coded portable vault with tamper-resistance feature (auto-destruction of the contained secret scroll if not used the proper way). Such a device did never exist! Regarding cinema, the German Enigma encryption machine feeds at least two anticipation movies: Michael Apted's *Enigma* is a free adaption about Alan Turing and the Bletchley Park history (reverse-engineering of the Enigma machine), whereas Jonathan Mostow's *U-571* is about American submariners trying to capture the Enigma machine within a German U-boat(of course, historically false). More related to modern cryptography, the interesting Harold Becker's *Mercury rising* involves Bruce Willis as a FBI agent who is assigned to protect a 9-year old autistic boy who is the target for assassins, mandated by the NSA, after cracking a top secret government code. An enigma from the code is published, for testing, in a general public puzzle magazine, with a reward (after calling a secret NSA phone number) if resolved. Another interesting movie is *Sneakers* in 1992 with Robert Redford as a white-hat hacker. Leonard Adleman (the "A" of RSA) served as consultant for this movie regarding the dialogs about cryptography. Moreover this movie funnily shows how to cheat with a voice recognition device: all words of the vocal passphrase "My name is *firstname lastname*, my voice is my passport, please verify" were recorded separately and then concatenated on an audio tape!

3.1.3 Science of Secret

The fundamental objective of cryptography is to enable two people, usually referred to as Alice and Bob, to communicate over an insecure channel in such a way that an opponent, usually referred to as Eve, cannot understand what is being said. Cryptography is historically used in military and diplomatic communications, and more recently, few tens of years, has found application in Information Technology security. Applications in IT security include communication encryption, digital signature, and user/device authentication.

3.1.4 Cryptology: Cryptography & Cryptanalysis

Cryptology is the science of cryptography and its dual: cryptanalysis. Cryptanalysis is the science of attacking cryptographic systems and evaluating their level of resistance to secret information leakage to recover even a very small part of the message and/or the key. Obviously there is no frontier between cryptography and cryptanalysis, researchers developping a cryptosystem must prove its security, thus developping cryptanalysis toolbox. Cryptanalysis is usually referred to as "*codebreaking*".

3.1.5 Symmetric and Asymmetric Cryptography

As of today, two kinds of cryptosystems exist. The first one being used since Antiquity, a cryptosystem where the same *secret key* is shared between communicating parties, often referred to as *Secret Key Cryptography* or *Symmetric Cryptography*. The second one, more recently, was introduced in 1976 by **`Whitfield Diffie`** and **`Martin Hellman`** in "New directions in Cryptography" [33], followed by a first implementation in 1977 by **`Ronald Rivest, Adi Shamir`** and **`Leonard Adleman`** in "A method for obtaining digital signatures and public-key cryptosystems" [98]. This brand new scheme uses two complementary keys, the *private key* and the *public key*, such that only the public key is needed for certain operations, encryption for instance, and only the private key is needed for other operations, decryption for instance. This latter cryptosystem is often referred to as *Public Key Cryptography* or *Asymmetric Cryptography*.

3.1.6 Applications

Today's IT applications use cryptography extensively, both conventional and modern. Applications cover banking, mobile telephony, video broadcasting, eGovernment and ID documents, access control, security over the Internet, this latter one being the best example of the public and insecure communication channel. In general such an application is associated to a security element able to compute cryptographic algorithms and/or store large cryptographic keys, a smart card electronic chip is most frequently used.

3.2 Cryptography Goals

3.2.1 Confidentiality

The very first and intuitive goal of cryptography is the protection of *confidentiality*; anyone intercepting an encrypted message must be unable to recover the original message, without having access to the cyphering key. This confidentiality feature is obtained with encryption/decryption schemes. Encryption, for instance, is a so-called *primitive* of cryptography, i.e. one tool of the toolbox.

3.2.2 Integrity

The second, out of the three most important features, is *integrity*. This ensures the receiver that the message is the original one and has not been modified by a malicious third-person. The integrity primitive is the so-called *hash function*.

3.2.3 Authentication

The third important one is *authentication*. This ensures the receiver that the message is really coming from the right sender, who couldn't be impersonated by a malicious third-person. The integrity primitives are the so-called *Message Authentication Code (MAC) function* or *Digital Signature* when using asymmetric primitives.

3.2.4 Identification

This notion is close to the previous one, here the goal being to directly authenticate our interlocuter and not a message. The person is generally authenticated with a secret that he or she possesses. This identification feature is based on the so-called *Challenge-Response* protocol.

3.2.5 Others

Several other cryptographic goals could be acheived with classical primitives depending on the application needs. Here is a non-exhaustive list: anonymity (e.g. electronic voting), commitment (e.g. online gaming), non-repudiation (e.g. financial transactions), randomness (e.g. online gaming), zero-knowledge (e.g. online user authentication), availability of services.

3.3 Basic Primitives of Cryptography

3.3.1 Encryption/Decryption Functions

Encryption is the process of transforming clear information (referred to as *Plaintext* or *Message*) to an unreadable information (referred to as *Ciphertext* or *Cryptogram*), except for whose having the special knowledge (referred to as the *Key*).

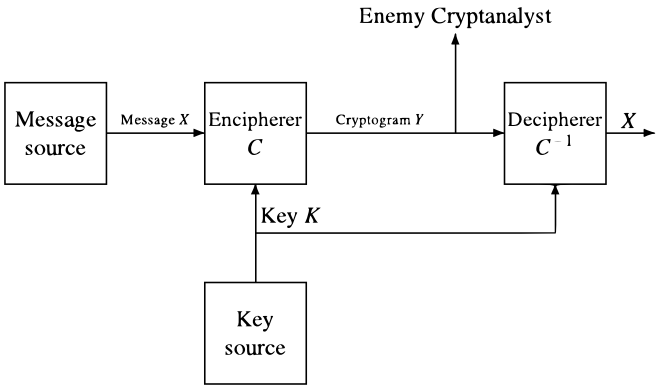


Figure 3.1: The Shannon encryption model

Figure 3.1 depicts the symmetric encryption/decryption system referred to as the Shannon model, whereas figure 3.2 depicts the asymmetric encryption/decryption system, where the sender uses the receiver's public key to encrypt while the receiver uses his own private key to decrypt.

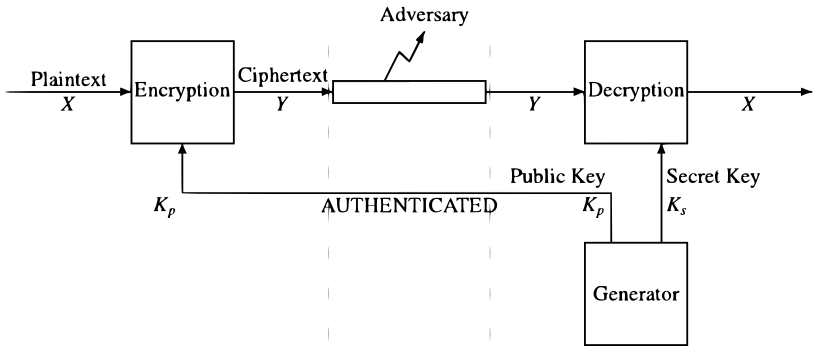


Figure 3.2: The asymmetric encryption model

We may notice here the existence of systems dedicated to predetermined data (e.g. file encryption) referred to as *block ciphers* and systems dedicated to on-the-fly encryption (e.g. voice in mobile telephony) referred to as *stream ciphers*. Some well known solutions, and very much used in IT, are DES and AES for symmetric cryptography and RSA for asymmetric cryptography.

3.3.2 Hash Functions

Hashing is the process of transforming and reducing clear information, the message, to a very short data representative of the message(generally 128, 160 or 256 bits), (referred to as *Hash value* or *Fingerprint*. For obvious reason during this dissertation we will only use the term Hash value!). Hashing is a one-way function (i.e. irreversible: one can't retrieve the message from its hash value) and is ideally collision-free (i.e. two different messages can't have the same hash value). Basically, hashing is a lossy compression function.

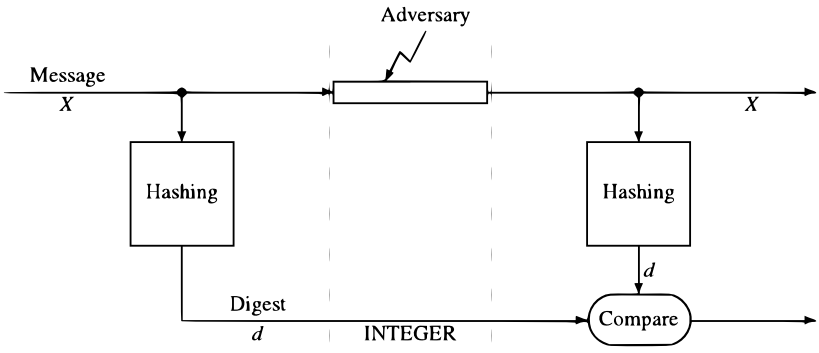


Figure 3.3: The integrity channel

Some well known solutions, and very much used in IT, are MD5 and SHA-1. The hash is used to ensure the message integrity, since the couple {message, hash value} can't be forged, in theory. Beyond cryptography, hashing is a classical technique to index data in arrays and is widely used in large database management systems.

3.3.3 Message Authentication Codes (MAC)

MAC is basically a keyed hash function: the obvious way to build a MAC from a message is to hash it and to encrypt the hash value with a secret key. This ensures the message for both integrity and authentication at the same time. Sometimes wrongly called digital signature in symmetric mode, see next section.

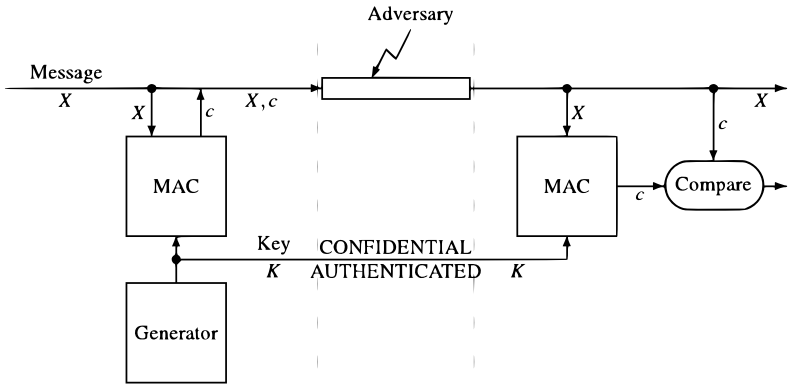


Figure 3.4: The authentication channel

3.3.4 Digital Signatures

A Digital Signature is a mathematical scheme for demonstrating the authenticity of a digital message or document. A valid digital signature gives a recipient reason to believe that the message was created by a known sender, and that it was not altered in transit. Basically, digital signature is the reverse of the encrytion scheme in asymmetric mode: one uses his own private key to cipher, whereas the verifier uses the sender's public key to decipher.

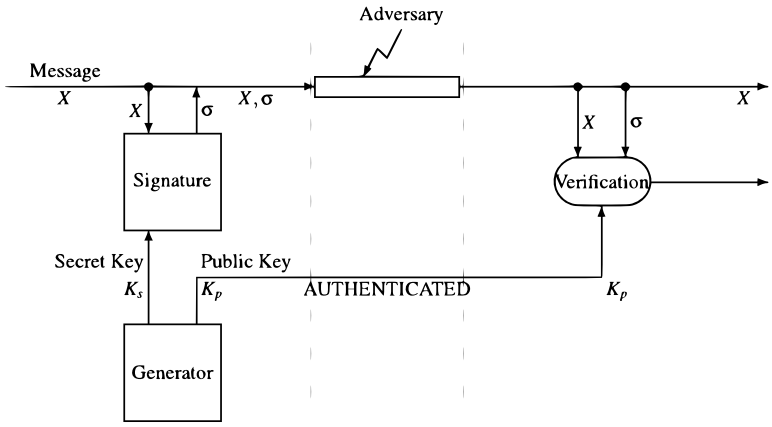


Figure 3.5: Digital Signature

The most important feature for digital signature is non-repudiation (i.e. the signing party can't deny being the author). This non-repudiation can't be achieved with a MAC, described in the previous section, since any MAC verifier owns the same key as the MAC issuer and thus can

forge the message and generate its MAC. Non-repudiation can only be achieved with asymmetric cryptography since the verifier uses the public key of the sender, whereas the sender uses his private key to generate the digital signature.

3.4 Basic Protocols of Cryptography

3.4.1 Challenge-Response & Mutual Authentication

Challenge-Response authentication is a family of protocols in which one party presents a question ("challenge") and another party must provide a valid answer ("response") to be authenticated. The simplest example of a challenge-response protocol is password authentication, where the challenge is asking for the password and the valid response is the correct password. Nowadays, in computer science, this protocol proves the challenged party knows a secret, but without communicating it. Basically, the challengers send a random value to the challenged party, which is intended to encrypt this random value with the secret key and sends back the result to the challenger, for comparison with his own calculated value, as depicted in figure 3.6.

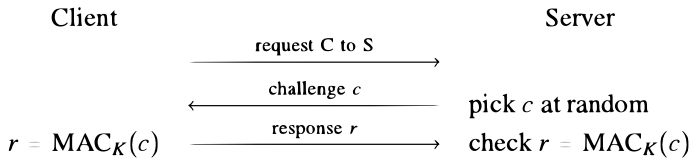


Figure 3.6: Challenge/Response

Mutual authentication is just a two-way challenge-response, where after the first authentication the challenger become the challenged party and vice-versa. Using asymmetric cryptography, this can also be done by signing the challenge with a private key.

3.4.2 Key Generation & Key Agreement

Key(s) generation in both symmetric and asymmetric systems must rely on random (actually pseudo-random) features, avoiding predictability of the current, or next, key(s). In the particular case of asymmetric cryptography, the random process generates one key of the key-pair, the complementary key being mathematically computed from the first one. A necessary, but usually not sufficient, condition for an encryption scheme to be secure is that the key space is large enough to preclude exhaustive search.

Key agreement is a mechanism in which a shared secret is derived by two parties, as a function of information associated with each of these, such that no party (and especially a malicious third-party) can predetermine the resulting value. A well-known key agreement protocol is the Diffie-Hellman [33] exponential key exchange, in which two parties jointly exponentiate a generator with random numbers, in such a way that an eavesdropper has no way of guessing what the key is. However, exponential key exchange does not specify any prior agreement or subsequent authentication between the participants; thus being an anonymous key agreement protocol, it is vulnerable to Man-in-the-middle attacks.

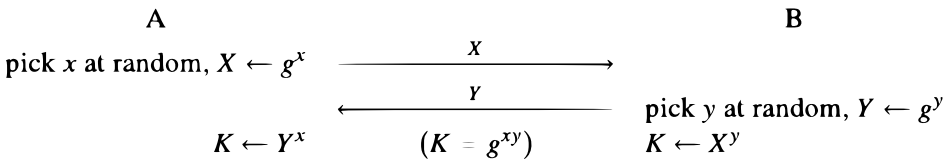


Figure 3.7: Diffie-Hellman key agreement

3.4.3 One-Time Passwords (OTP)

The purpose of a One-Time Password is to make it more difficult to gain unauthorized access to restricted resources, like a computer account. Traditionally static passwords can more easily be accessed by an unauthorized intruder given enough attempts and time. By constantly altering the password, as is done with a one-time password, this risk can be greatly reduced.

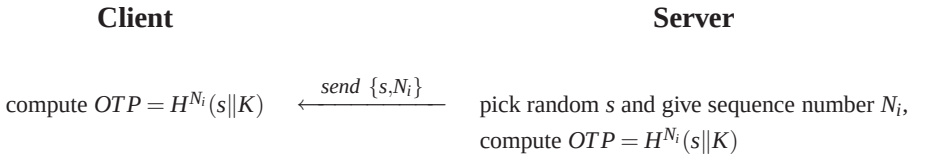


Figure 3.8: One-Time Password scheme

The most deployed solution is based on the internet standard RFC 2289: the server and the user share a secret key K, the server sends a random s and a sequence number N to the user, then both compute N times, with chaining, the hash of concatenated $s||K$. For each further authentication, both parties will decrement N thus agreeing on the same unique value. See figure 3.8.

3.5 Interaction with Smart Cards

Interaction between smart cards and cryptography is an old and well-known subject. A human can't memorize large numbers, such as ones needed for secure cryptographic keys and can't easily manipulate these numbers to process a response to a challenge. A smart card is the perfect secure container and generator (with native pseudo-random number generation) of such keys and computer of such authentication operations. A smart card is able, within few seconds, to generate a key pair for the RSA cryptosystem and is able, within a second, to agree on a key with another digital system. Cryptographic algorithms and tools - such as DES, 3DES, AES, RSA, MD5, SHA-1 - are natively implemented in smart cards' cryptoprocessors. OTP smart cards (in different form factors) are becoming the de-facto standard for secure remote authentication. Actually the smart card entirely plays here the role of personal token, because the system will authenticate the owned object instead for the real user. However, only a secret (e.g. PIN code) delivered by the user to the card will activate this latter one to process an online authentication. Smart cards and cryptography will thus achieve a perfect two-factor authentication by means

of *something-you-have* and *something-you-know*. Beyond human authentication, mutual digital devices authentication extensively uses SAM cards (Secure Access Modules), plugged in both communicating systems by means of transparent and removable/upgradable security features.

3.6 Interaction with Biometrics

Interaction between biometrics and cryptography is a quite recent and unexplored research area. These two technologies are somewhat paradoxical: on one hand, cryptography needs identically reproducible and uniformly distributed data, whereas on the other hand, biometric data are intrinsically not identically reproducible (we measure a distance between samples) and non-uniformly distributed (e.g. only one nose, in the middle of the face) data. More classically, in most current applications such as ePassport, cryptography helps to keep biometric data secure (regarding both integrity and authentication, and furthermore confidentiality, even if biometric data are public by definition). Beyond these common applications, thinking about advanced interaction may lead to unrealistic solutions (usually convenience-only driven), as often cited by unadvised people: use our fingerprint as a cryptographic key or password (and especially digitally signing with our fingerprint), with no idea about confidentiality and reproducibility issues, or use error-correction codes to retrieve the original "key" biometric data. More realistic, biometric data being never identically reproducible, one could use a biometric capture as a source of randomness to generate keys by hashing the captured data for instance. Obfuscation techniques to hide reference biometric data, and retrieve it, using a secret and a candidate data, as a positive response to a challenge if and only if the candidate is a right one, are also thinkable (will be described in chapter 16, part V). We will also overview in the last part interesting concepts such as Fuzzy Extractors, Secure Sketches, Fuzzy Vaults or Cancelable Biometrics.

"When theory (crypto) meets practice (human)...
a not so simple combination"

We will discuss this overall subject in the last part (V) of this dissertation.

PART II

Security Issues with Biometrics

CHAPTER 4
Introduction

Contents

4.1 General Issues

We may usually list eight security vulnerabilities in a biometric system [121], as depicted in figure 4.1

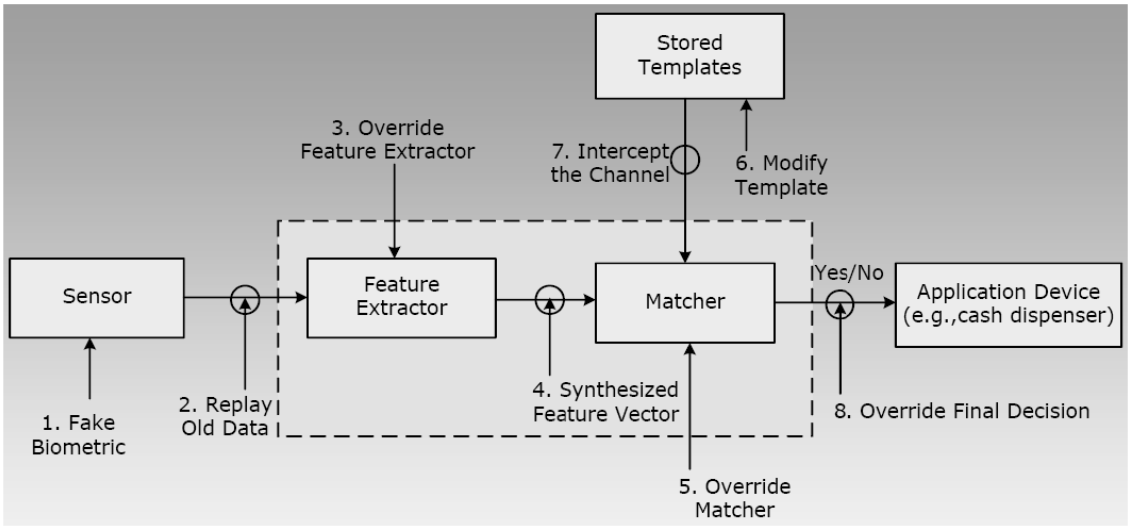


Figure 4.1: Flaws in biometric systems

1. Fake Biometri)c: this denotes the use of a faked biometric trait. We will detail this attack regarding fingerprint-based systems in chapter 7.

2. Replay Old Data: this denotes the replay of captured old matching data to gain access to the system later on.
3. Override Feat)ure Extractor: this denotes the replacement of the extractor by a malicious program, which will output the desired data for the attacker.
4. Synthesized Feature Vector: this denotes the use of artificially generated templates to bias the matching process. We will detail this attack regarding fingerprint-based systems in chapter 8.
5. Override Matcher: this denotes the replacement of the matcher by a malicious program, which will output artificially high scores.
6. Modify Template: this denotes the manipulation of the references database to add desired templates or modify existing templates.
7. Intercept the Channel: this denotes the capture and replacement of the reference data with the desired one.
8. Override Final Decision: this denotes the replacement of the final decision to force the access to the system. We may note here one of the major problems of any biometric system: in the end, only one bit of entropy!

All these vulnerabilities are related to identity theft, where the attacker will impersonate an authorized user.

Another issue of such systems is the possibility of the multiplication of identities within 1:1 systems (multi-enrollment). Depending on the security needs of the application, a 1:1 system may process a 1:N comparison within its local database, if any, for each new enrollment to check for duplication of identities.

As usual, attacks 2 to 8 may be tackled by using cryptographic tools for data integrity and authentication. Only type 1 attacks can't be tackled by classical security tools of IT systems. The only techniques to protect against theses attacks are the so-called Liveness Detection systems, aimed at verifying the validity of the living biometric sample (attached to a living person). We will detail theses techniques in chapter 9. Once again, even if one of these techniques would be perfect, this is not the panacea of security since the system may face a living person giving his biometric trait under threat (e.g. at gunpoint to an unstaffed system). In this case, classical silent alarms techniques may be useful (e.g. the user will place another finger, giving access to the system but activating the silent alarm).

4.2 Biometrics with Smart Cards

The use of biometrics without any personal device to store the reference template leads to privacy concerns: the centralized database which stores all biometric information from every user could be hacked. The use of a smart card here allows building up a distributed database where every user is the carrier of his own biometric reference, hence alleviating the previous privacy concern. Depending on the application, the smart card could handle differently the biometric data:

1. Storage-on-Card (SoC): the reference biometric template is stored on the smart card and is read by the system at any authentication request. This only uses non-volatile memory, hence allowing cost-effective smart card. However the reference template is exposed to different attacks when communicated out of the smart card.

2. Match-on-Card (MoC): the reference biometric template will never be communicated out of the smart card once written during the enrollment step. The candidate template is sent to the smart card and the comparison is processed internally. This protects the reference biometric template but needs a more powerful smart card in terms of processor and memory resources. This is particularly interesting when the result of the authentication has only to be used locally: applet activation, access to private key for digital signature. A malicious terminal capturing a candidate template will never have the information in return if it matches or not.

3. Partial Match-on-Card (PMoC): this solution has the advantages of both previous solutions, permitting cost-effective smart card and protecting the biometric information. The biometric information is split in two on the smart card: a public part to be read by the terminal and a private part which will be locked on the smart card [8]. The process is like a biometric challenge-response: the terminal reads the public part of the biometric information, process complex computation and send a candidate to the smart card to be compared with the private part of the biometric information using very light computation on the smart card's chip. Processing and decision entities are clearly separated.

The combination of biometrics and smart card is an old topic [50] but the idea of using biometrics to replace the PIN code for security reasons is too often cited. Biometrics capabilities are always overestimated. First of all, any biometric data is not a *secret*: a face can be seen on any picture or video recording even without the owner's authorization, fingerprints are left everywhere, voice can be recorded. Let's say Biometrics are *public* data, hence a biometric data can be captured and replayed [77, 76].

Different attacks and countermeasures are possible depending on the context of use of biometrics and smart card. We define here three contexts of use:

1. Attended (or Staffed) Terminal : the applicant is in front of the authority [‡]
2. Trusted Third-Party: under video surveillance [††]
3. Uncontrolled Area: user at home with the smart card and biometric device [‡‡]

For instance, only the last context of use would permit a manipulation of the biometric reader to bypass the captured image and replay a matching candidate; same for using a large man-in-the-middle device.

Only the first context of use will prevent from the discrete usage of a fingerprint copy or bad-looking smart card copy; same for using a discrete man-in-the-middle device. Classical attack paths are:

[‡] e.g. face to face with a policeman
[††] e.g. ATM, banks, shops
[‡‡] e.g. e-voting, e-commerce

1. Man-in-the-middle (capture and replay)
2. Finger substitution (gummy fingers)
3. Smart card substitution (forged cards, yes-cards)
4. Fingerprint and smart card readers manipulation (probing)

Most of these attacks, finger substitution apart, can be stopped by using cryptographic tools (e.g. mutual authentication, session key). As previously mentioned, the countermeasures against finger substitution are aliveness detection systems built in the biometric reader itself (e.g. pulse detection, skin conductivity), see chapter 9.

A miscellany of threats and countermeasures can be found in tables 4.1 and 4.2.

Context of use	Threats
Attended terminal	False cards, YesCards
Trusted third-party	same as above + false finger
Uncontrolled Area	same as above + Reader manipulation

Table 4.1: Threats

Context of use	Countermeasures
Attended terminal	Secure printing, signed data
Trusted third-party	signed data, aliveness detection
Uncontrolled Area	signed data, aliveness detection, tamper resistance

Table 4.2: Countermeasures

4.3 Biometrics vs Passwords

First of all, the security of a password-based authentication tool such as ones in Unix or Windows systems are based on the local storage of only cryptographic hashes of passwords, no passwords themselves. This is possible because of the *deterministic* nature of password authentication: if the entered candidate password is the right one then its hash value equals the stored hash value and the authentication succeeds; if the entered candidate password is a wrong one then its hash value is different and the authentication fails [39].

Such an approach of security is impossible with biometric data. Any new capture of a biometric candidate results in slightly different data which leads to the *statistical* nature of Biometrics-based authentication (distance evaluation between two samples). The hash value of a reference biometric template will be totaly different from the hash value of any matching candidate. This means that biometric references have to be stored locally in clear text (or maybe encrypted but encryption is a reversible function unlike a hash function which is a one-way function).

	Characteristics	PIN code	Biometrics
1	Secrecy	Secret	Public
2	Delegation ability	Yes	No
3	Changeability	Yes	No
4	Personalization	Easy	Difficult
5	Comparison process	Simple	Not so trivial
6	User convenience	No	Yes
7	Vulnerability to Eavesdropping	Yes	No
8	Vulnerability to Brute Force attack	Yes	Not so trivial
9	Attacks countermeasures	Mature	Immature
10	"Real" user authentication	No	Yes
11	Capture	Easy	Expensive

Table 4.3: Biometrics vs Passwords

A deep characteristics analysis of both passwords and Biometrics shows a clear opposition:

1. Secrecy: a password/PIN code is a secret, whereas biometric data are public. But we have to make here a distinction between biometrics leaving traces (e.g. fingerprints) and others (e.g. hand geometry)

2. Delegation: depending on the application, the delegation ability is mandatory (banking, mobile communications) or must be impossible (civilian identification documents)

3. Changeability: in case of compromise, a password is denied and another one is issued. It's not that easy with biometrics

4. Personalization: a PIN code is mailed (e.g. banking), whereas biometrics request user enrollment(i.e. the user has to go in a security area of the registration authority)

5. Comparison process: the comparison between two PIN codes is a very trivial task for a smart card, whereas comparing fingerprints needs far more computation resources

6. User convenience: a PIN code must be memorized and we often manage several PIN codes, whereas biometrics need no effort

7. Vulnerability to eavesdropping: a discrete monitoring of our actions could reveal our password, whereas biometric data can't be captured that way

8. Vulnerability to brute force attack: passwords are few characters long, whereas a biometric template is few hundreds of bytes

9. Countermeasures: attacks against PIN code and passwords are experienced for many years and countermeasures are mature. Attacks against biometric systems is a novel area with no mature countermeasures for the time being

10. "Real" user authentication: user authentication with PIN code is only a legal trick: the law says "this PIN code is personal, do not communicate it". Biometrics is a stronger link with the user himself

11. Capture: entering a PIN code is simple and cheap (small keyboard), whereas capturing a biometric trait is an expensive task (cost and maintenance of a reader)

This opposition confirms the good complementarity of passwords and biometrics. The replacement of one with the other should be carefully studied depending on the targeted application.

Despite the aforementioned vulnerabilities of biometrics, we need to counterbalance with situations where biometrics are in any case more secure than passwords: weak passwords, bad-managed passwords, password-based authentication deactivated by the user. Many Information System administrators complain about users writing their password on a Post-It® note stuck under their keyboard or even on their computer's screen. Many mobile phone users leave the default PIN code (e.g. 0000, 1234) to unlock the phone or even deactivate this security feature considered as counter user convenient. Too many passwords, to be memorized, are short and explicit hence could be easily guessed with a simple dictionary attack [32] or more sophisticated attacks [84].

Thus, in most cases involving non security-aware users in an environment requesting a minimum of security, the use of biometrics will provide a "weak, but easy" security tool.

CHAPTER 5
Fingerprints in Details

Contents

5.1 Introduction

Fingerprint recognition is the oldest and most deployed biometric technique, both in civil and criminal applications, because of its high maturity and cost-effective capture and processing. Different biometric market studies clearly show the domination of fingerprint, more than 65% by adding "AFIS" and "Fingerprint" in figure 5.1.

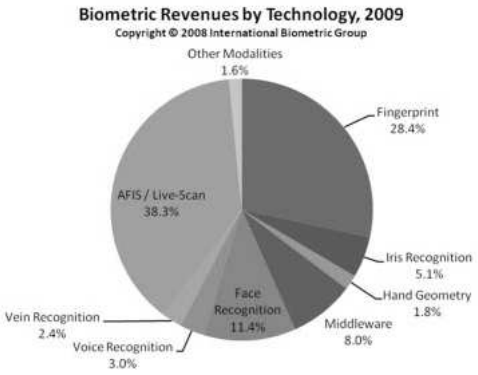


Figure 5.1: Biometric Market

The particular interest for fingerprints in the criminal area is because of latent impressions that remain on objects that are touched or handled. These are a deposited residue made up of a combination of perspiration, organic solids such as amino acids, and inorganic solids such as salts, blood or other susceptible material the finger might have touched recently.

5.2 Galton's Classification

A fingerprint is a set of skin lines, locally parallel, named *ridges* and empty space between two consecutive ridges named *valleys*. The three global shapes of this pattern, devided in arches, loops and whorls, are the first level of information we may examine to classify fingerprints, see Figures 5.2 and 5.3. The average value of ridge to ridge frequency is of about half a millimeter and the average value of valley to ridge height is of about 0.1mm. By convention, the fingerprint image is displayed as the trace the inked finger would leave on a paper, or, in other words, as the latent print of the finger. Of course this first level information is useless to proceed with fingerprint verification. Fingerprint classification is mandatory for an efficient research for matching candidates within large databases such as AFIS and 1:N systems in general.

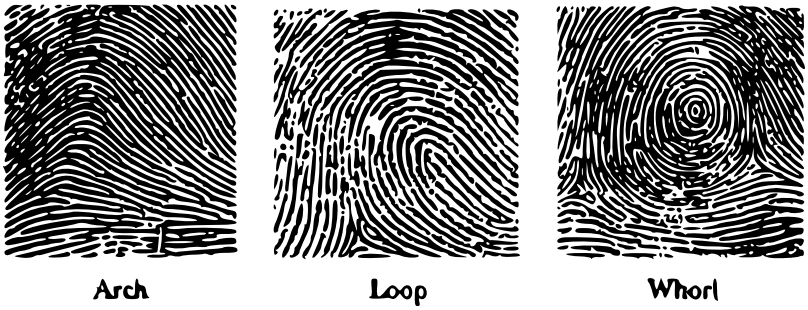


Figure 5.2: Fingerprint Characteristics - 1st level: Classes

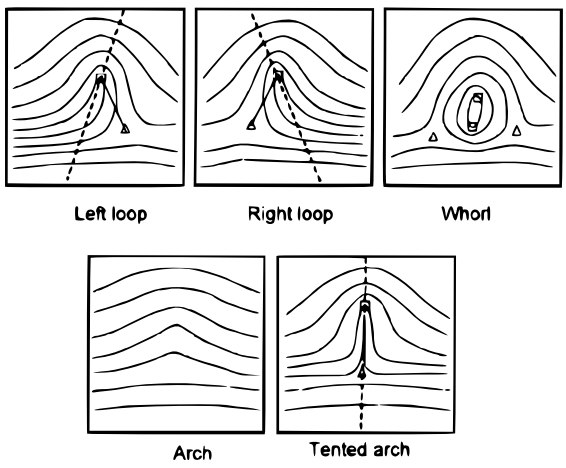


Figure 5.3: Extended Fingerprint Classes

Table 5.1 shows the non-uniform distribution of the three classes within the population.

Classe	Representativeness
Loop	65%
Whorl	30%
Arch	5%

Table 5.1: Fingerprint Classes Distribution

5.3 Galton's Details

The second level of information is the so-called *minutiae*. These are specific points of the fingerprint where a ridge is ending or bifurcating. Tens of such points may be extracted from a fingerprint, and are enough to proceed with reliable fingerprint verification. This is the way criminal sciences have been conducting fingerprint identification for more than one hundred years. Other, but not sufficient, second level information are *core(s)* and *delta(s)* location, see Figure 5.4. The pattern of ridges and valleys, with its minutiae, core(s) and delta(s) are unique to each individual (different even for identical twins) and this pattern is known to be stable during the lifetime. Usually a correct matching between only eight to twelve minutiae is enough to conclude with a positive fingerprint recognition.

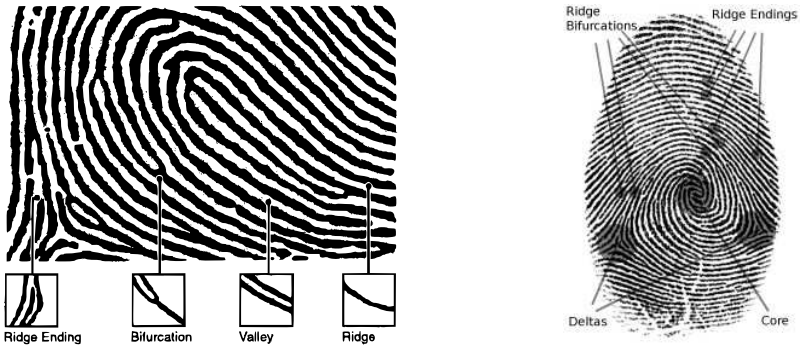


Figure 5.4: Friction Ridges, Minutiae, Core, Deltas

5.4 Pores

The third level information is *pores* location along the ridges, see Figure 5.5. The use of pores location is young, and coming with the improvement of new generation fingerprint sensors, able to capture such details. As of today, fingerprint recognition algorithms using this technique are not mature enough to replace minutiae-based ones, but promising [63]. Regarding fake fingers techniques described in the next chapter, we are able to reproduce first and second level information. Copying third level of information on a fake finger is still a challenge, hence often cited as a potential fake finger countermeasure.

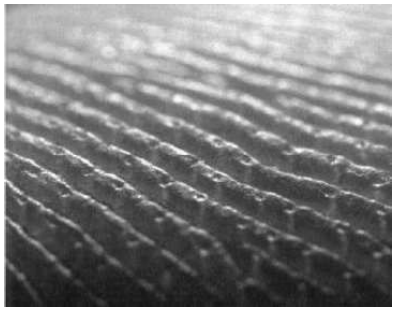


Figure 5.5: Pores along Fingerprint's Ridges

5.5 Existing Standards

5.5.1 Finger image data

The reference standard is ISO/IEC 19794-4 [59]. In comparison to the other techniques described below, the image level is the most interoperable format. The dominant image formats are WSQ for 500 dpi images and JPEG2000 for 1000 dpi images. The Wavelet Scalar Quantization (WSQ) grayscale fingerprint image compression algorithm [17] is the standard for the exchange of 8-bit, 500ppi fingerprint images within the criminal justice community. The average template size in this format is of about 12 kB. This is the format stored in biometric ePassports.

5.5.2 Finger minutiae data

The reference standards are ISO/IEC 19794-2 [57] and ANSI INCITS 378 [1]. This is the most widely used representation. Basic information stored per minutiae are x and y location (figure 5.6), angle value (figure 5.7) and minutia type (ridge ending, ridge bifurcation, unknown). These four values are stored using either 5 bytes (full format) or 3 bytes (compact card format). ISO sets the maximum number of stored minutiae per finger at 60.

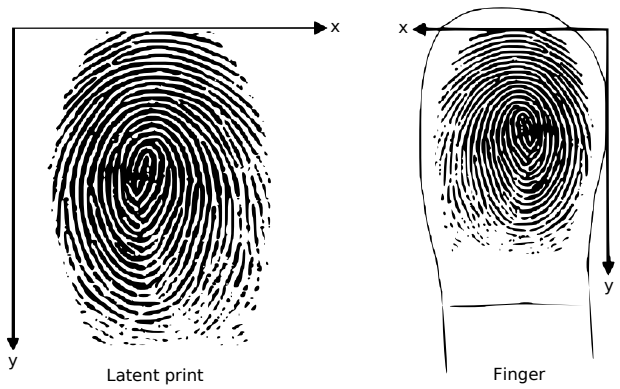


Figure 5.6: Coordinates system for minutiae positioning

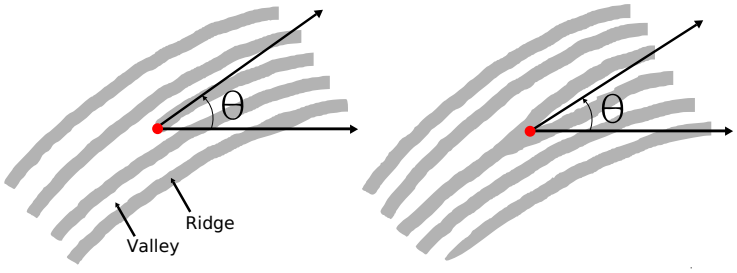


Figure 5.7: Minutia angle coding

Optionally, one may store additional information such as core(s) and delta(s) location, or ridge-count. Ridge count is depicted in figure 5.8: for each pair of minutiae, the system will count how many ridge lines are crossed by the direct line between these two minutiae. However recent evaluation [48] shows poor efficiency of ridge count in relation to its high cost: imperceptible improvement in error rates for a template size about 5 times larger.

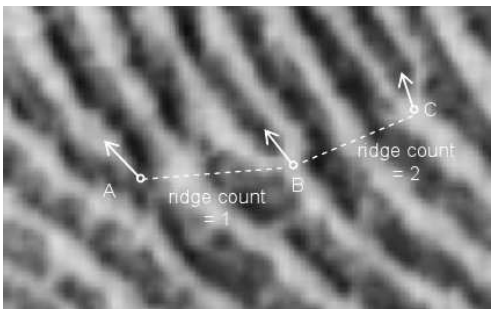


Figure 5.8: Example of Ridge Count

ISO 19794-2 is the format stored in most national ID cards and embedded electronics systems. The average template size in the compact card format is of about 200 bytes (without options), twice for the full format. We will discuss the efficiency of such formats in chapter 12, related to MINEX evaluation.

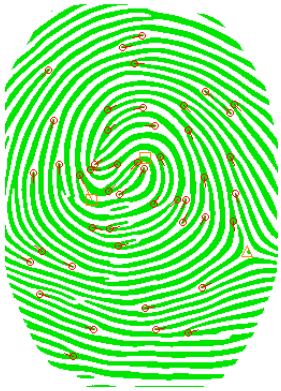


Figure 5.9: Minutiae, cores and delta extracted from fingerprint

5.5.3 Finger pattern data

These standards are somewhat anecdotal since never really adopted by a large majority of the biometric community in opposition to minutiae or WSQ images.

Finger pattern spectral data

The reference standard is ISO/IEC 19794-3 [58]. This standard describes a technique where a fingerprint image is partitionned in smaller cells containing only few ridges, each cell being then coded with three values: wavelength λ , angle θ , and offset δ (see figure 5.10). This is coming from frequency spectral analysis such as Discrete Fourier Transform, adding multiple sinusoidal functions. The average template size in this format is of about 400 bytes.

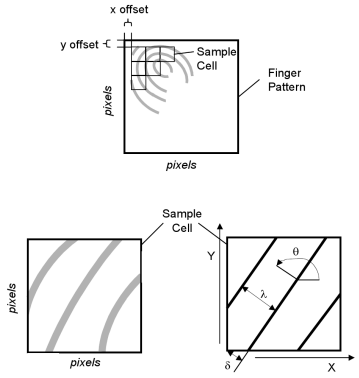


Figure 5.10: Pattern Spectral data

Finger pattern skeletal data

The reference standard is ISO/IEC 19794-8 [60]. This standard describes a technique where ridge lines are represented by single-pixel segments, each line being coded from a starting point, named virtual minutiae, to its ending point with consecutive segments, each segment being relative to the previous one by a change in direction. The average template size in this format is of about 500 bytes. This standard was initially pushed at normalization because of its compressed size in comparison with full compressed image and its ability to be used by both minutiae and pattern spectral techniques for interoperability.

Figure 5.11: Pattern Skeletal data (on the right)

CHAPTER 6
Sensors Technologies

Contents

6.1 Introduction

A complete overview of fingerprint sensors technologies can be found in [71]. Different captures of the same fingerprint image (or biometric trait in general) will never give exactly the same image. This is mainly due to the different technologies available to automatically capture a fingerprint, and even with exact same sensor, images will never be the same. One family of sensor will give black ridges over a light grey background, whereas another technology will give light grey ridges over a white background. About twenty years ago, automatic devices to "cleanly" capture fingerprints were developed to replace the *ink&paper* technique. Most of today's sensors have a resolution of about 500dpi. The different technologies available are based on many properties, including optical properties of the skin surface, electrical properties of the skin (silicon-based sensors, e.g. capacitive), thermal properties of ridges in friction with a pyroelectric material, pressure of the ridges on a piezoelectric material, ultrasonic fingerprint relief measurement. This list being non-exhaustive. However, the current predominant technologies on the market are optical ones and silicon-based ones (both capacitive and thermal), thus we will give further details about these latter techniques.

6.2 Optical technologies

Most of the optical sensors on the market are based on light reflection on the fingerprint. The user just puts his finger on a glass substrate and the reflection of light going to the imaging component is modified, giving dark image points for the ridges and light image points for the valleys (figure 6.1, letf). Recent, more compact, technology uses light propogation on the surface of a finger in close contact with the imaging electronic device through a thin optic fiber layer. (figure 6.1, right).

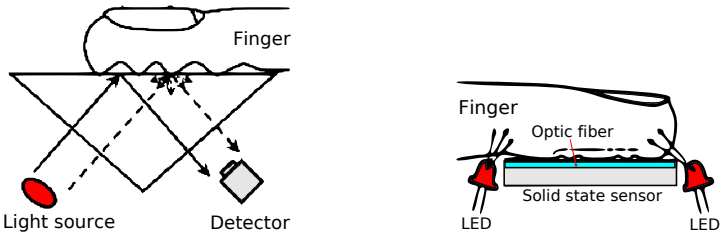


Figure 6.1: Infrared FP sensors (left: reflection, right: propagation)

Another marginal optical technology, and most bulky, is based on light transmission through the finger, the light source being placed above the finger, on the nail side, and the imaging component be positioned under the finger, on the fingerprint side (figure 6.2).

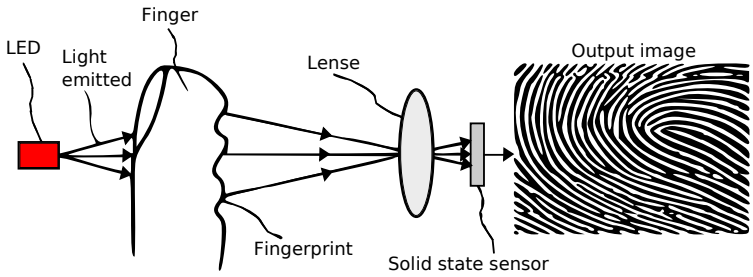


Figure 6.2: Infrared FP sensors (transmission)

A more recent technique is based on multispectral illumination of the finger, being known to be resistant to fake fingers by analysing skin reflection properties on a wide range of colour wavelengths [102].

6.3 Silicon-based technologies

6.3.1 Capacitive Sensors

Capacitive measurement of the fingerprint is the most used technique in embedded electronics. This results from the sensors being compact and the ability to produce such sensors with regular semiconductor processes in industry. The skin and the surface of the sensor are used to build a

capacitor, both being one of the electrode plates of the capacitor. The sensor itself is a matrix of very small capacitor plates, each being a pixel of the captured image. Charge and discharge microcurrent cycles of each capacitor are analysed to measure if the skin is directly in contact with the sensor or not, hence providing a map of the fingerprint.

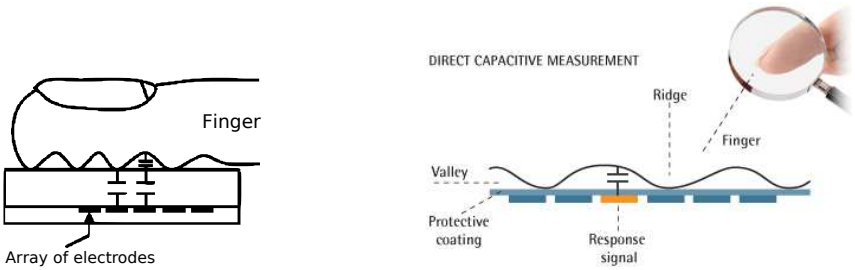


Figure 6.3: Capacitive FP sensor

6.3.2 Field Effect Sensors

Field effect sensors are actually a variant of capacitive sensors, sometime called active capacitive sensors. These sensors are characterized by an electrically conductive ring around the sensor plate. When capturing a fingerprint, a current will be applied to the ring, hence generating an electric field that will "charge" the finger. Starting from the principle that the electric potential field in a dielectric region follows the shape of nearby conductive surfaces, an array of sensors (actually micro-antennas) placed in the dielectric can remotely measure the shape of the conducting surfaces. Inverting the previous ring principle, each micro-antenna will be charged by a current generated by the electrical field running through it. The ring's current can be dynamically varied by real-time controls, to optimize detection of various types of fingerprint features. Another claimed advantage is the ability to capture good images even with worn fingerprints, the sensing principle reading the ridges and valleys in the live layer of skin cells located just beneath the dead cells that make up the skin surface (i.e. at the Epidermis-Dermis junction).

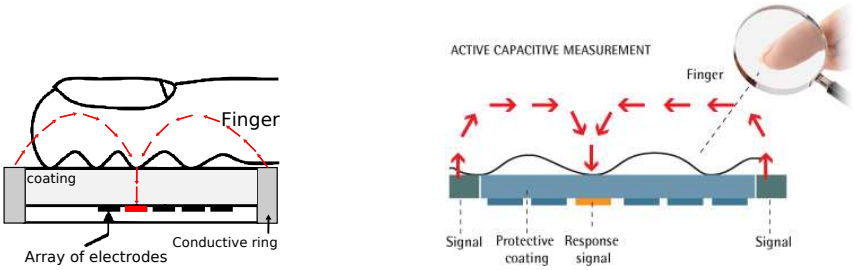


Figure 6.4: Field effect FP sensor

6.3.3 Thermal Sensors

Another widely used silicon-based technique is the thermal one. Actually, this technology uses a combination of silicon and pyroelectric material (i.e. material generating an electric signal proportional to the heat), while remaining with classical semiconductors manufacturing techniques. This technology is only available in the swipe form-factor (see section 6.5). For technical reasons due to the property of pyroelectric material, the requirement of uniform heating of the material very quickly when placing the finger leaves no time to measure differences between ridges and valleys. Hence, the technique is to swipe the finger across a small slice of silicon, friction of the ridges will heat the pyroelectric material, whereas valleys will have no effect on the pyroelectric material. However, we may note that Atmel finally stopped the manufacturing of such a device in 2008 after more than ten years of R&D and business development efforts.

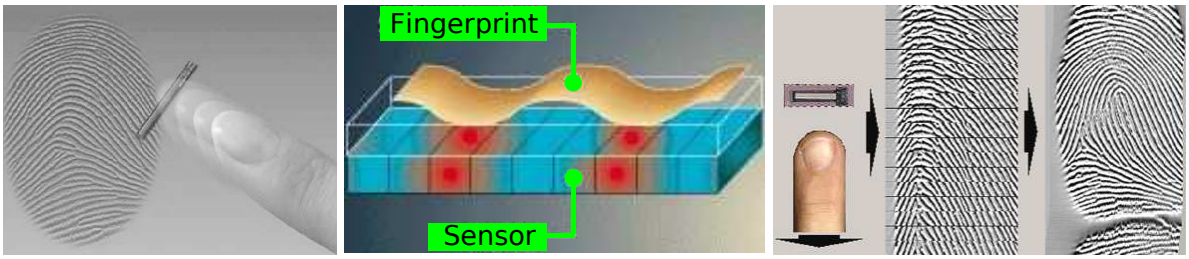


Figure 6.5: Atmel swipe thermal FP sensor

6.4 Other Technologies

Other silicon-based, but very young, techniques (or polymer-based) use MEMS technology (MicroElectroMechanicalSystems) [22], electro-optical technology or pressure-based technology. A totally different technology uses ultrasonic fingerprint imaging [13], and has been in the field for many years but is still at a development stage. This latter technology claims to be secure against fake fingers because of reading the fingerprint structure in the dermis, the sub-surface of the skin, rather than the surface.

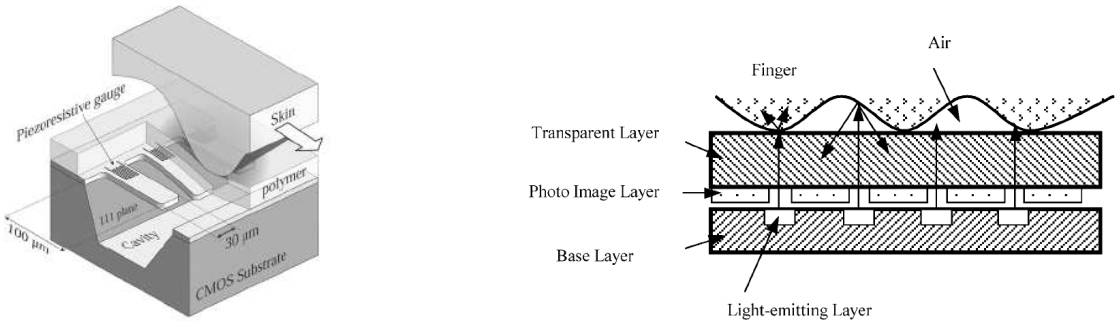


Figure 6.6: Other silicon-based FP sensors

6.5 Form-factors

A silicon-based fingerprint sensor may come in two different form-factors: the touch (or full matrix, or 2D) sensor or the swipe (roughly speaking, 1D) sensor. The touch sensor is very easy to use, but suffers from several drawbacks: its silicon area size makes it expensive, its surface may leak a complete latent print (trace left by a fingerprint, may be used for an attack), with heavy usage its surface became dirty, hence difficulties to capture an image, and its size makes it difficult to integrate in small embedded electronic devices.

On the other hand, a swipe sensor may be considered as less user-convenient, because the user needs to learn how to correctly swipe his finger across the sensor to obtain a good image. However the correct gesture comes in a minute of training. This form-factor has several advantages: cost-saving (more sensors on one silicon wafer), no latent print issue, no cleanness issue (due to the applied friction), and very easy to integrate in small electronic devices such as laptops, PDAs, mobile phones, USB dongles, etc...

The swipe technique uses a fast capture of a lot of slice images during the swiping, and then a dedicated algorithm is here to reconstruct the correct image from the tens of slices. This appeared a little bit crazy when invented by the Atmel company (formerly Thomson-CSF semiconductors) in the mid-nineties [72] but, as of today, every silicon-based sensor manufacturer comes with this form-factor in their product range. However, constantly reducing the silicon area, hence reducing information from a finger, will reach a feasability limit regarding the smallest acceptable size for a fingerprint sensor [73].

6.6 Thin Flexible Sensors

The emergence of polymer electronics allows the development of thin flexible fingerprint sensors with ultra-low power consumption, and these two characteristics are ideal for application in smart cards and other embedded electronic devices. Another promising characteristic is the ability to adapt to the fingerprint curve, hence capturing more information on the finger, just like the rolled-finger technique. However, since the early 2000's, several start-up companies (and dedicated structures of major electronic companies) have risen and fallen in business development of such sensors, with the major issue being the immaturity of the technology in opposition to its promises.

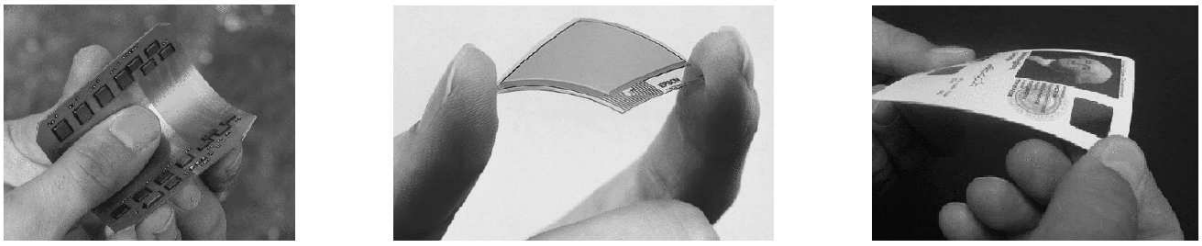


Figure 6.7: Flexible polymer-based FP sensors

CHAPTER 7
Dummy Fingers

Contents

7.1 Introduction

Recently, in March 2008, the Chaos Computer Club, an activist group in IT, published the fingerprint of the German Interior Minister [96]. The print was included in more than 4,000 copies of the March 2008 issue of the magazine, which was published by the CCC. The image was printed two ways: one using traditional ink on paper, and the other on a film of flexible rubber that contains partially dried glue. The latter medium can be covertly affixed to a person's finger and used to leave an individual's prints on doors, telephones or biometric readers.

In [124], a description of attack paths are presented either with finger owner cooperation (molding techniques, 3D) or without his cooperation (use latent fingerprint and Printed Circuit Board technique to create the mold, 2D). This paper shows examples of fake fingers built either with silicone or stamp-type rubber. Fake fingers could also be build with gelatin [77], play-doh like material (polymer clay) [108], wood glue [26] or latex [3]. We build fake fingers with all these materials and had pretty good results with different sensor technologies, however, apart from gelatin, we add few troubles to obtain easily reproducible attacks with silicon-based sensors, such as capacitive one. In this case, we usually need to find out some tricks such as blowing on the sensor, or using water spray, before capturing to enhance the humidity coupling between the fake finger and the sensor. Please note these known tricks are also useful to enhance the quality of the image with optical fingerprint sensors.

We will describe all our techniques and tests in this chapter. The purpose of this work, sponsored and funded by French governmental agencies, being the ability to build up easily reproducible attacks with a lot of long-life fingerprint copies on the shelf to address French certification needs for fingerprint systems at medium term, as the ones currently conducted for smart cards' security evaluation (common criteriae). This work targets fingerprints at first glance because of its domination of the current market and its particular ability to leave traces. Similar research programs targeting other deployed biometric technologies are in the pipe.

7.2 Negatives by Molding

Negative referred to as the cast in which we will mold the *positive*, referred to as the fake finger. When having the user cooperation, the straightforward way to obtain his fingerprint is to use classical molding techniques. We usually find the necessary general public and low cost material in do-it-yourself or leisure dedicated shops. Here is an exhaustive list of different materials we used, a description of most of them can be found at "http://en.wikipedia.org" or "http://www.homecrafts.co.uk". The trademark name is generally used.

Siligum

Siligum is a bi-component paste (soft silicone, blue + hardener, white) for accurate molding. Siligum has a rapid setting time of 5 minutes and molds all objects within 10 minutes. The cast has the advantage of being still flexible and resistant after hardening, ideal for removing a positive. We obtain a mold with very high details of the fingerprint. The mold is reusable multiple times but the used material is not reusable to mold another fingerprint. Siligum is ideal for 3D molding of the entire fingertip.

Plasticine

Plasticine is an oil-based clay usually used in animation. Very easy to manipulate, we rapidly obtain a mold with high details of the fingerprint. However, we have to take care of the mold when using it because of its constant softness: it never dries, hence the material can be reuse multiple times to mold anotherbeing finger. However a shaped mold may be used only few times.

FIMO/Sculpey

Fimo is a polymer clay containing PVC. Once shaped, Fimo is baked in a standard oven for about 30 minutes at 110°C to harden it. This material is not very convenient to manipulate, not so soft, but gives good results in fingerprint's details. The shaped mold is reusable multiple times, the used material is not reusable.

Patarev/iClay

Known as "world's softest clay", Patarev is a polymer modeling compound, close to the well known Play-Doh, but with a more convenient texture and is very lightweight. After few hours, the mold is hardened, however remaining a little bit flexible just enough to ease the removing of a positive. This gives a very good result in fingerprint's details. The shaped mold is reusable

multiple times, the used material may be reused (let the dried material for one day in a humid towel).

Alginate

Alginate is the most used material for molding in leisure activities, legal medicine, dental impressions. Alginate is simple to use by blending alginate powder and water, then immersing a fingerprint into the mixture for less than one minute. Molds can be used immediately. This material is not convenient with its grainy texture and sticks to fingers. The fineness of details is poor at ridges and valleys scale. The shaped mold shows a lot of defects and we definitively decided to not use this material.

UtilePlast

UtilePlast is a modeling plastic, delivered in very small balls. Just put the desired quantity of white balls in boiling water for seconds, then remove when clear, shape at will and wait a few minutes for it to harden and cold. This gives correct fingerprint's detail, however the material is very hard when cold and this sometimes difficult to remove the positive. The shaped mold is reusable multiple times, the used material may be reused (just boil it again and again).

Natural latex

In hobbycraft, latex comes in the form of a liquid or soft paste. For fingerprint molding we use the pasted texture, whereas we will use the liquid form for producing positives. We obtained not average quality details with this material and its needs few or several days to dry, depending on the thickness of the mold: definitively not convenient. Once hardened, the mold is very flexible and resistant. The shaped mold is reusable multiple times, the used material can't be reused. Warning: some people have a serious latex allergy, and exposure to latex products such as latex gloves can cause anaphylactic shock.

Candle wax

Industrial candle wax is usually made of paraffin and stearine, whereas handcraft candle here made of beeswax. Molding fingerpint is very easy and ultimately cost effective but the mold is very fragile and, of course, can't be heated. We just put a fingerprint in the medium hot wax and wait for solidification in a minute. We obtain good details of the fingerprint. We usually break the mold to remove the positive, then the used material is reusable.

Bi-component repair paste

Generally used in home maintenance and available at any home depots in the glues shelf, this material has the texture of stone once shaped and dried. After mixing the two components in hands for few minutes we obtain a nut of clay-like material, mold a fingerprint and let dry for few hours. Not so convenient to manipulate (sticky), we obtain acceptable fingerprint's details. The shaped mold is reusable multiple times, the used material can't be reused.

Other tested materials

We tested many other materials with no acceptable results. We may cite among them different rubber cements generally used in construction or soldering metals.

In summary

Siligum, Platicine, Patarev and Fimo are the most convenient materials. Figure 7.1 shows obtained results with selected materials, whereas table 7.1 gives a global overview of all tested molding materials.

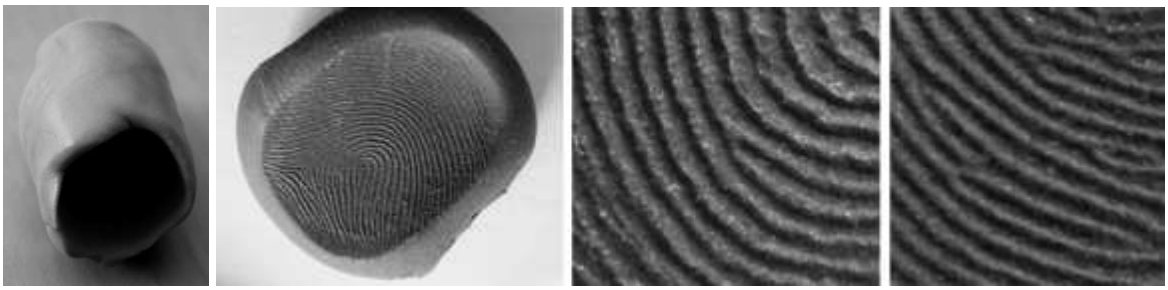


Figure 7.1: Molds and Details

per 100gr	Price (€)	Molds	Appreciation	Setting time	Remark
Siligum	4	6	+++	5 min	resistant, flexible
Plasticine	1.5	10	++	1 min	no hardening, reusable
Fimo	2.5	7	+	30 min	hard material
Patarev	10	25	++	half day	light flexibility
Alginate	6	20	--	5 min	not convenient
UtilePlast	5	8	+	30 min	hard material, reusable
Latex	2	10	-	few days	not convenient
Candle Wax	1	20	-/+	10 min	very fragile, reusable
Repair Paste	20	20	+	half day	hard material, porous

Table 7.1: Fingerprint molding materials

7.3 Negatives by Latent Print and Etching/Grinding

Building a mold from a latent print or digitally captured image is the issue when not having the fingerprint's owner cooperation. A latent print could be retrieve on a glass or paper and revealed, depending on porous or non-porous substrate, with dedicated powder [112], cyanoacrylate fuming [127], eosin [111] or ninhydrin [109].

During our experiments we start by using a captured fingerprint image to mature our technique of creating a mold using Printed Circuit Board techniques. We classically use photolithography and chemical etching technique:

- We print a negative image (ridges are white on a black background) of the fingerprint on a transparent sheet
- We use this transparent as a mask on a cooper board coated with UV-sensible varnish and put to UV light
- We reveal and fix the positive image (cooper=ridges, varnish=valleys) in the varnish with classical photo chemical
- We etch the bare cooper with ferric chloride, varnished part being protected from etching
- We obtain the desired mold (the ridge-to-valley relief being defined by the thickness of copper, classical values are 17.5μm on Kapton (0.1mm flexible sheet), 35μm or 70μm on epoxy boards.

Beyond this old fashion technique, we had the chance to have access to a modern PCB prototyping equipment: manufactured by LPKF company, this equipment is able to directly grind the copper on the board (no chemicals, no risks, clean) to create the desired electronic circuit. We decide to investigate the use of this brand new equipment to build our fingerprint molds. We rapidly face one major issue: the equipment is dedicated to mechanical drawing formats such as classical electronic layout Gerber or Autocad's .dxf, all these being vectorized images, no bitmap supported! Whatever, we spend some times to investigate the vectorization of bitmap-based fingerprint images and after many trials we finally obtain the best result with the free and open source InkScape software on Linux to vectorized a fingerprint skeleton bitmap to export to .dxf format. We were then able to directly grind the ridge skeleton, finally obtaining the desired mold. An interesting feature here is the capability to set up the equipment to also grind the epoxy substrate (z positioning of the grinding tool), hence having more ridge-to-valley relief than the only copper thickness in opposition with etching technique, we hence obtained more detailed positives.

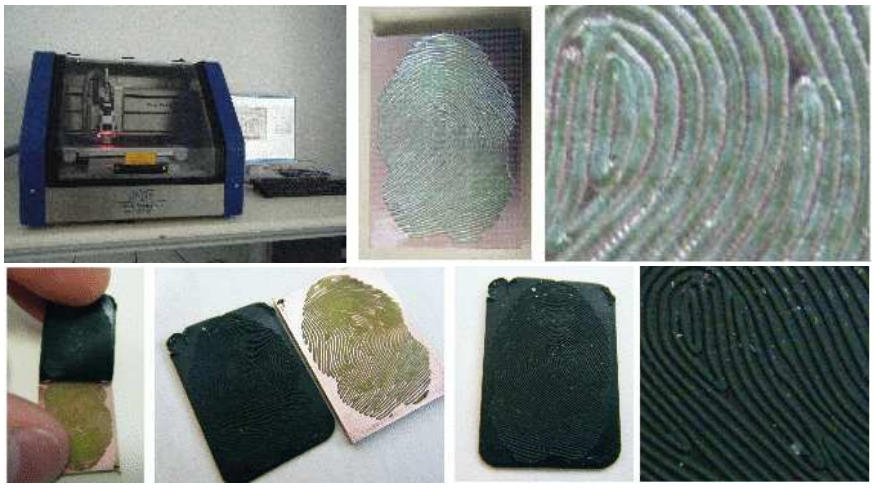


Figure 7.2: PCB mold, positive and details

7.4 Positives by Pouring

A *positive* is the fake finger itself. It reproduces the exact copy of the targeted fingerprint at the same scale, at least in two dimensions. The third dimension, copying the height of the ridge may be variable and optional depending on the targeted fingerprint sensor. Here we pour some liquid material in a mold and wait for some times to let the liquid dry. Depending on the pouring material and the mold material, we may be at room or hot temperature. The thickness of the desired fingerprint copy may vary from a tenth to few millimeters or complete three dimension copy of the fingertip depending on the targeted sensor technology. As for molds, we usually find the necessary general public and low cost material in do-it-yourself, leisure dedicated shops or cooking shelves. Here is an exhaustive list of different materials we used:

Wood Glue

Just pour some woodglue into a cast and let it dry for a day, we obtain a white and soft fingerprint copy. One of the first technique described in any related literature. Few experiences with woodglue are just using an inverted fingerprint image on regular paper with laser or inkjet printer as a mold and the woodglue fingerprint copy directly pass on old optical sensors.

Natural Latex

We may just pour liquid latex in a cast and wait for at least a day or more, depending on the thickness of the latex layer. Latex drying may be very long. A prefered technique is to lay down latex with a paintbrush in the cast, activate drying for one minute with a hair dryer and let dry for just few hours. Manipulating a single layer of latex (too thin) is very difficult, we advise to remove the copy with a piece of tape and use it directly on sensors. For an "autonomous" copy, the best technique is to proceed with few or several layers: once dryed, lay down another layer of latex, let dry, redo as many times you need. A thin copy is light brown and enough transparent, a thick copy is more brown and less transparent. The copy is resistant and flexible. A latex copy pass a lot of sensors.

Latex-based glue

Not so transparent and flexible as natural latex but easier to manipulate and dries faster. The fingerprint copy is light brown coloured, just as skin, and both thin and thick layers are possible. This glue is originally dedicated to textile.

Silicone Glue

Silicon glue, originally dedicated to watertight joints in bathroom or kitchen, is usually coming in transparent or white material and is not easy to manipulate. However the result is quite good, the fake finger as strong details and is both flexible and resistant. A silicon variant comes in the form of rubber stamp (needs special dedicated equipment, not do-it-yourself).

High flexibility glue

This brand new family of glues (polymer-based) is dedicated to flexible objects under twists and vibrations such as shoes (repair insole-textile connexion) or any textile or plastic in vehicules,

toys. Pour the glue in the mold and wait for a day, this gives a very good fake finger.

Decoratives paints

Also known as windows color or 3D paints, these paints are dedicated to glass and porcelain. Coming in a wide range of colors, once dried this give a thin flexible layer being repositionable. Some of this paints are very liquid, hence perfect to pour on fingerprint mold and copy detailled shape. Needs a day to dry and easy to remove from PCB or Siligum molds. This gives flexible and resistant fingerprint copy.

Gelatin

A complete study of gelatin-made fingers, so-called gummy fingers, can be found in [76, 77]. Gelatin have the advantage to be extracted from animals' tissues and thus have chemical properties close to human skin and different materials produced by human body (e.g. sweet, grease), in opposition to latex, silicone or other material. This is the reason why this material is particularly efficient with silicon-based fingerprint sensors. However, it suffers from a critical drawback: a gelatin-made thin fake finger is very limited in time, it must be used within the hour of production, after this delay gelatin becomes totally dry and twisted like shown in Figure 7.3. A gelatin-made thick fake finger must be used within the day of production, after this delay gelatin begins its decomposition like shown in Figure 7.3. Gelatin is derived from the partial hydrolysis of collagen (the most abondant protein in mammals).

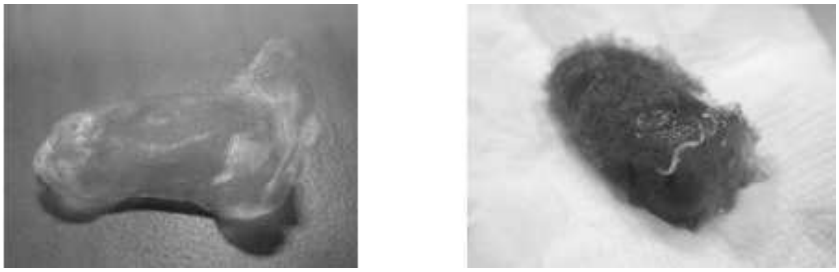


Figure 7.3: Gelatin - dry thin layer, rotten thick layer

Glycerin

During our work we identified and disclose a, let's say, brand new material in this field: glycerin. We essentially identified glycerin for these characteristics: rot-proof and softness over its lifetime period, the two main drawbacks of gelatin-made fake fingers. For the experiment, we are using baby's suppositories, originally addressing baby's constipation. Actually this material is made of about 86% glycerin and 14% gelatin and is very easy to use: use a lighter or a microwave oven to heat the suppository directly in the cast to liquefied it, then refrigerate for few minutes to solidify the fake finger, see Figure 7.4. Here we were using wax, silicone, FIMO paste or any classical molding material to build a negative of the finger.

Figure 7.4: Glycerin - left: thin layer - center: thick layer - right: 3D models

Rabbit's skin natural glue

One of the oldest glue, made from smashed rabbit's skin. Contains natural collagen with added glycerin for stabilization, coming in the form of granules to dissolve in water (for few hours). When the mixture is ready, it may be used just as gelatin or glycerin: pour on the mold, put in a fridge for ten minutes. The fake finger with high details is skin-colored, flexible and rot-proof.

Other tested materials

During our experiments we used other (almost each type of) glues (e.g. epoxy, cyano, neopren) and other materials (e.g.: siligum, plasticine, polymer clay) with no interesting results.

In summary

All described material has its own advantages and drawbacks. Figure 7.5 shows obtained results with selected materials, whereas table 7.2 gives a global overview of all tested molding materials. For 100gr of product we may obtain several tens of thin-layer fakes, setting time is also given for a thin-layer sample.

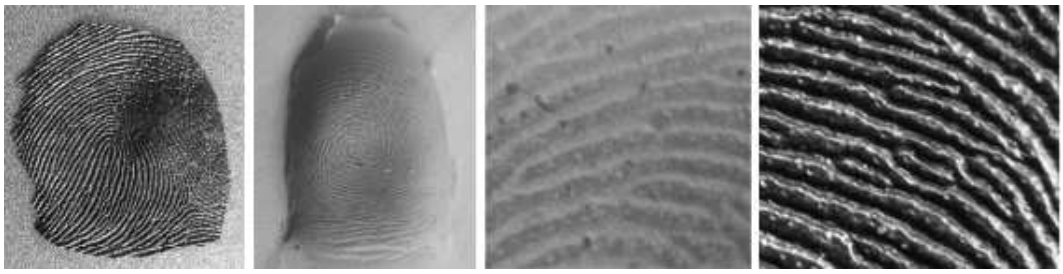


Figure 7.5: Fakes and Details

Of course, some molds and fakes materials are incompatible. Materials with the same basis tend to stick together (e.g. silicon glue in siligum mold), sticky materials in porous molds (e.g. any glue in bi-component repair paste). Table 7.3 gives a list of impossible/inadvisable/advisable associations.

per 100gr	Price (€)	Appreciation	Setting time	Remark
Woodglue	5	++	1 day	the more thin, the more fragile
Latex	3	+++	1 day	flexible, resistant, skin-colored
Silicon glue	5	-/+	1 day	flexible, resistant
Flexible glue	30	+++	1 day	flexible, resistant
3D paints	10	++	1 day	flexible, very good details
Gelatin	7	+	30 min	fragile, short usability
Glycerin	3	+++	10 min	fragile, reusable
Rabbit Glue	4	+++	30 min	the more thin, the more fragile

Table 7.2: Fake fingerprint materials

	Siligum	Plasticine	Fimo	Patarav	Repair Paste
Woodglue	advised	ok	ok	ok	inadvised
Latex	ok	advised	ok	ok	ok
Silicon glue	impossible	advised	ok	ok	advised
Flexible glue	advised	ok	ok	impossible	inadvised
3D paints	advised	ok	inadvised	impossible	inadvised
Gelatin	advised	impossible	ok	ok	ok
Glycerin	advised	impossible	ok	ok	ok
Rabbit Glue	advised	impossible	ok	ok	ok

Table 7.3: Mold versus Fakes materials

7.5 Positives by Printing

Classical

Just printing the image of the fingerprint (and not the latent, vertical symmetry) on regular white paper with plain black ink (ink-jet or laser) may pass optical sensors. For a better result, due to optical coupling between sensors'glass and copy substrate, a transparent sheet is prefered. During our experiments we had success with just placing a thin smooth transparent glycerin layer between the sensor and the fingerprint image on regular paper.

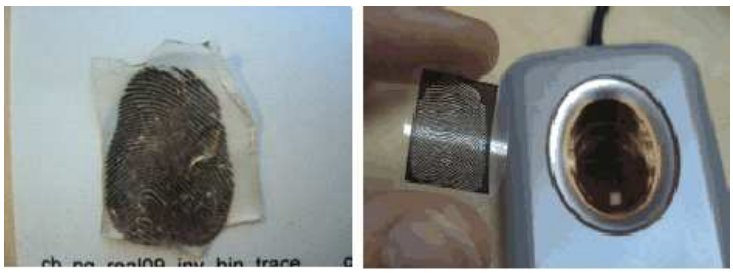


Figure 7.6: Faking with classical prints

Conductive InkJet

We had the chance to have access to a brand new manufacturing prototype (in-house MicroPackS) dedicated to the so-called *Polymer Electronics*, *Organic Electronics* or *Printed Electronics*. This equipment is a particular inkjet printer, able to deposit a polymer ink charged with a scalable amount of silver (or other conductive material) nano-particles on different types of substrate (e.g. Kapton, PET, PVC) to obtain desired electrical conductivity. Depending on the natures of ink, nano-particles and substrate we may build basic electronic elements such as diodes or transistors. This technology is coming from years of research in OLED (Organic Light Emitting Diodes) and really starts with the Nobel prize in Physics to Alan Heager in 2000. Printed Electronics is believed to be the next step after the current all-silicon for the electronic industry: low-cost process, flexible substrate (plastics, textiles), power efficient electronic functions: ideal for embedded electronics.

We then intended to use this state-of-the-art equipment to print positive images of fingerprints and try our samples on the different fingerprint sensing technologies. As already explained, we are able to control both the color and the conductivity of the ink, and the color and nature of the plastic (or glass) substrate (e.g brown, white; opaque, semi-transparent, transparent) opening the way to multiple possibilities, hence spending a lot of time to print and test samples. For the moment we focus on silver ink on kapton and transparent PET with interesting results as described in few sections (see 7.8).

Figure 7.7: Jetpac conductive printing

7.6 Positives by Etching/Grinding

As already described, we used PCB prototyping to build a three-dimension mold from any two-dimension fingerprint latent image and also had success in directly printing the fingerprint image. We then decided to try to etch or grind the direct fingerprint image in copper, both on epoxy boards and flexible kapton, and tried any other conductive support (e.g. aluminium on PET). The idea here is to take advantage of the conductivity of copied ridges on silicon based sensors and obtain robust samples, even on swipe sensing (previously seen silver ink deposit

is fragile regarding friction on swipe sensors). Preliminary results will be described in 7.8, however we are still in experiments for the time being.

7.7 Samples Characterization / Certification Issues

Samples characterization is a big issue for certification purposes since we need detailed and robust elements to be picked up on-the-shelf when a fingerprint system comes for evaluation without loosing time to reproduce all necessary samples. For instance, we have equipments to profile a mold or a fake to ensure the quality of details (e.g. valley-to-ridge height) and evaluate the potential loss of these details after each use of the sample (especially when swiping the fake). We currently measure a valley-to-ridge height of about $50\mu m$ for a mold and $30\mu m$ for a fake (figure 7.8).

We also need to evaluate the best preservation conditions of samples: light, temperature, humidity, pressure and so on. This conservation environment will be particularly sensible on fakes made of gelatin, glycerin, collagen and conductive ink. During our experiments we store samples in a clean room (controlled environment dedicated to silicon manufacturing) at 22°C . Several months later, all samples (except gelatin of course) are still usable.

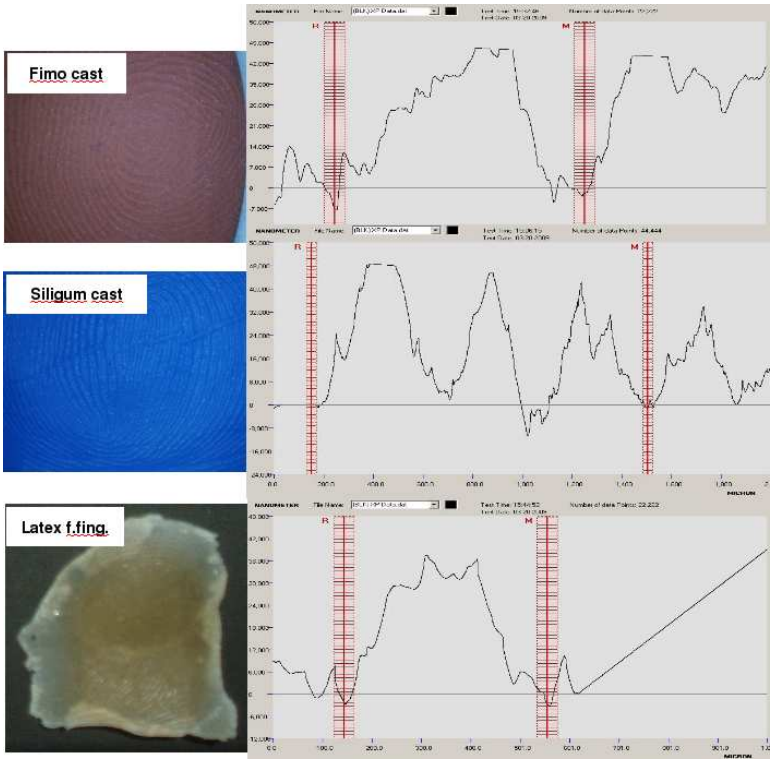


Figure 7.8: Molds and fakes characterization (ridge flow profile)

7.8 Samples Test on Sensors

During our experiments we had access to a large panel of fingerprint sensors, two having an aliveness detection system (#1: multispectral optical, #2: infrared skin surface transmission). The range of sensors covers optical, capacitive and thermal swipe sensors. We used two fingerprint recognition softwares available for download on the web at [55, 83] (see figure 7.9), these programs are able to compare fingerprints from image files or directly from a wide range of supported fingerprint sensors. Several sensors come with their own software suite but too often replacing the logon screen of Windows, not very practical for our tests. However, in a standardized evaluation the sensor will be tested with its own recognition software, if provided. Table 7.4 gives an overview of results.

Figure 7.9: Software tools for testing

Wood Glue

Woodglue fakes are robust and just enough flexible to perfectly stick to the contact surface of sensors. Faking optical sensors, even with the #1 countermeasure, is easy. Thermal swipe sensors are also easily faked. By humidifying the contact surface (just blow on, or use water spray) of the capacitive sensor, we may pass the fake (this trick is working for all “dead materials” i.e. all described materials but gelatin, glycerin, rabbit collagen). Very thin layer on a real finger, however semi-transparent white, may enable a very discrete attack.

Latex

Both natural and glue latex give positive attacks on optical sensors, even with #1 and #2 countermeasures. May pass capacitive sensors by humidifying the contact. However latex is useless with swipe sensors, its texture being too elastic and tends to deformate the ridge flow under friction. But we had occasional success by, once again, humidifying the contact surface to ease the swiping movement. Very thin layer on a real finger, skin-coloured, enables a very discrete (even perfect) attack.

Silicone Glue

Here we use usual white silicone. We succeeded with optical sensors but failed with optical countermeasures, may succeed with capacitive sensors and thermal swipe sensor with same tricks as above (silicon have the same elastic texture as latex). We also succeeded with capacitive sensor by applying silver lacquer (only on ridges)on the samples. Usual sample thickness is of about 1mm, can’t build very thin layer (0.1mm) in opposition to woodglue and latex.

High flexibility glue

This gives same results as silicone (same texture, same thickness, usually semi-transparent), but is far more convenient to manipulate when creating the fake.

Decoratives paints

This gives a fake layer close to woodglue but a little bit more flexible, enough robust to pass thermal swipe sensors. Countermeasure-free optical sensors and capacitive sensors with usual trick are faked.

Gelatin

Gelatin-based fake fingers easily pass optical (even with #1 countermeasure) and capacitive sensors naturally (no tricks needed). However can't be used with swipe sensors because of its sticky texture.

Glycerin

Glycerin-based fake fingers give same results as gelatin-based, but the preparation of the fake finger is more convenient and the lifetime period is indisputable as already seen.

Rabbit's skin natural glue

Same results as gelatin and glycerin, but naturally skin-colored, maybe the reason why it passes also on optical sensor with #2 countermeasure. During our experiments we usually add some more glycerin to the mixture for smoothness and lifetime stabilization.

Jetpac Prints

For the time being, we only proceed with very first trials, and currently working on substrate and ink colors for better results on optical sensors. Curiously we suceeded with #1 countermeasure optical sensor, and not regular sensors, with silver ink on transparent substrate. Naturally succeed on capacitive sensors, but useless with swipe sensors since the friction damage the silver ink.

In summary

For an evaluation, the choice of the fake finger material is depending on the targeted sensor. In the real world, this choice would also depend on the environment of the attack (needed discreteness). Latex and rabbit skin glue (with added glycerin) would be our first choices. Beyond the fake finger itself, several tricks are successful (e.g. blowing on the sensor surface, silver lacquer to enhance electrical conductivity, nail lacquer to strengthen smooth materials).

Table 7.4 gives an overview of obtained results.

	Optical	Capacitive	Thermal	Opt. with #1	Opt. with #2
Woodglue	pass	may pass	pass	fail	pass
Latex	pass	may pass	may pass	pass	pass
Silicon glue	pass	may pass	may pass	fail	fail
Flexible glue	pass	may pass	may pass	fail	fail
3D paints	pass	may pass	pass	fail	fail
Gelatin	pass	pass	fail	pass	fail
Glycerin	pass	pass	fail	pass	fail
Rabbit Glue	pass	pass	fail	pass	pass
Jetpac/Kapton	fail	pass	fail	fail	fail
Jetpac/Clear PET	fail	pass	fail	pass	fail

Table 7.4: Fakes test on Sensors technologies

7.9 Our Contribution

A part of our work has been published in [9]. The complete work has been reported in different technical reports of two funded projects. We obtain further results than presented here, these reports being either confidential (VulnBio, funded by DCSSI) or restricted (Asfip [4], funded by ANR). See Appendix A.

7.10 Conclusion

Obtained results clearly prove the lack of security of current fingerprint sensors. As already discussed, this is an issue only for unstaffed terminals, and eventually for systems with a control at distance (video screening). The only possible countermeasure would be the utopian *perfect aliveness detection*, able to detect both fake fingers and dead fingers. We will discuss this in chapter 9.

CHAPTER 8
Electronic Fake Fingers

Contents

8.1 Introduction

Electronic Fake Finger stands for computer-aided artificial generation of fingerprint templates or images. Beyond physically attacking the fingerprint sensor, one could digitally attack the matching module to falsely obtain access to the biometric system: this means either providing a false template to the matching module or a false image with forged information (e.g. minutiae characteristics) to the extraction module (refer to figure 4.1), but in order to cheat with the matching module with the extraction output. We may attack a biometric system just like we are used to attack a cryptographic system to evaluate its strength. This type of attack intrinsically depends on the FAR of the system: a system with FAR of 1 out of 1 000 will need an average of 500 real templates to gain access to the system (FAR = 0.1% is a generally chosen value by system administrators for a corporate use).

8.2 Brute Force Attack on Fingerprint Templates

The storage format of a template is usually known, based on ISO or ANSI standardization (seen in section 5.5). Thus we may attack the system by randomly generating thousands of

ISO templates and propose each one to the matching module, with hoping a success in several iterations. Being purely random, this attack is thus inefficient by generating a lot of improbable templates (e.g. all minutiae in only one quarter of the image space), then going beyond the theoretical value of 500 iterations for a 0.1% FAR system.

8.3 Dictionary-like Attack on Fingerprint Templates

Large enough fingerprint image databases (hundreds of images) are available (e.g. DVD included with [74]). By analogy with dictionnary attacks in cryptography, we are then able to send 500 images to the extraction module, having a high probability to gain access to the system. This approach has the advantage to not depend on the knowledge of the template format, hence being suitable to full proprietary system using somewhat secret representation of the original fingerprint.

8.4 Hill-Climbing

Colin Soutar in [113] firstly described the so-called hill-climbing attack, using the score returned by the matching module to improve the candidate image at each iteration, but did not disclose quantitative results. Uludag and Jain in [121] described another approach of the hill climbing attack on templates, falsely gaining access the system in an average value of 271 iterations. Here is the process of this classical hill-climbing:

- pick 100 minutiae template at random
- send each one to the matcher
- keep the best matching candidate (highest score) as the reference
- modify this best candidate by either
  - add a minutia
  - remove a minutia
  - modify a minutia
  - swipe minutiae
- send this new candidate to the matcher: keep it if better, else throw away.
- iterate until success

8.5 Synthetic Fingerprint Template Generation

In the ASFIP project, we were able to largely improve hill-climbing attacks by taking advantage of very large fingerprint databases owned by one of the partners, and applying statistically obtained information on minutiae data for each fingerprint class (e.g. average minutiae locations and local orientations). By using these statistics for both initial template generation and minutia modifications, we then obtained access to a system in less than 150 iterations since we avoid here a lot of useless minutiae modifications such as minutia placement in statistically "almost empty" regions or incoherent minutiae angle regarding its neighbourhood.

8.6 Synthetic Fingerprint Image Generation

8.6.1 SFinGe

SFinGe stands for **S**ynthetic **Fin**gerprint **Ge**neration. This tool is developed by David Maltoni's team in the university of Bologna, Italy. Both documentation and software are available in [74]. The primary goal of SFinGe was to propose the building of very large fingerprint images databases to overcome with limited access to moreover insufficient public databases. The second goal was to enhance both the population and sensors representativeness of the databases: the tool may create a fingerprint image of desired class, erosion, contrast, skin deformation, background and orientation.

Figure 8.1: Two sets of fingerprint impressions generated by SFinGe

8.6.2 Optel

The polish company Optel [13], developping ultrasonic fingerprint sensors (already seen in the sensors related chapter), also proposed a synthetic fingerprint image generation tool available for download at *www.optel.pl*. The generation algorithm is based on a proprietary mathematical model of finger ridge patterns and only very few information are available. However, in opposition with SFinGe, this tool lacks of realistic fingerprint representation: no image processing to modify the original image to take usual defaults into account, such as erosion, background and contrast. Apart from the advantage of existing, on our opinion this software is useless in an evaluation context.

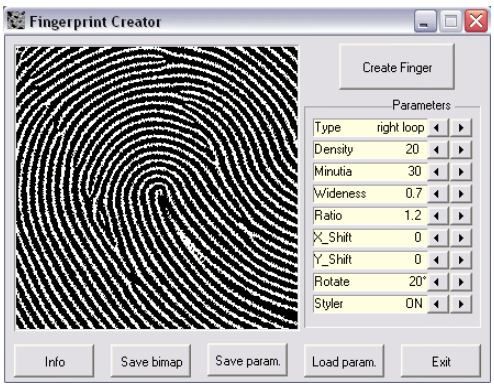


Figure 8.2: Synthetic fingerprint generation by Optel

8.6.3 ASFIP

In the Asfip project we propose a brand new approach for synthetic fingerprint generation. Based on works from Fleury and Watanabe [40, 41], the model is using a complex morphogenesis algorithm which simulates the phenomenon of ridges creation during embryonic growth. The ridge pattern is due to surface tensions and mechanical forces at certain locations (e.g. region of proeminent pulp, nail borders) on original skin-flat fingers between the seventh and sixteenth weeks of gestation. This tool is able to build ridges that are perpendicular to two points of constraints (e.g. one point on each nail border), with many iterations of this process, we are able to construct a full fingerprint image. Once the image is constructed, as SFinGe, the tool is able to apply different image processing to simulate real-life fingerprint capture. Figure 8.3 is an example of obtained fingerprint images.

Figure 8.3: Synthetic fingerprint generation by Asfip - Original models & real-life simulation

8.7 Reconstructing Fingerprint Image from Minutiae Template

It has been traditionally assumed that the minutiae template of a user does not reveal any information about the original fingerprint. Ross *et al* [100, 101] and Maltoni *et al* [21, 20] challenge this notion and show that three levels of information about the parent fingerprint can be retrieved from the minutiae template alone: the orientation field, the class of the fingerprint and parts of the friction ridge structure. These information are retrieved from both minutiae angle information and statistics about minutiae density for each class (see figure 8.4). Examples of retrieved information can be seen in figures 8.5 and 8.6.

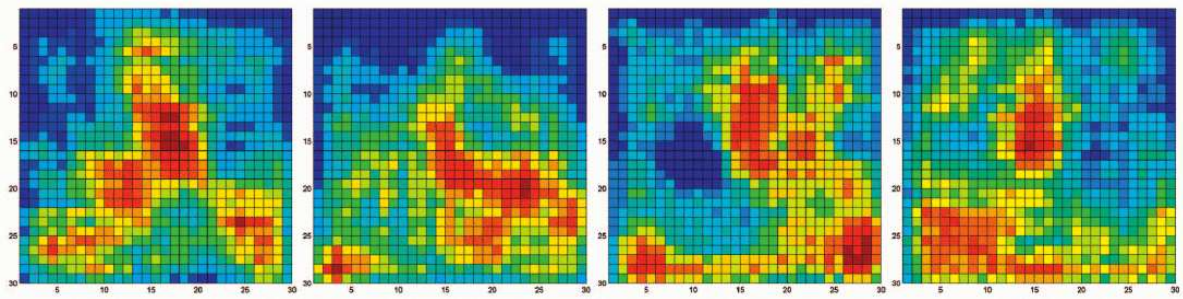


Figure 8.4: Minutiae density map for arch, whorl, left loop, right loop

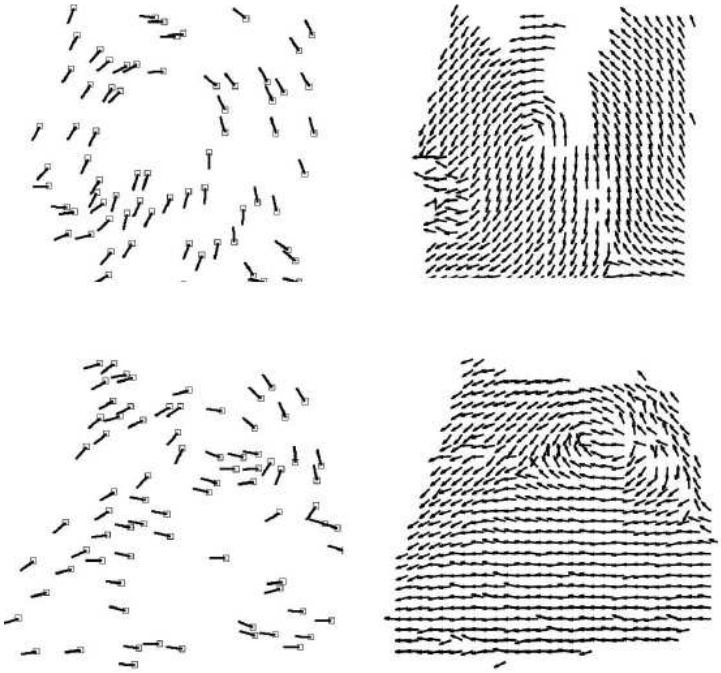


Figure 8.5: Direction map from minutiae template

Figure 8.6: Direction map and ridge structure from minutiae template

8.8 Our Contribution

In opposition to previous works, our approach in the ASFIP project wasn't the *re*construction of *the* owner's fingerprint image from its minutiae set but rather the (virtual) construction of *a* fingerprint image around *the* owner's minutiae set. This results in a different fingerprint image but still exactly containing the same minutiae with associated characteristics x, y and θ and $type$: the (virtual) candidate image will hence match with the (real) reference minutiae template. This class of attack is known as *Masquerade Attack*. However for the time being these images can't visually cheat with an experienced human eye, but pass a lot of recognition algorithms (see figure 8.7)

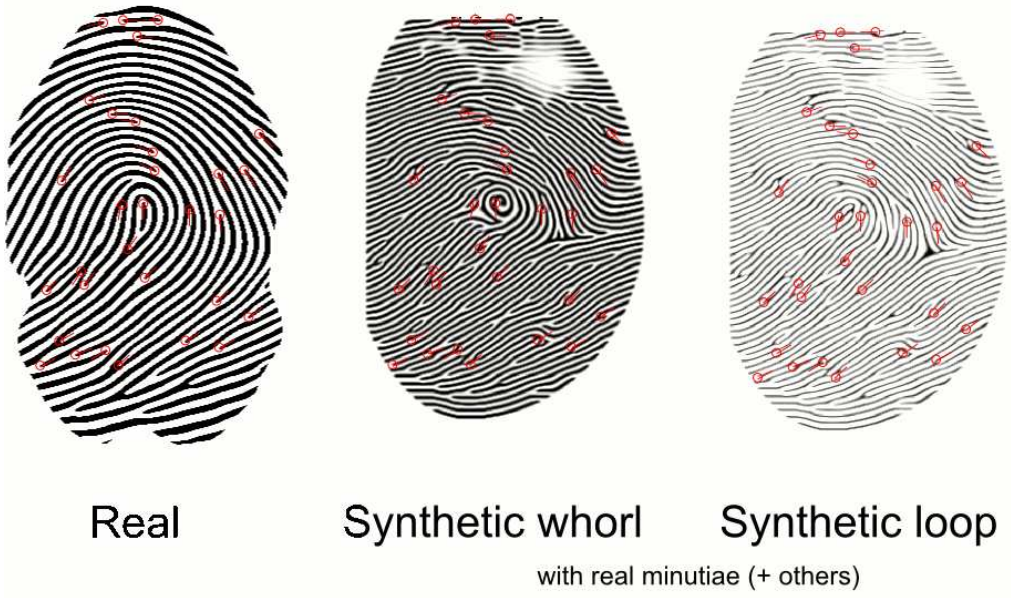


Figure 8.7: Reconstructing fingerprint image from minutiae template

8.9 Conclusion

The class of hill-climbing attacks have a major limitation: this is possible if and only if we have access to the score of the matching module either by direct access or indirect access (side channels, just like in cryptography). Brute force shows a number of iteration relative to the FAR of the recognition system but doesn't need the score. Classical countermeasures such as long reactivation delay after few or several negative trials may prevent such intensive attacks.

With the only knowledge of the template we may produce a non-invasive attack just by building a synthetic fingerprint from these minutiae and creating a fake finger from it to present to the fingerprint sensor.

In an evaluation context, the synthetic generation of both images and templates as a major advantage: the ability to build very large databases for testing purposes with no privacy issues related to these data.

CHAPTER 9
Aliveness Detection Systems

Contents

9.1 Introduction

An interesting study can be found in [104]. Aliveness Detection (aka Liveness Detection or Vitality Check) stands for the ability of a biometric system to detect if the provided sample is coming from dead or dummy material. Even more challenging, a perfect system should be

able to detect, at verification stage, if the provided living sample is really attached to the enrolled person and not just another living human. This can be performed either at the acquisition stage (hardware counter-measures), or at the processing stage (software counter-measures). For years, several companies in biometrics imaging field were falsely claiming vitality check, however it has been publicly shown that artificial fingerprints, static facial images and static iris images may fool many systems. From a couple of years, we see the advent of real aliveness detection features to efficiently complement imaging systems. Regarding hardware counter-measures possibly built in sensors, we can cite temperature of the skin, optical properties of the skin, pulse oximetry, blood pressure, blood presence, electric resistance of the skin and relative dielectric permittivity. Regarding software counter measures we can cite skin deformation during finger placement, perspiration detection, pores location. However, even if working properly, an aliveness detection system always results in a more complex acquisition stage (e.g. long process, false alarms with real fingers), hence the recognition system is prone to higher False Rejection Rate, not applicable in convenience-oriented systems and dedicated to security-oriented systems. This chapter will focus on aliveness detection in fingerprint verification system, where the challenge is to process this checking only with the small part of the fingertip in touch with the contact surface of the fingerprint sensor. One will even better understand the complexity of the challenge once stated that epidermis is actually a dead material!

9.2 Nature of the Measurable Characteristic

9.2.1 Living Properties

The most straightforward idea is to check a relevant property of a living person: body heat, cardiac pulse, blood circulation, skin conductivity, breathe, movements and so on. Of course the capturing of such information shall be convenient, non intrusive and thus without any pain/damage to the end user. Ideally this aliveness detection is totally transparent to the end user.

9.2.2 Non-Living Properties

The idea here is to individually detect materials used in fakes production instead of detecting living properties. This detection of non-living properties induces the management of a blacklist with retrievable and repeatable data inherent to latex, woodglue, gelatin and so on. However such a detection system will need to be constantly updated with newly used materials or derivation of a known material (e.g. latex may come in the form of pure liquid latex used for objects molding, latex-based glue used for textile or paste used for face/body make-up special effects in cinema) with different physical properties. The wide range of materials and the limited time for a fingerprint capture make such countermeasures difficult to implement, this is the reason why most aliveness detection systems actually try to check a living property instead of a non-living property.

9.2.3 Living voluntary stimulus response

This class of vitality check is generally easy to implement, and just need the user cooperation. For instance one may ask the end user to gently move and/or rotate the fingertip on the contact

surface regarding fingerprints. If applicable to such or such biometric technique, one may ask the end user to react in a determined manner to any tactile, visual or audio stimulus.

9.2.4 Living involuntary stimulus response

This class of detection is generally less convenient and acceptable for the end user. For instance this may be fingertip reaction (e.g. skin contraction) under smooth electrical stress (or muscle reaction just like in electromyogram), pupil retraction in front of a violent flash light, eyelid closing reaction to flash light or air pulse.

9.2.5 Measurable physical characteristics

The nature of the measured physical characteristic could be of different origins: optical (e.g. skin or blood light absoption), thermal (e.g. skin temperature and thermal conductivity), electrical (e.g. resistance/conductivity, complex impedance), mechanical (e.g. skin distortion under pressure), chemical (e.g. biochip analysing latent fluids of the finger, detecting the presence of sweat, grease, proteins, collagen).

9.3 "Sensor-natural" Aliveness Detection

It is important to note that the original goal of automated fingerprint capture systems was to cleanly replace ink & paper technique. Most of the first optical sensors, arriving on the market in the late nineties, were just designed to digitally capture an acceptable image of a fingerprint. Aliveness detection was definitely not a concern for sensors' developers at that time. Moreover, fingerprint capture was only considered within an attended (staffed) environment (border control, forensics). This is the reason why mediatic buzz such as Matsumoto's attacks sounded a little bit ridiculous to experts in the area.

9.3.1 Optical

Most of the optical fingerprint sensors on the market are based on Frustrated Total Internal Reflection (FTIR) using a glass prism, where the finger comes in contact with one side of the prism, a light source comes to a second part of the prism and a CMOS camera will receive transmitted light on the third part of the prism. The index of refraction of glass and a finger being the same (1.5), ridges tend to absorb light, whereas valleys, hence air with an index of refraction of 1 will reflect the light on the other side of the prism. Because FTIR devices sense a thee-dimensional surface, they cannot be easily deceived by presentation a printed fingerprint image. Beyond FTIR theory, we have however previously seen such sensors could be defeated with 3D copy of fingerprint and even 2D copy by interfacing thin water (index of refraction = 1.33) layer or thin glycerin layer at the interface between the contact surface and the finger.

9.3.2 Capacitive

This is the mostly used technique in silicon-based (aka solid-state) sensors. The technology measures here a difference in the magnitude of electrical charges stored in microcapacitors between the silicon component and the finger. Hence this is depending on the nature of the

dielectric between the silicon and the skin (ridges: coating of the sensor, valleys: coating of the sensor + air). Table 9.1 gives examples of static Relative Dielectric Permittivity (RDP) approximate value at room temperature of selected materials.

Reading table 9.1 clearly explains why gelatin and glycerin are so easily captured on capacitive sensor, the same for humid latex or silicone.

Material	RDP
air	1
paper	3
silicone/rubber/latex	4
dry finger	20
alcohol	25
normal finger	30
wet finger	40
glycerin/gelatin	40
water	80

Table 9.1: Relative Dielectric Permittivity of used materials

9.3.3 Field Effect

Here the signal transmitted by the drive ring is known to be modulated by the derma structure (subsurface of the finger skin). Measuring the ridge flow directly in the living part of the finger protects this technology against fake fingers, however state-of-the-art shows that gelatin-based fake fingers gives pretty good fingerprint images. One may superimpose thin copies of the same fingerprint with different material just to simulate a change in the material structure at the intersection of each copies.

9.3.4 Thermal

The pyroelectric effect being produced by the friction of ridges on the sensor during swipe movement, any 3D copy with the same friction capabilities would succeed. However this technology needs a large temperature differential between the sensor (heated to be far above finger temperature) and the finger, hence the fake finger must be close to external body temperature, and not at room temperature to obtain a correct image.

9.3.5 Pressure

Only an intrinsic protection against 2D copies here, since one may easily guess any 3D copies will fool the imaging system.

9.3.6 Ultrasonic

Ultrasound sensing may be viewed as a kind of echography. The technology is able to measure changes in acoustic impedance at the intersection of air and epidermis, then epidermis and

dermis, then dermis and fat substructure of the finger. This intrinsic protection against fake fingers may be fooled by a superimposition of different materials just as depicted above with field-effect sensors.

9.4 Aliveness Detection by Additional Hardware

This class of aliveness detection techniques is generally considered as costly but rather efficient.

9.4.1 During Acquisition

This category is more an enhancement of the image capturing technique to better check vitality with available hardware, or additional hardware if using a different approach than the one used for capturing the fingerprint image.

Temperature

The average temperature of the finger is generally of about 10°C above the ambient temperature. The idea here is to measure the temperature of the presented material at the contact surface of the sensor to detect latex or any other fakes being at room temperature [36], however one just needs to warm the fake at desired temperature (e.g. five minutes between closed hands). Thermal conductivity is also considered. The wide range of application environments (dry and cold countries, or wet and hot) and human characteristics dispersion induce a high tolerance about this kind of measurement.

Electrical Conductivity / Finger Impedance

Measuring the electrical resistance (static current) or complex impedance (alternative current) of the finger is another straightforward idea [36]. But once again depending on the environment and human nature this measurement must be very tolerent (because of large differences between dry and wet fingers). Generally, just humidifying the fake is enough to bypass such a countermeasure.

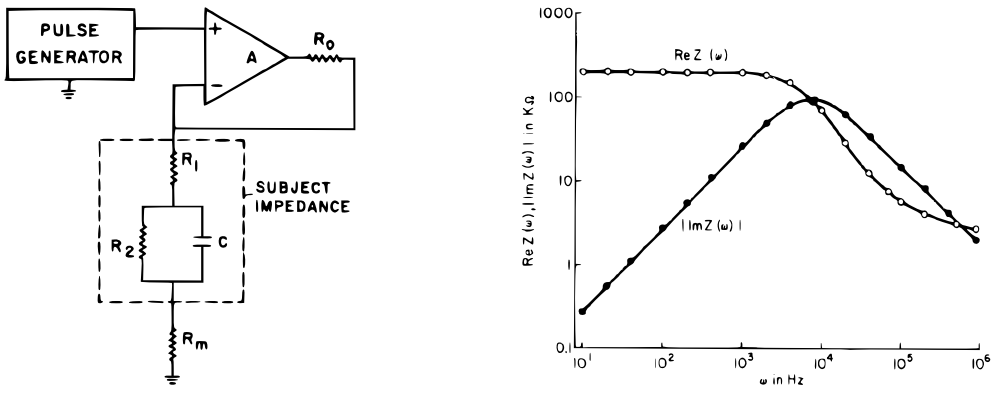


Figure 9.1: Finger electrical model and impedance graph [97]

Electromagnetic Permeability

Just like field-effect sensors, the idea is to measure a relevant change in the environment permeability with the presence of living tissues of the subcutaneous finger [82]. This detection is bypassed by presenting a real finger just behind a thin fake copy.

Dielectric Permittivity

Already discussed in a previous section about intrinsic aliveness detection within capacitive fingerprint sensors, just humidifying a fake may bypass this detection. Another known technique, if the functioning range is tight, is to wet the fake with a mixture of alcohol (RDP=25) and some water (RDP=80), by natural alcohol evaporation this liquid will span from RDP 25 to RDP 80 then reaching the classical RDP value of a finger (of about 30/40) and this will enable the capturing process of the fingerprint image [104].

Skin Optical Properties

Optical properties (absorption, transmission, reflection) of human skin may be checked under different lightning conditions (e.g. UV, blue, green, red, infrared). Similar to the multispectral imaging technique [102], we have seen this is fooled by latex or gelatin. Another optical based detection claimed as LFD (Live Finger Detection) [43] was easily bypassed.

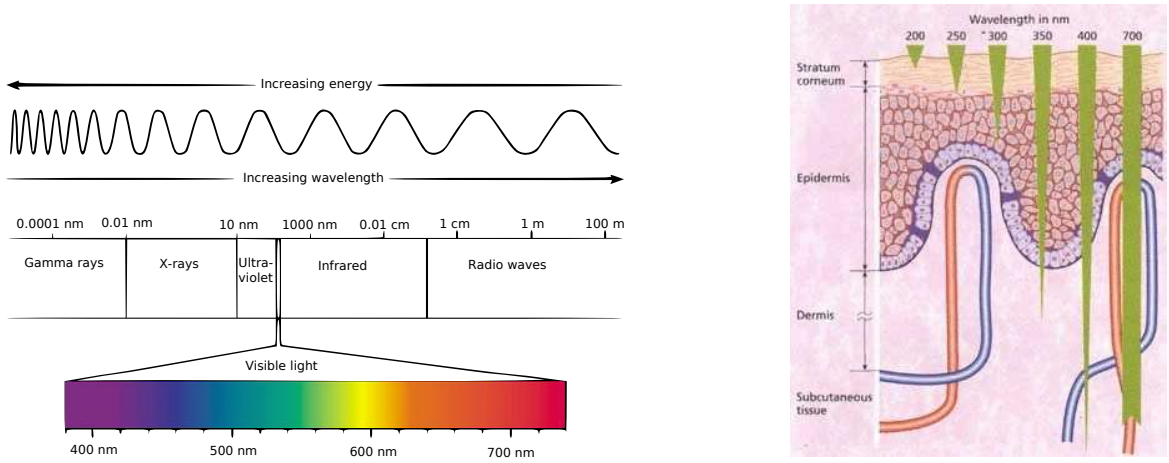


Figure 9.2: Light spectrum and human skin properties

9.4.2 Pre or Post Acquisition

Here there is a clear split between aliveness detection feature and image capturing feature, by means of different hardwares.

Odor

Maltoni *et al* [7] propose a novel method using “electronic nose” (i.e. biochip able to detect the presence of a desired chemical component, such as silicon-based protein or DNA microarrays

[51]). They prove distinguishing between a real finger and a fake latex finger, however one may think about scrubbing the fake with his real finger to lay down the wanted chemical component.

Heart Pulse

One approach is based on the analysis of fine movements of the fingertip surface, which are induced by volume changes due to the blood flow [37]. A US patent entitled *Anti-Fraud Biometric Sensor that Accurately Detects Blood Flow* by SmartTouch LLC describes how two light emitting diodes (LEDs) and a photo-detector are used to determine whether blood is flowing through the finger. This aliveness detection method basically implements pulse oximetry (see right next section), but only uses the pulse rate information. Just presenting a real finger just behind a thin fingerprint copy allows to bypass this detection. Moreover pulse rate may be very low (e.g. 50 pulses/minute) thus requiring several seconds only to measure a relevant pulse rate, hence a long process, not convenient for the end user.

Pulse Oxymetry

Pulse oximetry is a non-invasive method allowing the monitoring of the oxygenation of a patient's hemoglobin. A sensor is placed on a thin part of the patient's anatomy, usually a fingertip or earlobe and a light containing both red and infrared wavelengths is passed from one side to the other. Based upon the ratio of changing absorption of the red and infrared light caused by the difference in color between oxygen-bound (bright red) and oxygen unbound (dark red) blood hemoglobin, a measure of blood oxygenation can be made [95]. Once again this is a long process and just presenting a real finger just behind a thin fingerprint copy allows to bypass this vitality check.

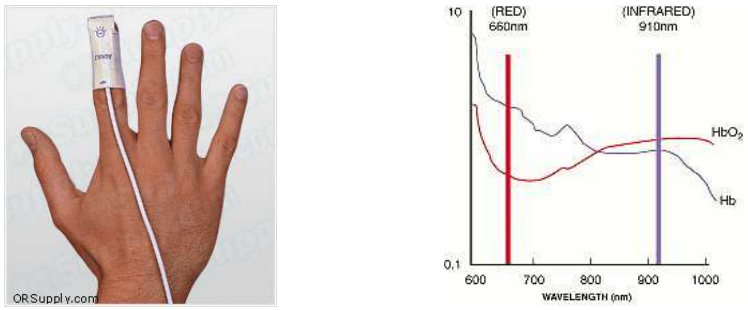


Figure 9.3: Pulse Oximetry

Blood Pressure

Similar to the two previously described techniques, this is even more complicated when only using fingertips: since this requires measurement at two different places on the body, we must use one fingertip on both hands. Not convenient and bypassed by previously seen technique.

Finger Optical Properties

The idea here is to measure optical characteristics of different body components deeply inside the finger: presence checking of arterial blood, venous blood, water, lipids, melanin. And once again this is fooled by presenting a real finger just behind a thin fingerprint copy.

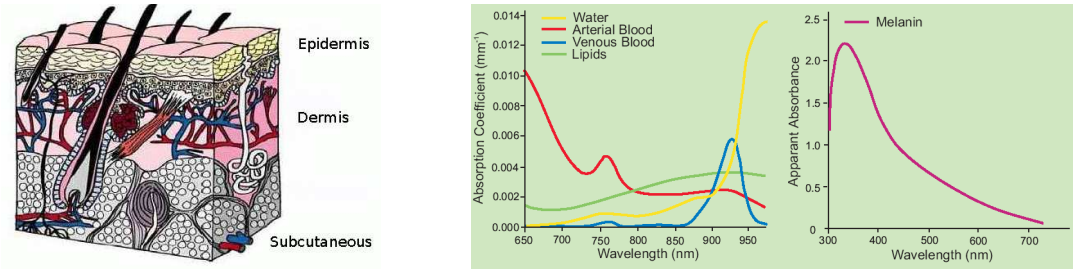


Figure 9.4: Finger components spectrometry

9.5 Aliveness Detection by Software

This class of aliveness detection techniques is generally considered as cost-effective but with limited efficiency. The *static* category refers to only analyzing information in the captured image, whereas *dynamic* category refers to active measurements during the imaging process.

9.5.1 Static

Linear Traces in Printed Images

The detection of technology-inherent printing patterns enters in the non-living properties category and is only able to detect printed fingerprint. Straightforward, no more to say about it and with a very limited scope since most fingerprint copies come from handycraft activity.

Pores Detection

Pores detection along fingerprint ridges in the image comes from the idea that classical molding techniques aren't able to copy such small details. However we had some interesting (visual) results in effectively copying pores onto positive copy of fingerprints, but having no pores-based aliveness detection software at disposal, we simply guess knowing the right methodology and materials to reproduce pores location along ridges.

Image Quality

Few proposals suggest to use the standardized NIST fingerprint image quality software to reject bad images often obtained with fake fingers in non-expert hands. As already seen many times, just humidifying the fake will result in a pretty good image both on optical and capacitive sensors. Moreover this approach would for sure results in a very high FRR or FTE in a system having to deal with elder people, manual workers and so on. Each sensor product giving its

Figure 9.5: Fakes with pores clearly visible (ridges appear in white here)

particular fingerprint image, we may guess this aliveness detection technique strongly needs fine tuning dedicated to such or such particular sensor.

Image Statistics

Fingerprint aliveness detection based on the wavelet analysis of the fingertip surface texture and background noise in the image [80] shows promising experimental results. The authors claim successful differentiation between live fingertips from fake fingertips made of the most commonly used materials in fingerprint spoofing. Another approach [23] combines image statistics and analysis of individual pore spacing with an efficiency level at 85%. Once again, we may guess these aliveness detection techniques strongly need fine tuning dedicated to such or such particular sensor. This methodology is somewhat close to detecting printing patterns in such a way that texture analysis will highlight granularity defaults inherent to the molding technique and materials used to build fake fingers.

Bubbles Detection

The idea here is to detect few non-living materials, often made by an inexperienced one, where air bubbles appear when drying in many materials such as glues, latex, and so on. An (little) experienced one will be able to quickly produce nice fakes. This straightforward idea will however also reject real fingertip with diseases such as psoriasis or eczema, showing circular damages.

9.5.2 Dynamic

Perspiration

The Biomedical Signal Analysis Laboratory at West Virginia University, USA, is developing a aliveness detection algorithm which is based on the detection of perspiration in a time progression of fingerprint images [86]. The original idea is based on the specific property of capacitive sensors to provide higher contrast with moist fingers (higher RDP): the user is asked to touch the sensor for several seconds, hence the natural perspiration process from sweat pores will constantly highlight ridges while perspiration will diffuse along them. Moreover, the system

Figure 9.6: Bad fingerprint copies

may check that the phenomenon really starts for the pores and analyse each inter-pores space to darken from the sides to the middle. The authors highlight the fact that the perspiration process doesn't occur in cadaver fingers. However, going out of the laboratory, this technique will face many real-life limitations: lack of perspiration in dry and cold environments, seasoning effect, perspiration disorders and abnormal skin conditions are not so rare. Moreover, the attack technique described in subsection 9.4.1 will result in capturing dry to wet images because of changing Relative Dielectric Permittivity, hence mimicking the perspiration process.

This is a long and inconvenient process, some patent descriptions propose to lightly heat the contact surface of the sensor to accelerate the perspiration process (for the reader information, lightly heating the fingertip is also proposed to dilate pores to ease their detection).

Skin Elasticity

This aliveness detection technique relies on fingertip deformation when pressing onto the contact surface of the sensor, or requires the user to move his finger while keeping a good contact with the surface [2, 64]. Using a thin (2D) fakes will not reproduce such non-linear distorsions of the fingertip, however a very thin layer of gelatin correctly glued onto a real finger would reproduce such distorsions, the same with our 3D glycerin fake finger technique having a smoothness close to a real finger.

Skin Bleaching

This new approach deals with detecting fake finger based on the property of color change exhibited by a real live finger when the finger touches a hard surface [134]. The force exhibited when the finger presses the hard surface changes the blood perfusion which results in a whiter color appearance compared to a normal uncompressed region. However we may guess a very thin, moreover transparent, gelatin layer glued onto a real finger will perfectly show real finger bleaching to the sensor.

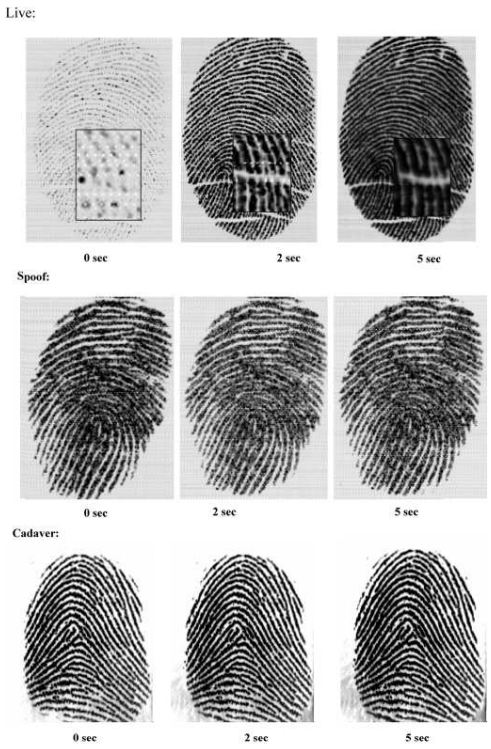


Figure 9.7: Perspiration effect after 2 seconds and 5 seconds on capacitive sensors

9.6 Our Contribution

In sections 9.4 and 9.5 we described known aliveness detection techniques and gave for each one either a solution to bypass such a detection based on our tests (when we had access to the technology) or just an idea on how to bypass (when we didn't have access to the technology), or highlighted the very limited scope of the technique (e.g. linear traces in fingerprint images). Most of the described techniques are even not commercial ones but only come from scientific publications or patents descriptions, hence we may only suggest straightforward ways to bypass these systems only based on our level of expertise.

9.7 Conclusion

Any aliveness detection system suffers from too many drawbacks in comparison to its advantages: additional hardware is costly, acquisition takes more time and is less convenient, the needed broad functioning range eases attacks, intrinsic rising of both FTE and FRR, and technical difficulties at the implementation stage. This clearly shows the only dissuasive effect of claiming vitality check embedded in the imaging system. We are here just at the beginning, ongoing efforts both at academic and industrial levels prove the progression margin of this technology, a first straightforward idea being to combine at least one hardware-based and one software-based countermeasures. For the reader information, the ASFIP project is currently working on a brand new and promising aliveness detection technique we can't disclose here.

CHAPTER 10
Biometric Systems Certification Issues

Contents

10.1 Introduction

Remembering figure 4.1 (flaws in biometric systems), we intend here to be able to exhaustively evaluate each attack path. We have at disposal several attack methodologies:

- Modify environmental conditions
  - Close to side-channels attacks in smart cards, we may cheat with the sensor and/or the processor running algorithms and decision to falsely gain access to the system. Temperature, humidity, light, current, voltage, clock, electromagnetic perturbation may be modified at will
- Direct attacks
  - The most straightforward idea to gain access to a system is to impersonate an authorized user either by having a fake copy of his registered fingerprint or digitally replaying his reference template or any matching candidate template

- Brute force attacks and/or iterative approximations
  - Use techniques described in chapter 8 either at sensor stage, extractor stage or matcher stage
- Forging databases and/or softwares
  - Ability to overwrite one particular template (with the hacker's one) in a reference database to gain access to its associated privileges. Ability to overwrite matching/decision algorithm with a "yes-match" one, ability to overwrite extractor with one delivering the desired template, whatever is the entry image

Apart from above described *practical* security, the evaluation is also intended to assess *theoretical* security (e.g. check FAR/FRR/FTE levels claimed by manufacturers).

10.2 Existing Initiatives & Standards

A UK initiative from Communications-Electronics Security Group (CESG), known as Best Practices in Testing and Reporting Performance of Biometric Devices built some basis in this field [75]. Common Criteria also edited a Biometric Evaluation Methodology (BEM) supplement in 2002 [6]. These seminal works, and others, led to international standardization initiative from ISO SubCommittee 37 "Biometrics" Working Group 5 "Biometric performance testing and reporting", i.e. ISO/IEC 19795-x :

- 19795-1: Principles and framework
- 19795-2: Testing methodologies for technology and scenario evaluation
- 19795-3: Modality-specific testing
- 19795-4: Interoperability performance testing

10.3 Our Approach & Contribution

Coming from the world of smart cards, with already certification schemes in place, our pragmatic approach relies on mature methodologies developed for Common Criteria for Information Technology Security Evaluation (ISO 15408), a worldwide standard for recognized certified IT secure products. Lessons learned for years in smart cards security level evaluation pave the way to other IT security systems certification. We are using here our work and results in both VulnBio and Asfip funded projects to propose (at ISO level) a framework and methodology for the certification of fingerprint recognition systems.

10.4 Rating Criteriae

Our proposal is inspired by the current rating grid for smart card's security evaluation. This is splitting in two different phases of an attack: *preparation* of the attack, then *realisation* of the attack, since the answer for a specific criteria differs most of the time. We target here

all possible attack paths: hardware, software, with/without specific countermeasures and so on. Preparation phase stands for lab activities and discrete, non-invasive study of the targeted system in real world conditions. Realisation phase stands for actively attacking the system in real world conditions, applying techniques learned at preparation phase.

10.4.1 Time elapsed

During the preparation phase, the reported needed time will depend on the time for an attacker to find out a vulnerability in the targeted system and the time to build the different material, software, tools to realize the attack. During the realisation phase this will just depend on the needed time to apply the previously identified technique.

Time elapsed	Preparation	Realisation
< Day	0	2
< Week	1	4
< Month	3	6
< 3 Months	5	8
< 6 Months	7	10
> 6 Months	10	n/a

Table 10.1: Rating criteria #1 - Time elapsed

10.4.2 Expertize needed

While needing a certain level of expertize to find out a vulnerability and fully develop an attack technique (preparation), applying this technique in real world conditions could be done by any newcomers (realisation).

- Novice: totally newcomer to the targeted technology
- Competent: having basic knowledge about the targeted technology
- Expert: high level of knowledge about the system and its interaction with the environment
- Multiple Experts: needed high level of knowledge in different technologies (e.g. multi-modal biometrics, optics & electronics)

Expertize needed	Preparation	Realisation
Novice	0	0
Competent	2	2
Expert	5	4
Multiple Experts	7	6

Table 10.2: Rating criteria #2 - Expertize needed

10.4.3 Knowledge of the target of evaluation(TOE)

This criteria relies on the ability to obtain technical information, public or not so public, about the system. This handles the necessary knowledge to find out a vulnerability, define and then realize an attack scenario.

- Public: publicly available data (e.g. internet)
- Restricted: Information available under NDA (Non-Disclosure Agreement)
- Sensible: data only available in the corporate development department
- Confidential: only very few persons in the company know the information

TOE Knowledge	Preparation	Realisation
Public	0	0
Restricted	2	1
Sensible	4	3
Confidential	6	5

Table 10.3: Rating criteria #3 - Knowledge of the target of evaluation

10.4.4 Window of opportunity

This particular criteria has several issues. The targeted system may not be available at will, depending on opening hours, staff presence limiting visible attacks, and so on. This window of opportunity could be consider either with time in a row, or summing multiples different access. The system may be accessible either only locally or remotely, and it would be easier if multiple instances of the same system are deployed at different locations.

- Unlimited: no risk for the attacker to be detected and stopped
- Easy: the attacker will be detected and stopped within a month
- Moderate: the attacker will be detected and stopped within a week
- Difficult: the attacker will be detected and stopped within a day

Time elapsed	Preparation	Realisation
Unlimited	0	0
Easy	1	4
Moderate	3	6
Difficult	5	8
Impossible	n/a	n/a

Table 10.4: Rating criteria #4 - Window of opportunity

10.4.5 Equipment needed

This criteria highlights the need of tools, either hardware or software, to find out a vulnerability, then prepare and realise the attack.

- None: not a single tool needed
- Standard: publicly available tool (e.g. a printer to print the fingerprint image)
- Specialized: professional tool (e.g. JetPac conductive printing tool available in our lab)
- Dedicated: need to develop a dedicated tool to bypass the targeted technology
- Multiple dedicated: need to develop at least two dedicated tools to bypass the targeted technology

Equipment needed	Preparation	Realisation
None	0	0
Standard	1	2
Specialized	3	4
Dedicated	5	6
Multiple dedicated	7	8

Table 10.5: Rating criteria #5 - Equipment needed

10.5 Final security levels

The sum of scores obtained in the different categories will give the overall score of the evaluation, then defining the obtained security level. However we would certainly need to improve our methodology since this does not take into account some paradoxical cases with linked criteriae (e.g. Time elapsed > Window of opportunity).

Obtained score	Certificate level
0-19	None
20-29	Level 2
30-35	Level 3
36-42	Level 4
> 42	Level 5

Table 10.6: Certification levels

10.6 Conclusion

Regarding all the realized work and obtained results in security related topics during previous chapters, we may advise minimum requirements for a biometric systems (however our very first target being fingerprint recognition systems) to obtain at least a level 2 certificate:

- Even if the area currently lacks of maturity, the system should at least implement an aliveness detection feature (all biometrics)
- Regarding Hill-Climbing attacks, the system should not disclose a matching score but only the binary information pass/fail (all biometrics)
  - Hill-Climbing attacks suppose the knowledge of the template format, this is often the case because of strong interoperability needs of biometric systems. A secret (proprietary) template format would be protected against these attacks, however security by obscurity too often proved being a bad solution (Kerckhoffs principle).
- Regarding image reconstruction around real minutiae, the system should also implement some pattern analysis approach (fingerprints dedicated)
  - However certain approaches, even purely minutiae-based, are intrinsically protected against the current state of the art reconstruction: the image reconstruction depicted in figure 8.7 has the default to produce several false minutiae points, this will completely change a local matching approach (minutia with its minutiae neighbourhood) such as the one we developed and will be described in the next part of this dissertation using Delaunay triangulation.
- Swipe sensors are a little bit more secure: no latent image, some swiping difficulties with several fakes (fingerprints dedicated)

This evaluation methodology assumes certifying for a certain security level of the targeted technology, rather independently from the context of use in the real world application: obtained results for the realisation phase are dedicated to the targeted attack scenario, the same system in another environment (e.g. non-staffed system instead of staffed) should not claim the same security level.

For sure, evaluation methodologies for such new technologies and applications need more time to reach the good level of maturity. However regarding market needs, we may assume certification is medium-term, if not short-term, topic.

PART III

Biometric Algorithms for Smart Cards

CHAPTER 11

Match-on-Card by Fuzzy Delaunay Triangulation

Contents

11.1 Introduction

As already discussed in the introduction of this dissertation, Match-on-Card defines a biometric system architecture where a smart card not only stores the reference template, but is also able to run the matching algorithm between the stored reference and a freshly captured and received candidate template. Beyond several initiatives with different biometric modalities, just few techniques give acceptable results and products on the market: fingerprints (using the minutiae approach), iris (simply an exclusive-or between two iris codes) [30] or vein pattern (banking MoC application in Japan, exclusive-or operation). The smart card component is very resource-limited and other biometrics are computationally demanding. The particular case of minutiae matching is one of the few approaches where the PC-based matching algorithm, being not only an exclusive-or, is suitable to smart cards with just few trades-off.

11.2 Problem Formulation

Our work presented here aims at proposing operations as simple as possible in order to compare two minutiae templates, the preferred target being a computing platform with only fixed-point operations, no floating-point. This constraint is particularly important in regards to usual rotations the matching algorithm has to handle. Usual operations to handle between two minutiae template are translation, rotation and scaling effect (homothety), these two last ones being conventionally computed using floating-point capability. These operations are mandatory to compute both a spacial distance and a direction distance between a reference minutia and a candidate minutia when mapping a candidate template onto a reference template (see figure 11.1).

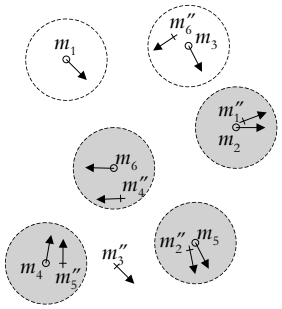


Figure 11.1: Matching Minutiae (in gray)

11.3 Our Approach

Our approach was to find out a different way to represent a minutiae set in order to be immune to translation, rotation and scaling effect. Moreover, this transformation and the comparison between two transformed templates must be easily computable. We decided to use a coding of information relatively to a triplet of minutiae in the same set. The obvious way to retrieve the same triplets in two equivalent sets is the Delaunay triangulation [14]. Most known algorithms to compute the Delaunay triangulation have a time complexity of $O(n\ log\ n)$ and space complexity of $O(n)$. We choose the ordering of the minutiae set with ascending y coordinates and

use the sweepline, from top to bottom, method to compute the Delaunay triangulation [42]. Our input set is the ISO Compact Card representation of a fingerprint minutiae set, using three bytes per minutia point $m\ \{x,\ y,\ \theta,\ t\}$, whereas our output set is composed of each Delaunay triangle, with its related information (area, circumcircle radius, vertices information) and optionally the barycentric coding of each other minutia in the set relatively to this triangle.

11.4 Previous Related Works

11.4.1 Isometries

Different approaches have been published that propose a geometric mapping function to be independant from isometries such as translation or rotation. The Point Pattern Matching (aka PPM) is a well known area in image processing and pattern recognition [29], similar tools are also heavily used in astronomy for constellation matching and automatic guiding of telescope upon stars mapping.

In [99], the authors use the notion of constellation matching by representing each minutia and its $k-closest$ neighbouring minutiae as a "constellation" using polar coordinates relatively to the reference minutiae. Each constellation is then insensitive to rotations and translations, and the matching algorithm aims at retrieving similar constellations between the reference and the candidate. However this approach only brings local structure comparison and no global structure comparison. We also notice this approach is increasing the space complexity by a factor in the order of $3k$ since it retains three information bytes per neighbouring minutia: radial coordinates, angular coordinates (for spatial distance) and $\delta\theta$ (for direction difference). A global structure comparison may be obtained by choosing $k = n$, the total number of minutiae in the set, but the space complexity will be $O(n^2)$ hence not very efficient, in general k is about 10. The time complexity of the mapping of the minutiae set onto the constellations set is $O(nk)$.

In others approaches [45, 12, 24], Geometric Hashing techniques are used to represent invariant local structures in order to ease the matching.

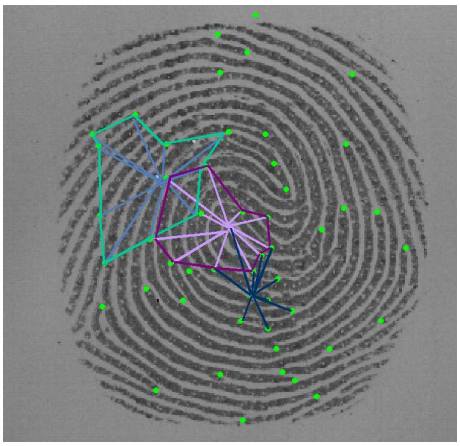


Figure 11.2: reference minutia and its nearest k-neighbors minutiae

11.4.2 Delaunay Triangulation in Fingerprint Recognition

Geometry of the triangle being invariant under translations and rotations, triangulation of a minutiae set is a preferred approach in fingerprint recognition. Classical minutiae triplets methods such as Germain *et al* [45] and Bhanu *et al* [12] are considering all the possible triangles within the minutiae set, thus a complexity of $O(n^3)$:

$$\binom{n}{3} = \frac{n!}{(n-3)!3!}$$

The FLASH algorithm described in [45] stores nine information per triangle (length of each side, ridge count between each vertices, and angle differences). For instance, an average template with 30 minutiae of about 90 bytes will be mapped onto a template with 4060 triangles of about 36540 bytes! Whereas being suitable to a PC implementation, this time and space complexity is definitely not suitable to smart card resources. As stated in [12], the FLASH algorithm is not optimized since information like lengths are sensitive to scaling effect and non-linear distorsions, and ridge counts are sensitive to image quality.

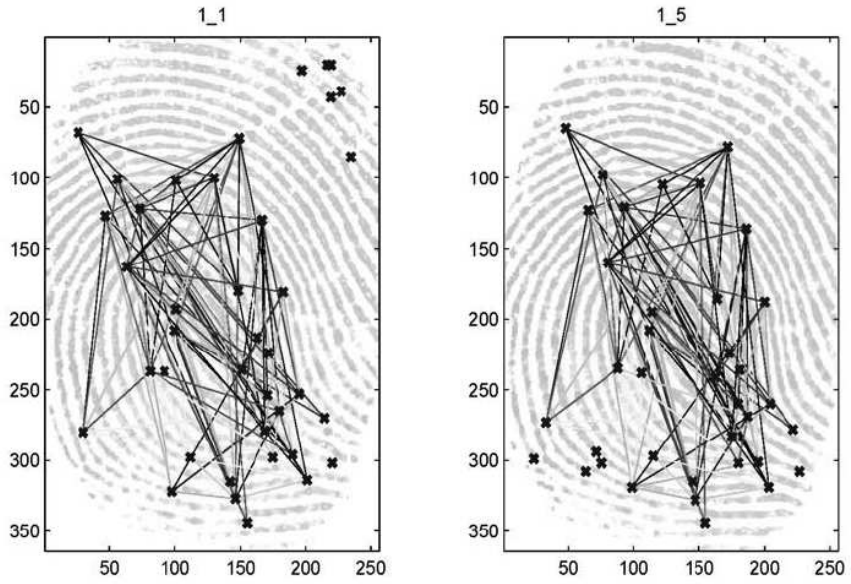


Figure 11.3: Matching Triangles between two instances of the same fingerprint

In order to reduce this complexity, a straightforward idea to select specific triangles in the minutiae set is the Delaunay Triangulation which results in $O(n)$ triangles. Using Delaunay triangulation for PC-Based fingerprint matching algorithm has been proposed in [11, 87].

Bebis *et al* [11] propose this approach and also handle invariance to scaling effect by using a 3-dimentional index based on two length ratios to the largest side and the angle opposed to the largest side. Beyond being of low space complexity (a 30 minutiae template of about 90 bytes will be mapped onto about 60 triangles of about 180 bytes) and invariant to scaling effect,

this approach is too simple since using only x and y location of each minutiae and no θ. This naturally leads to poor results given in their paper: about 0.13 FRR @ 0 FAR even with a set of three candidates per finger. Our reference to 0% FAR during this chapter is relevant due to the small size of public test databases, usually few hundreds of samples.

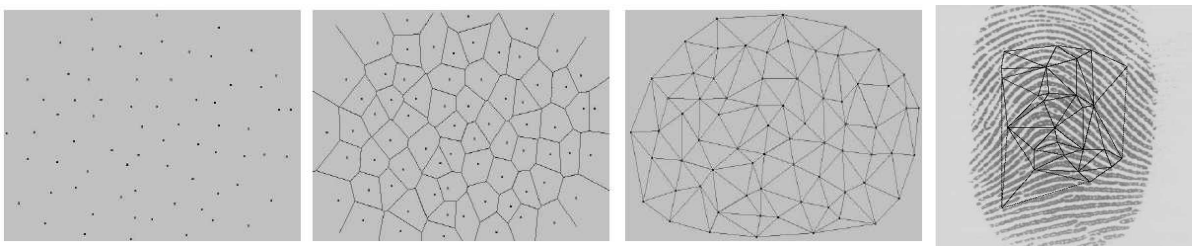


Figure 11.4: Points set, Voronoi diagram, Delaunay triangulation and its application to minutiae points set

A more efficient approach is described by Parziale *et al* in [87] and uses the θ information of each minutia in addition to the Delaunay Triangulation. In opposition to other methods using minutiae triplets, this method is using minutiae pairs (i.e each segment of the triangulation) and set up a 4-dimensional index per segment: L, $\delta\theta$, β_1, β_2 (see figure 11.5). This results in $O(n)$ segments and $O(4n)$ index template. Results disclosed in the paper stated about 0.04 FRR @ 0 FAR with only one candidate per reference, far better than the previously seen method with the same complexity.

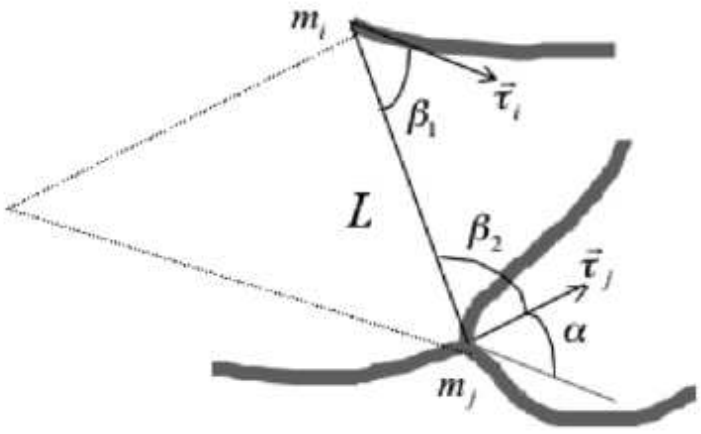


Figure 11.5: Parziale *et al* segment indexing

We may notice this method is not invariant to scaling effect by the use of L, the Euclidian distance between the two minutiae of the pair, and is only considering local structure matching of each segment with its neighbourhood.

11.5 Our Triangulation

11.5.1 Introduction

Above-mentioned results are obtained on PC-based algorithms with full accuracy of input data (x, y, θ) and operations using the "float" type. The challenge of our approach is to deal with smart card limitation of fixed-point operations on integers and the related loss of accuracy after each operation. Moreover our input set is limited by the ISO 19794-2 compact card format: one byte for x, one byte for y and one byte for minutiae type and θ, this last one being splitted in the two most significant bits for the minutiae type and the six least significant bits for θ. The limitation of θ, being a really discriminent data as seen in the last section, is the main bottleneck: only 64 possible values, hence a granularity of only 5.6 degrees per angle.

The Delaunay triangulation has the interesting properties of being unique for a set a point, hence similar sets of points will have similar Delaunay triangulation. The fundamental property of a Delaunay triangle is the absence of any other point of the set in its circumcircle apart from the three vertices of the triangle (see figure 11.6).

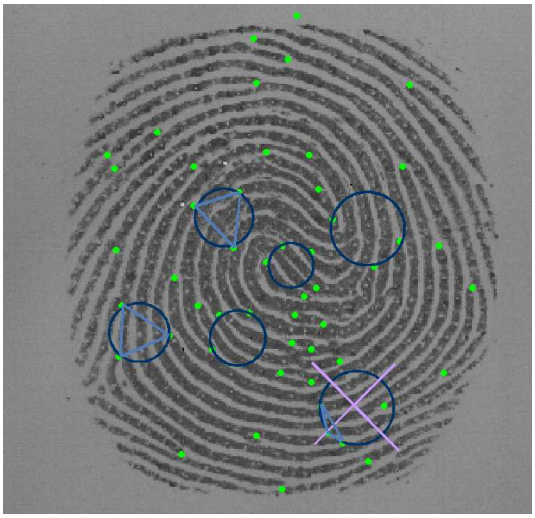


Figure 11.6: Delaunay triangles within a minutiae point set

Based on Parziale *et al* results, we studied the complexity of the Delaunay Triangulation for a smart card chip with the intuition that this only needs spatial distance computation, hence just a lot of simple fixed-point operations with integers such as additions and multiplications, and just few divisions.

After having computed and identified all triangles, we will consider each one as a reference and build up our brand new n-dimensional index by the 1-byte coding of information such as triangle area, radius of its circumcircle, occupation rate of the triangle within its circumcircle, direction differences between minutiae vertices, barycentric coordinates of each other minutia point in the set relatively to the reference triangle.

11.5.2 FDT : Fuzzy Delaunay Triangulation

Once the Delaunay Triangulation computed, we sort triangles in order to eliminate known problematic structures such as triangles with a very small -or very large- side, too small triangles and triangles with one very large inner angle. This mainly concerns triangles with two vertices belonging to the convex hull of the point set. Green-highlighted triangles on figure 11.8 are few examples of garbaged structures.

We also retain few *non-Delaunay* triangles when the tested point is very close to the circle, this helps in finding a match even under small non-linear distorsions of the fingerprint: we define our approach as the "Fuzzy Delaunay Triangulation". For the time being, the closeness of the tested point is decided upon a higher bound, empirically defined by our practical tests on fingerprint databases. Beyond non-linear distorsions, this is useful for interoperability to handle the divergence of templates generated, from a same inmage, by different extraction algorithms, especially on the positioning of minutiae (see figure 11.7). An example is depicted in figure 11.8 where the minutia point D is few pixels close to the circumcircle of the triangle (ABC), hence we also retain the green triangle.

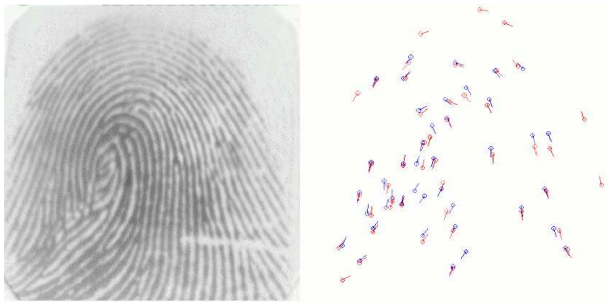


Figure 11.7: Minutiae positioning issue with different extractors

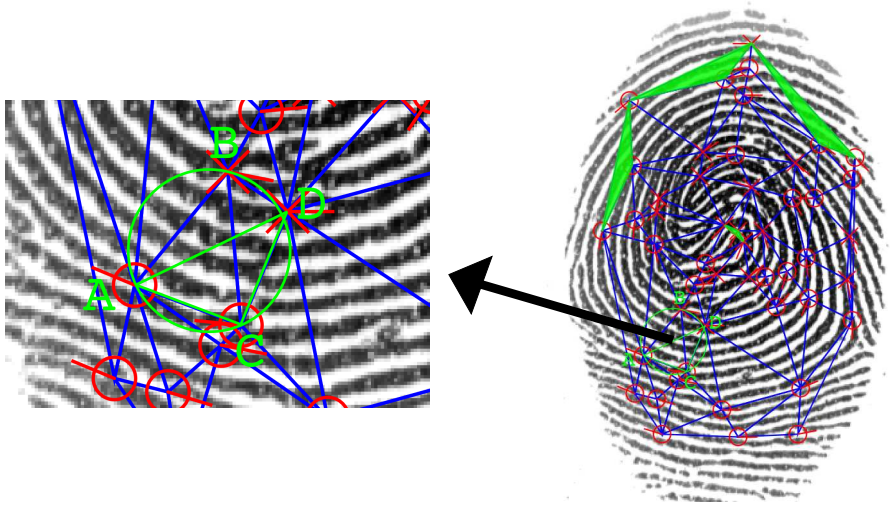


Figure 11.8: FDT: Fuzzy Delaunay Triangulation

11.5.3 Inputs of Triangulation

As already explained, we focused on the standardized ISO 19794-2 minutiae template, compact format (e.g. 3 bytes per minutia). Here are our notations:

Reference template:

$$T^R = \{M_1^R, M_2^R, M_3^R, \ \ldots\ , M_{n^R}^R\}$$

Candidate template:

$$T^C = \{M_1^C, M_2^C, M_3^C, \ \ldots\ , M_{m^C}^C\}$$

Each minutia point in Reference:

$$M_i^R = \{x_i^R, y_i^R, \theta_i^R, t_i^R\}$$

Each minutia point in Candidate:

$$M_j^C = \{x_j^C, y_j^C, \theta_j^C, t_j^C\}$$

x & y are 8-bit values, θ is 6-bit value (0 to 64) and t is the minutia type.

11.5.4 Triangulation

We used a sweepline method from top to bottom on the ordered (ascending y, hence from minutiae at the top of the fingerprint image to minutiae at the bottom of the fingerprint image) minutiae set for the triangulation.
For each ordered (A at the top, C at the bottom) triplet of points $\{A(x_a, y_b), B(x_b, y_b), C(x_c, y_c)\}$, we compute the origin $O(x_o, y_o)$ and the radius r (actually $\rho = r^2$ the square of the radius to avoid one square root operation) of its circumcircle. We then compare the square of the distance d^2 of any other minutia point $G(x_g, y_g)$ to O while $d^2 < \rho + \delta\rho$.

Circumcircle computation

Prerequisite test: A,B,C are not aligned

$$d_{ABC} = x_a(y_c - y_b) + x_b(y_a - y_c) + x_c(y_b - y_a) \neq 0$$

This is equivalent to check if the area A_T of the triangle (ABC) is not null:

$$A_T = \frac{1}{2} \mid d_{ABC} \mid$$

Computationally speaking, this test gives the area for free. This information will be useful later to compute the occupation ratio of the triangle within its circumcircle. This information could be an interesting entry in our index since this is invariant under all three usual transformations (translation, rotation and scaling).

Then, if previous test is true, solving the following system of circle equations:

$$\begin{aligned}(x_a - x_o)^2 + (y_a - y_o)^2 &= r^2 \ (= \rho)\\ (x_b - x_o)^2 + (y_b - y_o)^2 &= r^2 \ (= \rho)\\ (x_c - x_o)^2 + (y_c - y_o)^2 &= r^2 \ (= \rho)\end{aligned}$$

leads to the following relations:

Computing the square of the radius of the circumcircle of the triangle ABC:

$$\rho = \frac{((x_a - x_b)^2 + (y_a - y_b)^2) * ((x_a - x_c)^2 + (y_a - y_c)^2) * ((x_b - x_c)^2 + (y_b - y_c)^2)}{4d_{ABC}^2}$$

Computing the center of the circumcircle of the triangle ABC:

$$x_o = \frac{x_b^2 y_a - x_c^2 y_a - x_a^2 y_b + x_c^2 y_b - y_a^2 y_b + y_a y_b^2 + x_a^2 y_c - x_b^2 y_c + y_a^2 y_c - y_b^2 y_c - y_a y_c^2 + y_b y_c^2}{2d_{ABC}}$$

$$y_o = \frac{x_a^2 x_b - x_a x_b^2 - x_a^2 x_c + x_b^2 x_c + x_a x_c^2 - x_b x_c^2 + x_b y_a^2 - x_c y_a^2 - x_a y_b^2 + x_c y_b^2 + x_a y_c^2 - x_b y_c^2}{2d_{ABC}}$$

Please note the common denominator d_{ABC} has already been computed and non-null tested during the prerequisite test. Then the area A_C of the circumcircle is only costing one more multiplication:

$$A_C = \pi r^2 = \pi\rho$$

Finally we may compute the occupation ratio Occ_r of the triangle within its circumcircle:

$$Occ_r = \frac{A_T}{A_C} = \frac{|\, d_{ABC} \,|}{2\pi\rho}$$

Is FDT? test

"Is FDT?" stands for the procedure that decide to retain or not the tested triangle as an element of our Fuzzy Delaunay Triangulation index. In order to eliminate too small triangles (hence small ρ) and triangles with one too large internal angle (hence large ρ), bounds are given for ρ. For the time being, these bounds were empirically defined by our practical tests on fingerprint databases.

$$\rho_{MIN} < \rho < \rho_{MAX} \qquad (with\ \rho = radius^2)$$

If this test is true we then check for the Delaunay property: no other point G in the circumcircle of center O, hence $d(O,G) > \rho + \delta\rho$ (with $\delta\rho$ representing the tolerance factor for declaring the *fuzzy* Delaunay property).

Once the considered triangle is retained we then compute the *representative 7-dimensional index* Δ of the triangle ABC:

$$\Delta = \{\rho,\ Occ_r,\ \theta_1,\ \theta_2,\ Off_A,\ Off_B,\ Off_C\}$$

with

$$\theta_1 = (\theta_a - \theta_b)\ mod\ 64\ \ and\ \ \theta_2 = (\theta_a - \theta_c)\ mod\ 64$$

Off is the offset of the considered point in the ordered template. Keeping a trace of the three vertices is useful to quickly compute the neighbouring map of each triangle by intersection of two common vertices.

11.5.5 Barycentric Coordinates

Driven by our original idea of transformation invariance, the most natural way to represent a point G relatively to a reference triangle is the use of barycentric coordinates, i.e. the weights affected to $A, B\ and\ C$ such that G is the center of mass of the triangle. Invariance under translations, rotations and scaling effects is obtained, upon uniform transformations.

Actually, the use of barycentric coordinates brings the global structure matching in addition to the local structure matching of individual triangles properties and neighbourhood.

We then build a so-called *barycentric-enhanced constellation* C^b relatively to one FDT triangle:

$$C^b = \{\Delta,\ M_1^b,\ M_2^b,\ M_3^b,\ \ldots,\ M_{n-3}^b\}$$

where M_i^b is the barycentric index of the point G_i (detailled later).

Our final barycentric-enhanced constellations vector CV^b is composed of $n' = O(n)$ barycentric-enhanced constellations:

$$CV^b = \{C_1^b,\ C_2^b,\ C_3^b,\ \ldots, C_{n'}^b\}$$

Here are the detailled operations to compute the barycentric coordinates.
Prerequisite test: B,C,G are not aligned

$$d_{BCG} = x_g(y_c - y_b) + x_b(y_g - y_c) + x_c(y_b - y_g) \neq 0 \qquad (1)$$

Computing the barycentric coordinates and θ of a point G:
if (1) true,

$$w_a = 1 \ \ (choosen\ by\ convention)$$

$$w_b = \frac{x_g y_a - x_c y_a + x_a y_c - x_g y_c - x_a y_g + x_c y_g}{d_{BCG}}$$

$$w_c = \frac{x_b y_a - x_g y_a - x_a y_b + x_g y_b + x_a y_g - x_b y_g}{d_{BCG}}$$

$$\theta^b = (\theta_a - \theta_g)\ mod\ 64$$

if (1) false,

$$w_a = 0 \ \ (G\ is\ on\ the\ BC\ line)$$

$$w_b = 1 \ \ (choosen\ by\ convention)$$

$$w_c = \frac{x_b - x_g}{x_g - x_c}$$

$$\theta^b = (\theta_a - \theta_g)\ mod\ 64$$

then

$$M^b = \{w_a,\ w_b,\ w_c,\ \theta^b\}$$

11.5.6 Outputs of Triangulation

In summary, the output of our preprocessing of an ISO19794-2 minutiae template will be as follows for n minutiae and n' FDT triangles:

FDT Template:

$$\Delta T = \begin{bmatrix} \rho_1 & Occ_{r_1} & \theta_{1_1} & \theta_{2_1} & Off_{A_1} & Off_{B_1} & Off_{C_1} \\ \rho_2 & Occ_{r_2} & \theta_{1_2} & \theta_{2_2} & Off_{A_2} & Off_{B_2} & Off_{C_2} \\ \rho_3 & Occ_{r_3} & \theta_{1_3} & \theta_{2_3} & Off_{A_3} & Off_{B_3} & Off_{C_3} \\ \dots & & & & & & \\ \dots & & & & & & \\ \dots & & & & & & \\ \dots & & & & & & \\ \rho_{n'} & Occ_{r_{n'}} & \theta_{1_{n'}} & \theta_{2_{n'}} & Off_{A_{n'}} & Off_{B_{n'}} & Off_{C_{n'}} \end{bmatrix}$$

As an example, a 60 minutiae template (about 180 bytes) will result in about 80 FDT triangles of seven information bytes each (hence 420 bytes). This is still suitable to a smart card.

(or optionally) Barycentric-enhanced constellations template:

$$CT^b = \begin{bmatrix} \rho_1 & Occ_{r_1} & \theta_{1_1} & \theta_{2_1} & Off_{A_1} & Off_{B_1} & Off_{C_1} & M^b_{1_1} & M^b_{2_1} & M^b_{3_1} & \dots & \dots & \dots & M^b_{n-3_1} \\ \rho_2 & Occ_{r_2} & \theta_{1_2} & \theta_{2_2} & Off_{A_2} & Off_{B_2} & Off_{C_2} & M^b_{1_2} & M^b_{2_2} & M^b_{3_2} & \dots & \dots & \dots & M^b_{n-3_2} \\ \rho_3 & Occ_{r_3} & \theta_{1_3} & \theta_{2_3} & Off_{A_3} & Off_{B_3} & Off_{C_3} & M^b_{1_3} & M^b_{2_3} & M^b_{3_3} & \dots & \dots & \dots & M^b_{n-3_3} \\ \dots & & & & & & & & & & & & & \\ \dots & & & & & & & & & & & & & \\ \dots & & & & & & & & & & & & & \\ \dots & & & & & & & & & & & & & \\ \rho_{n'} & Occ_{r_{n'}} & \theta_{1_{n'}} & \theta_{2_{n'}} & Off_{A_{n'}} & Off_{B_{n'}} & Off_{C_{n'}} & M^b_{1_{n'}} & M^b_{2_{n'}} & M^b_{3_{n'}} & \dots & \dots & \dots & M^b_{n-3_{n'}} \end{bmatrix}$$

with

$$M^b_{i_{1\to n}, j_{1\to n'}} = \{w_{a_{i,j}}, w_{b_{i,j}}, w_{c_{i,j}}, \theta^b_{i,j}\}$$

As an example, a 60 minutiae template (about 180 bytes) will result in about 80 FDT triangles of $7 + 57 * 4 = 235$ information bytes each (hence about 18 kB!). This is too costly, both in space and time complexity, for a smart card. In conclusion, the barycentric approach may sound useful for a PC-based solution but is out of the scope of our resources-limited approach.

11.5.7 Algorithm

Algorithm 1 Fuzzy Delaunay Triangulation preprocessing algorithm

Parameter: $\delta\rho$, ρ_{MIN}, ρ_{MAX}
Input: T^R or T^C
Output: ΔT^R or ΔT^C
Complexity: $O(n\ log\ n)$

```
ΔT ← {0}
for i = 1 to n − 2 (i.e. minutia point A) do
  for j = i + 1 to n − 1 (i.e. minutia point B) do
    for k = j + 1 to n (i.e. minutia point C) do
      compute ρ_{i,j,k}
      if (ρ_{i,j,k} ⩽ ρ_MIN) || (ρ_{i,j,k} ⩾ ρ_MAX) then
        break
      end if
      compute x_o and y_o
      for l = k + 1 to n (i.e. minutia point G) do
        if sd²(O,G) ⩽ ρ_{i,j,k} + δρ then
          break
        end if
        if l = n then
          append {ΔT, {ρ_{i,j,k}, Occ_{r_{i,j,k}}, θ_{i,j}, θ_{i,k}, i, j, k}}
        end if
      end for
    end for
  end for
end for
sort ΔT by decreasing ρ
```

11.6 Matching approach

11.6.1 Introduction

Our basis is the matching of individual triangles between the reference and the candidate upon their (sorted with decrasing ρ) representative quadruplet $\{\rho,\ Occ_r,\ \theta_1,\ \theta_2\}$. Once the list of matching triangles is build, we check the coherence of the list by searching adjacent matching triangles by using $\{Off_1,\ Off_2,\ Off_3\}$ information. One triangle may have up to three adjacent triangles, the score of each individual triangle is weighted by the number of adjacent matching triangles. We may notice that matching one triangle and its three neighbours is equivalent to match six minutiae.

Beyond this local structure matching, we compute the translation and rotation parameters for each matching triangles $\{\delta x,\ \delta y,\ \delta\theta\}$. Then we check for the maximum number of matching triangles under the same transformation parameters in order to verify a global structure coherence, especially between distant (i.e. non-adjacent) triangles. This is a low-cost alternative to the barycentric approach.

As an example, two distant quadruplets of triangles (i.e. quadruplet: one matching triangle with its three matching neighbours) with the same transformation parameters is equivalent to match twelve minutiae, hence sufficient to declare a positive match.

Computationally speaking, the matching is just a lot of byte comparisons hence perfectly suitable to any low-cost processor. Time complexity of the matching algorithm is negligible in comparison to the time complexity of the FDT preprocessing algorithm. We may notice here that the enrolment process is costing one FDT template computation (for the reference, done once and stored), whereas the matching process is costing one FDT template computation (for the candidate) and one FDT comparison matching. Hence we may say the complexities of enrolment and matching are equivalent.

11.6.2 Inputs

Reference FDT Template of a n-minutiae set:

$$\Delta T^R = \{\Delta_1^R,\ \Delta_2^R,\ \Delta_3^R,\ \ldots,\Delta_{n'}^R\}$$

with

$$\Delta_{i_{1\to n'}}^R = \{\rho_i^R,\ Occ_{r_i}^R,\ \theta_{1_i}^R,\ \theta_{2_i}^R,\ Off_{A_i}^R,\ Off_{B_i}^R,\ Off_{C_i}^R\}$$

Candidate FDT Template of a m-minutiae set:

$$\Delta T^C = \{\Delta_1^C,\ \Delta_2^C,\ \Delta_3^C,\ \ldots,\Delta_{m'}^C\}$$

with

$$\Delta_{j_{1\to m'}}^C = \{\rho_j^C,\ Occ_{r_j}^C,\ \theta_{1_j}^C,\ \theta_{2_j}^C,\ Off_{A_j}^C,\ Off_{B_j}^C,\ Off_{C_j}^C\}$$

11.6.3 Fine-Tuning Parameters

The main parameter is the "Threshold Score":

$$S_T = 0 \; to \; 256$$

Secondary parameters are useful to compensate usual non-linear distortions:
Bounding ρ :

$$\rho_j^C = \rho_i^R \pm \delta_\rho$$

Bounding Occ_r :

$$Occ_{r_j}^C = Occ_{r_i}^R \pm \delta_{Occ_r}$$

Bounding θ :

$$\theta_j^C = \theta_i^R \pm \delta_\theta \quad (both \; for \; \theta_1 \; and \; \theta_2)$$

11.6.4 Outputs & Decision

For each matching triangles (i.e. all four secondary parameters above are matching) we store $\{i,\ j,\ \delta x,\ \delta y,\ \delta\theta\}$ in an index table of k entries. k is the total number of matching triangles.
$\{\delta x,\ \delta y,\ \delta\theta\}$ are computed upon the top-most point A and A' of each triangle:

$$\begin{aligned} \delta x &= x_A - x_{A'} \\ \delta y &= y_A - y_{A'} \\ \delta\theta &= \theta_A - \theta_{A'} \end{aligned}$$

Index table:

$$Match\Delta = \begin{bmatrix} i_1 & j_1 & \delta x_1 & \delta y_1 & \delta\theta_1 \\ i_2 & j_2 & \delta x_2 & \delta y_2 & \delta\theta_2 \\ i_3 & j_3 & \delta x_3 & \delta y_3 & \delta\theta_3 \\ \dots & & & & \\ \dots & & & & \\ \dots & & & & \\ \dots & & & & \\ i_k & j_k & \delta x_k & \delta y_k & \delta\theta_k \end{bmatrix}$$

Once the table is constructed, a first pass will eliminate entries with inconsistent $\{\delta x,\ \delta y,\ \delta\theta\}$ (so-called ***RemoveFakeIndex*** function). This usually eliminates falsely matching triangles (i.e. retain the correct pair when a reference triangle have more than one image triangle in the candidate set). The updated index table is of k' entries, with $k' \leqslant k$.

Each Delaunay triangle is surrounded three other Delaunay triangles or less (e.g. in the case of triangles with two vertices on the convex hull of the point set). By *surrounded* we mean adjacent triangles connected by two common vertices.

We let $S = T + E_{adj}$, where T is the number of matching triangles and E_{adj} is the number of edges adjacent to two matching triangles. The final score of the matching algorithm is S.

11.6.5 Algorithm

Algorithm 2 Matching algorithm

Parameters: δ_ρ, δ_{Occ_r}, δ_θ, S_T
Inputs: ΔT^R and ΔT^C
Output: *Accept* or *Reject*
Complexity: $O(n'^2)$

```
Match△ ← {0}
S ← 0
for i = 1 to n' do
  for j = 1 to m' do
    if |ρ^R − ρ^C| ⩾ δ_ρ then
      break
    else if |Occ_r^R − Occ_r^C| ⩾ δ_Occr then
      break
    else if |θ_1^R − θ_1^C| ⩾ δ_θ then
      break
    else if |θ_2^R − θ_2^C| ⩾ δ_θ then
      break
    else
      append {Match△, {i, j, δx, δy, δθ}}
    end if
  end for
end for
sort {Match△, RemoveFakeIndex{δx, δy, δθ}}
for i = 1 to k − 1 do
  w ← 1
  for j = i + 1 to k do
    if AdjacentTriangles(△_i, △_j) = true then
      w++
    end if
  end for
  S = S + w
  if S ⩾ S_T then
    Accept and stop
  end if
end for
Reject and stop
```

11.7 Prototyping with GCC, Octave and GnuPlot

We built a demonstrator out of our fingerprint recognition approach. We targeted open-source tools such as Octave (a Mathlab-like software) [38], GCC (C compiler), ImageMagick [114] or GnuPlot.

The C library for ISO19794-2 minutiae extraction was given under license by one of our partner since image processing and extraction is out of our competencies (see www.ikendi.com).

The live fingerprint capture is using a Futronic optical sensor with provided C library under Linux.

Our triangulation and matching algorithms were firstly tested on Octave, then ported in C language with GCC for efficiency. However Octave remains our interface to handle C libraries, ImageMagick for fingerprint image processing & display and GnuPlot to display minutiae (red) and the obtained triangulation (blue) onto the fingerprint image. **O** denotes a ridge ending, **X** denotes a ridge bifurcation and / within the symbol displays the minutia angle.

Figure 11.9 gives an example of a positive fingerprint recognition.

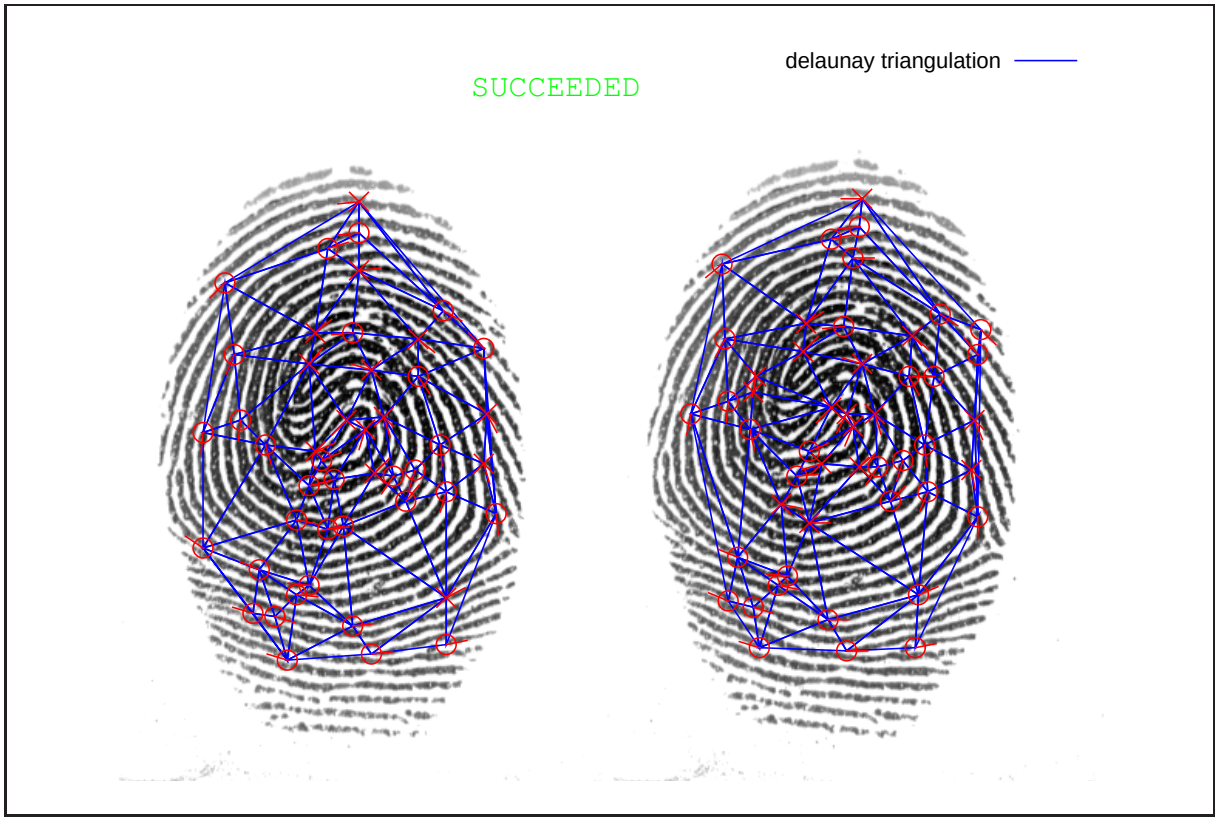


Figure 11.9: Livescan demo application

11.8 On-Card Prototyping

Our very first implementation in pure Javacard code ran in 0.4s, applet size was only about 500 bytes and RAM resources were limited to 256 bytes (only the APDU buffer of the smart card). But this preliminary version used only local structure information, accuracy loss were very high (8-bit registers) hence bounds were very tight to reach 0 FAR, and results were about 0.15 FRR @ 0 FAR. However it was promising in our opinion. Targeting performances and MINEX II evaluation, we decided to use a Javacard based on a 16-bit processor and developed native routines to accelerate the algorithm and complement it with global structure information. This resulted in a 5kB program, needing 1kB of RAM and running in half a second approximately.

11.9 Results & Conclusion

We tested our native implementation against different publicly available fingerprint databases of FVC contests available in [74]. We developed test scripts using Perl language with the same protocol as FVC to measure FAR and FRR. Our preliminary results gave about 0.05 FRR @ 0 FAR, equivalent to the *non-resources-limited* Parziale *et al* solution. This value has been confirmed with the MINEX II evaluation of MoC solutions conducted in 2009 by the US NIST (see next chapter).

CHAPTER 12
Evaluation of Match-on-Card Performances

Contents

12.1 Introduction to Minex & MinexII by NIST

The Minex II trial [47, 49] was firstly conducted in 2007 by United States' NIST (National Institute of Standards and Technology) to evaluate the accuracy and speed of Match-on-Card verification algorithms. These run on ISO-IEC 7816 smart cards. They compare conformant reference and verification instances of the ISO-IEC 19794-2 Compact Card fingerprint minutiae standard. This MinexII-PhaseI evaluation campaign follows the original 2004 Minex campaign [48], evaluating PC-based fingerprint verification algorithms using standardized ANSI 378 minutiae format. This original Minex aims to assess the interoperability of this ANSI 378 minutiae format. Minex II defines interoperability in terms of biometric matching error rates measured when a matching algorithm from company X compares two standard records produced by the fingerprint minutia extraction algorithms of companies Y and Z. An interoperable Match-On-Card implementation is one that gives low error rates irrespective of the producer of the input records. From a security point of view, MinexII did not evaluate cards or algorithms vulnerabilities: in 2007, NIST also conducted "Secure Biometric Match-On-Card", SBMOC [27]. This aimed at assessing the feasibility of adding MoC with cryptographic architecture to PIV cards (Personal Identity Verification). Both SBMOC and PIV will be fully described in part IV -Security Protocols for Smart Cards with Biometrics- of this dissertation. This has no links with MinexII. For MinexII-phaseII/III (Oct'08), our algorithm described in the previous chapter 11 appears under the name **Micro-PackS** and is implemented on Gemalto's smart cards.

12.2 MinexI

It has been generally acknowledged that the interchange of fingerprint image data provides the greatest interoperability between dissimilar fingerprint recognition systems. However, standards now exist that specify the location and formatting of processed minutiae locations data, or templates, for matching purposes. For many applications, minutiae templates offer a more space-efficient, less resource intensive, and more cost effective alternative to raw images. The Minutiae Interoperability Exchange Test was performed to determine the feasibility of using standard minutiae templates as the interchange medium for fingerprint information between different fingerprint matching systems. Here are learned lessons in MinexI:

- At fixed False Match Rate (FMR), the False Non Match Rate (FNMR) is divided by 2 to 5 (depending on the different implementations) with proprietary templates in comparison to standardized templates. This proves the superiority of proprietary templates, however this gap is not very significant and is even smaller when using a two-finger solution (and-rule fusion: both fingers must be authenticated). Unfortunately, no information is given about the average proprietary template size to be compared with standardized ANSI378 template size.

- Most applications using fingerprints store at least two fingers for security reasons (for backup in case of injury). This is the reason why the idea to evaluate the accuracy level of using two fingers instead of only one may be relevant. A quick look on results shows an order of magnitude of 10 regarding FNMR at fixed FMR: for instance at FMR 0.01 (1%) we obtain FNMR 0.01 (1%)@one-finger and FNMR 0.001(0.1%)@two-finger in best cases for the leading implementation. Average results give FNMR 0.1@FMR 0.01 for one-finger and FNMR 0.01@FMR 0.01 for two-finger.

- Options of the ANSI378 minutiae format are (1) cores and deltas location and (2) ridge count (i.e counting the number of ridges between two consecutive minutiae). These options give more information in a template, thus improving the accuracy. However, managing and storing such information is costly. Hence MinexI proposed to evaluate the efficiency of using the ridge count option in addition to the minutiae location. Results show an average template size of about 300 bytes for the ANSI378 format without the option, whereas showing an average template size of about 1300 bytes for the ANSI378 format with the option. Regarding accuracy, it shows the poor accuracy difference when using the option (less than a factor 2), thus justifying to save memory & processing resources by not using it.

We may notice here that one matcher among those tested in MinexI is presented as “Sagem Morpho Match-on-Card algorithm”, a PC-Based implementation of their MoC algorithm. This proves Sagem anticipated the need for Match-on-Card solution evaluation before Minex II.

12.3 MinexII

The Minex evaluation was intended to assess the viability of the INCITS 378 templates as the interchange medium for fingerprint data. The main objective was to determine whether standardized minutia reference templates can be subsequently matched against an authentication

template from another vendor. Minex II retains this objective but focuses the activity to a restricted class of matchers. Minex II is intended to measure the core algorithmic capabilities of fingerprint matching algorithms running on standardized ISO/IEC 7816 smart cards. The Minex II evaluation measures Match-on-Card performance at low false match rates with statistical robustness.

Evaluation protocol

We helped the NIST to define the evaluation plan [47]. This covers how to communicate with smart cards and how to handle biometric data within a smart card: we participated to the Application Programming Interface (API) definition (i.e. which APDU commands are mandatory to process such an evaluation). NIST is using its own very large (and private) database of fingerprint impressions to conduct the evaluation: thousands of intra-class comparisons (aka genuine match, i.e. comparing different impressions of the same finger) will give the FRR, whereas thousands of inter-class comparisons (aka impostor match, i.e. comparing impressions of different fingers) will give the FAR. The defined protocol proposes to use a different minutiae generator for enrollment and matching to reinforce interoperability testing. Figure 12.1 shows the complete cross-comparison table between all minutiae generators (first column) and all minutiae matchers (second row).

(a) FNMR at FMR = 0.01

Nfing	Column = MATCH-ON-CARD algorithm, Row = INCITS 378 generator																
2	MX2D	MX2E	MX2G	MX2H	MX2I	MX2J	MX2K	MX2M	MX2N	MX2O	MX2P	MX2Q	MX2R	MX2S	MX2T	MX2N†	MX2T†
A	0.0038	0.0055	0.0427	0.0324	0.0140	0.0146	0.0156	0.0399	0.0073	0.0041	0.0048	0.0047	0.0466	0.0523	0.0016	0.0127	0.0014
B	0.0020	0.0070	0.0168	0.0159	0.0080	0.0082	0.0086	0.0207	0.0066	0.0037	0.0041	0.0030	0.0356	0.0416	0.0031	0.0098	0.0053
C	0.0030	0.0112	0.0237	0.0231	0.0118	0.0119	0.0128	0.0270	0.0095	0.0056	0.0056	0.0035	0.0482	0.0561	0.0065	0.0101	0.0126
D	0.0015	0.0029	0.0105	0.0066	0.0046	0.0046	0.0048	0.0156	0.0038	0.0021	0.0025	0.0023	0.0175	0.0197	0.0013	0.0092	0.0030
E	0.0040	0.0055	0.0195	0.0143	0.0093	0.0096	0.0102	0.0514	0.0063	0.0041	0.0044	0.0040	0.0343	0.0387	0.0026	0.0078	0.0034
F	0.0030	0.0112	0.0234	0.0225	0.0115	0.0117	0.0125	0.0271	0.0094	0.0055	0.0056	0.0035	0.0481	0.0557	0.0066	0.0102	0.0126
G	0.0025	0.0080	0.0205	0.0152	0.0102	0.0105	0.0083	0.0310	0.0079	0.0042	0.0044	0.0031	0.0439	0.0508	0.0041	0.0099	0.0069
N	0.0049	0.0121	0.0224	0.0168	0.0116	0.0118	0.0123	0.0343	0.0114	0.0069	0.0059	0.0050	0.0499	0.0561	0.0063	0.0146	0.0094
1C	0.0030	0.0054	0.0254	0.0194	0.0087	0.0088	0.0092	0.0425	0.0049	0.0031	0.0040	0.0036	0.0314	0.0359	0.0022	0.0044	0.0036
1D	0.0052	0.0201	0.0315	0.0267	0.0177	0.0181	0.0190	0.0354	0.0147	0.0103	0.0102	0.0055	0.0611	0.0688	0.0128	0.0154	0.0211
1F	0.0046	0.0140	0.0315	0.0248	0.0158	0.0162	0.0170	0.0392	0.0126	0.0075	0.0077	0.0052	0.0548	0.0618	0.0075	0.0166	0.0096
1G	0.0046	0.0140	0.0315	0.0249	0.0158	0.0162	0.0170	0.0393	0.0126	0.0075	0.0077	0.0052	0.0548	0.0619	0.0075	0.0166	0.0096
1J	0.0032	0.0132	0.0272	0.0210	0.0133	0.0133	0.0140	0.0270	0.0087	0.0048	0.0075	0.0043	0.0484	0.0633	0.0086	0.0114	0.0188
1L	0.0023	0.0058	0.0201	0.0146	0.0093	0.0096	0.0101	0.0295	0.0057	0.0031	0.0040	0.0033	0.0355	0.0417	0.0023	0.0084	0.0034
1M	0.0022	0.0056	0.0197	0.0151	0.0071	0.0071	0.0074	0.0268	0.0064	0.0039	0.0038	0.0030	0.0474	0.0538	0.0031	0.0095	0.0062
1N	0.0038	0.0123	0.0290	0.0228	0.0134	0.0136	0.0143	0.0312	0.0077	0.0049	0.0064	0.0044	0.0485	0.0557	0.0062	0.0074	0.0114
1T	0.0022	0.0039	0.0174	0.0130	0.0078	0.0079	0.0060	0.0306	0.0048	0.0029	0.0030	0.0027	0.0246	0.0275	0.0019	0.0052	0.0036
1Y	0.0020	0.0046	0.0166	0.0124	0.0079	0.0080	0.0061	0.0274	0.0055	0.0033	0.0034	0.0028	0.0426	0.0482	0.0025	0.0088	0.0049
2A	0.0030	0.0112	0.0237	0.0231	0.0118	0.0119	0.0128	0.0270	0.0095	0.0056	0.0056	0.0035	0.0482	0.0561	0.0065	0.0101	0.0126

A = Cogent	E = Neurotechnology	1C = L1/Identix	1J = BIO-Key	1T = Neurotechnology
B = Dermalog	F = Innovatrics	1D = Precise Biometrics	1L = Motorola	1Y = Aware
C = Bioscrypt	G = NEC	1F = XTec	1M = Aware	2A = ImageWare
D = Sagem	N = Crossmatch	1G = Secugen	1N = Sonda	(ANSI 378 generators)

MX2D = Sagem Morpho	MX2I = Oberthur / ID3	MX2N = Gemalto / Innovatrics	MX2R = Gemalto / Micro-PackS
MX2E = Sagem Morpho	MX2J = Oberthur / ID3	MX2O = Gemalto / Innovatrics	MX2S = Gemalto / Micro-PackS
MX2G = Oberthur / ID3	MX2K = Oberthur / ID3	MX2P = Oberthur / ID3	MX2T = Gemalto / Cogent
MX2H = Oberthur / ID3	MX2M = Giesecke&Devrient	MX2Q = Oberthur / ID3	xxxx = (smart card / algorithm)

Figure 12.1: Minex II cross-comparisons

Accuracy & Speed results

Figure 12.2 gives the accuracy of MoC with two-finger implementation. Solid lines are corresponding to results obtained when using Cogent generator at enrollment and Sagem generator at matching. Dashed lines are corresponding to results obtained when using Innovatrics generator at enrollment and Sagem generator at matching. This shows the solid yellow line as the leading solution (Cogent on Gemalto smartcard). This is a little biased since this is also using Cogent as the reference generator. Yellow dashed line is more relevant. The leading implementations are Cogent (yellow), Sagem (red), ID3 (green) and Innovatrics (black). The very first implementation of our algorithm is closing the evaluation with 0.05 FNMR @ 0.01 FMR (i.e. 5% of FRR -false reject- for 1% of FAR -false acceptance-) but enters the ring of professional solutions.

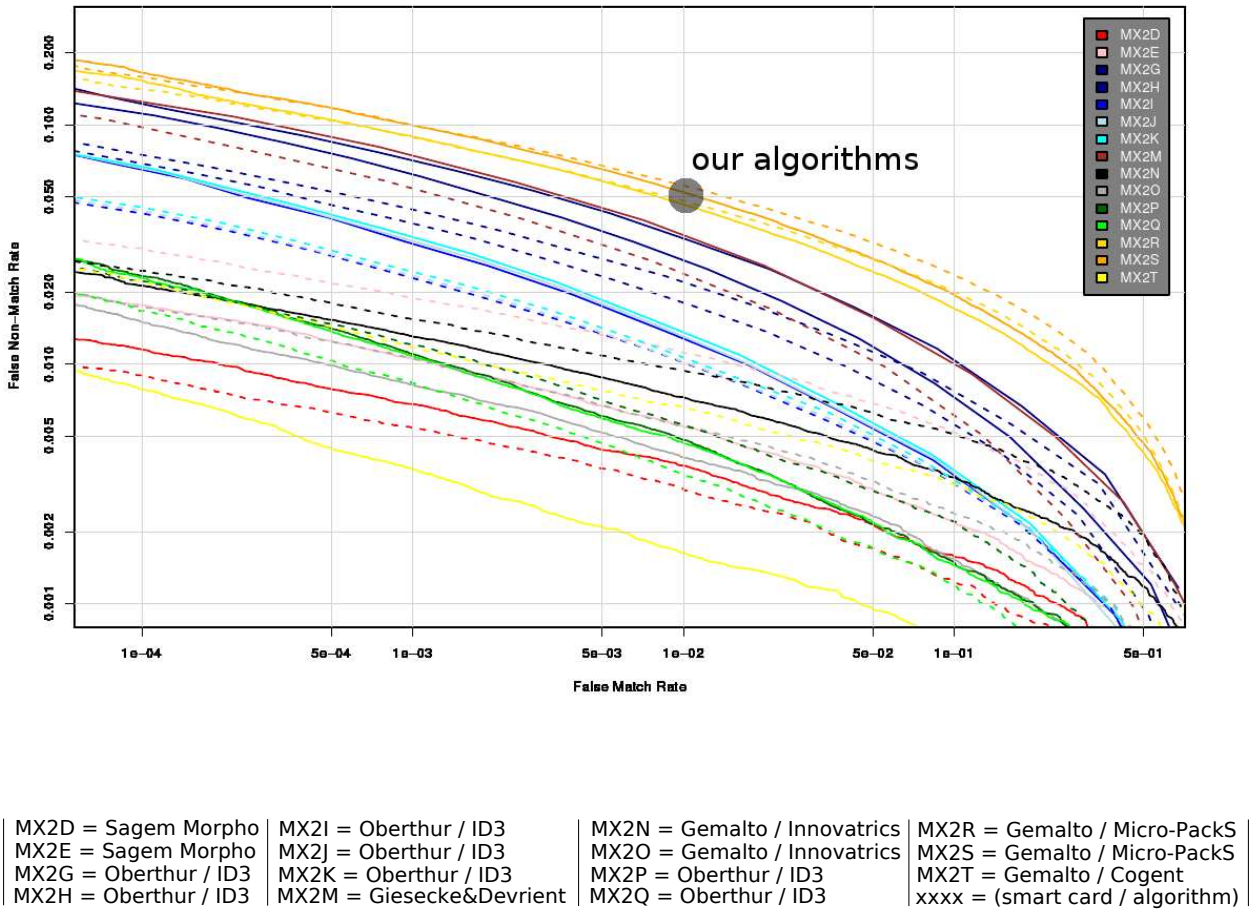
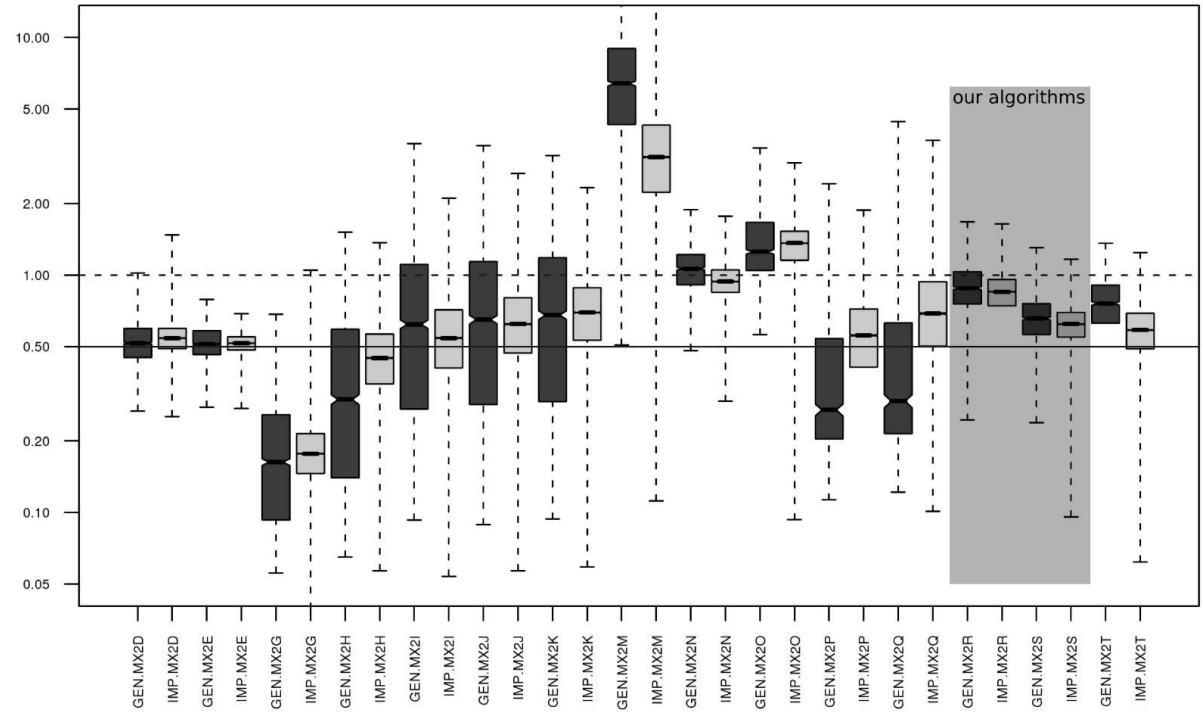


Figure 12.2: Minex II accuracy results (how to read: the lower it is, the better it is)

Figure 12.3 shows timing results of MoC solutions. The average response time is between 0.5 second and 1 second. Our best solution gives 0.7 second. Light boxes report timings for genuine matches, whereas dark boxes report timings for impostor matches.



MX2D = Sagem Morpho	MX2I = Oberthur / ID3	MX2N = Gemalto / Innovatrics	MX2R = Gemalto / Micro-PackS
MX2E = Sagem Morpho	MX2J = Oberthur / ID3	MX2O = Gemalto / Innovatrics	MX2S = Gemalto / Micro-PackS
MX2G = Oberthur / ID3	MX2K = Oberthur / ID3	MX2P = Oberthur / ID3	MX2T = Gemalto / Cogent
MX2H = Oberthur / ID3	MX2M = Giesecke&Devrient	MX2Q = Oberthur / ID3	xxxx = (smart card / algorithm)

Figure 12.3: Minex II timing results

12.4 MinexI vs MinexII

Minutiae format

Minex I uses ANSI 378 minutiae format (6 bytes per minutia) at both generation and matching, whereas Minex II is using ANSI 378 at generation and convert from ANSI 378 to ISO19794-2 compact card format (3 bytes per minutiae) to be sent to the smart card for matching.

Accuracy

About two years separate MinexI report and MinexII-PhaseII report, and three years with MinexII-PhaseIII report. This maybe justifies the small between performances reported for both PC-Based and Match-on-Card solutions. We also may notice the loss of information using three bytes per minutiae instead of six bytes. In addition to very limited resources, this very small minutiae format is another handicap regarding performances. However, the comparison in the report clearly shows that accuracy with compact card format makes MoC relevant. Data below shows the comparison between PC-based and MoC for best-of-class vendors participating to both Minex I & II:

- Cogent with its own generator
  - 0.0011 FNMR @ 0.01 FMR for PC-based
  - 0.0014 FNMR @ 0.01 FMR for MoC
- Sagem with its own generator
  - 0.0012 FNMR @ 0.01 FMR for PC-based
  - 0.0015 FNMR @ 0.01 FMR for MoC

This clearly shows the relevance of smart cards to compute the fingerprint comparison.

Speed

For MinexI, matching speed is not an issue. Whereas for MinexII, the matching being processed by the smart card chip, hence with very limited resources, matching speed is a major concern. For user acceptance, however depending on the application, we usually target a matching result within a second (1s). We have seen previously that the average Match-on-Card timing is of about half a second, hence MoC is clearly acceptable. For information, average matching time with PC-Based solutions reported by Minex I is of about few milliseconds and ANSI 378 generators needs few hundreds of millisecond.

12.5 Results with our MoC Algorithm

To summarize previously seen results with our MoC algorithm:

- Matching time is of about 700 milliseconds (enters in specification). Our platform is a 16-bit RISC processor cadenced at 15MHz.
- At fixed 1% False Acceptance Rate, the False Rejection Rate is at 1.75% when using the best minutiae extractor (still out of specification, i.e. FRR 1%)
- Needed RAM resources is of about 1kB for processing
- Needed EEPROM resources is of about 5kB for both algorithm code and data storage

Figure 12.4 is a comparison, on the same scale, between MinexII results and FVC2006 [44] light category results (in light background). Mainly academics and small companies compete at Fingerprint Verification Competition (FVC). The light category, targeting embedded electronics, provides more resources (4MB memory) than available in a smartcard, and may use proprietary templates up to 2kB. With less resources and a more compressed template, our algorithm and others prove the advancement of big players about Match-on-Card. At FMR 1%, FVC results show FNMR from 8% to 20%, whereas MinexII results show FNMR from 0.2% (Cogent) to 5% (Our algorithm).

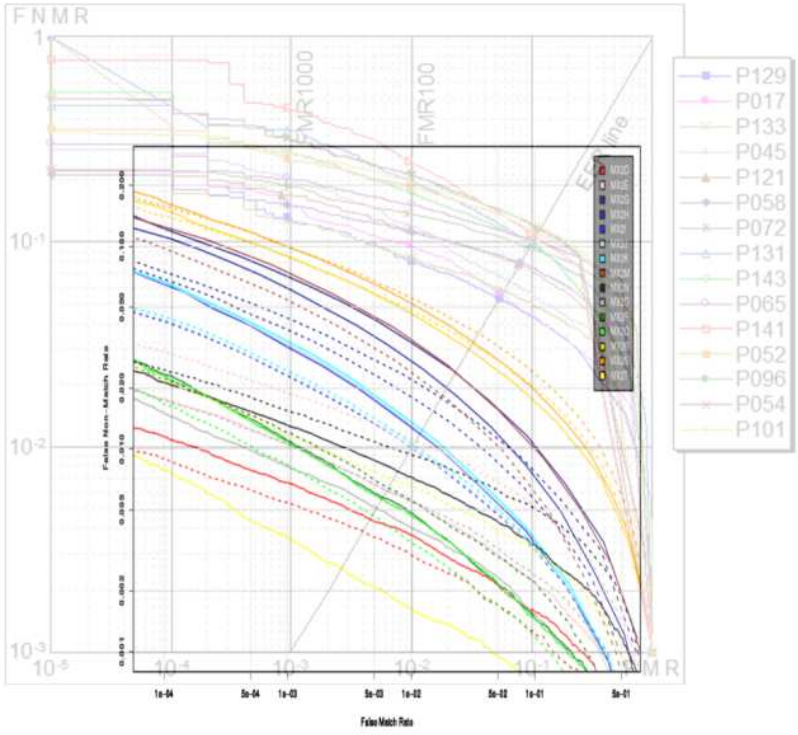


Figure 12.4: Minex II results vs state-of-the-art FVC2006 light category

12.6 Conclusion

We participated to the definition of NIST's evaluation plan since early stage as stated in acknowledgments on page 8 of NIST Interagency Report 7485 (Minex II Evaluation Plan) [47] and results with our algorithm have been published in NIST Interagency Report 7477 [49]. Our algorithm proves its suitability to smart cards or other lightweight embedded electronic platforms in terms of timing, whereas the accuracy still needs improvements, however entering in the "professional" race. We may notice that regarding our solution (MX2R & MX2S), the best results are obtained with Sagem generator (0.0175 FNMR @ 0.01 FMR, very close to the compliance specification). Extractor D (Sagem) gives generally the best results with every MoC, thus proving the superiority of Sagem for minutiae extraction.

PART IV

Security Protocols for Smart Cards with Biometrics

CHAPTER 13
A Simple Approach

Contents

13.1 Introduction

The concept of *Match-on-Card* (MoC) consists of a smart card which receives an applicant's candidate template T to be compared with the stored reference template T_{ref} by processing the complete matching algorithm during a biometric authentication request. The smart card will then output whether this comparison is positive or not. The main argument against MoC-enabled smart cards is that it opens the way for *YesCard* (i.e. an attack path previously seen in Banking, a card always returning "yes"). The threat regarding Biometrics is not only YesCard, but also *NoCard* as we will see in this chapter. We will propose a protocol to easily thwart these attacks by using simple cryptographic primitives such as symmetric encryption. This protocol will however only protect the system from malicious smart cards, but will not protect the smart card against malicious systems. Finally we will enhance this protocol to protect the smart card against its use as a so-called *oracle* to guess the stored reference biometric template.

13.2 Authentication Factors

13.2.1 Smart Card

The smart card chip contains a communication port for exchanging data and control information with the external world. It is the ideal container for cryptographic secrets such as symmetric secret keys and asymmetric private keys. The use of contactless smart card chip is now mandatory

in numerous travel documents [52] and national ID programs. Here, the role of the electronic chip is to authenticate the document (*something-we-have*) using cryptographic tools [54].

13.2.2 Password

A password is certainly the oldest and best known solution to provide user authentication. Although this sounds simple to use, we have to take care about how the password is communicated: a secure channel between the authenticator (the system or person controlling the authentication) and the applicant (the candidate user) must be available, notably at the primary exchange to set up the shared password. If these minimal precautions aren't taken, very simple man-in-the-middle attacks such as eavesdropping are possible. One of the most used password-based authentication is the PIN (Personal Identification Number) code authorizing the use of a banking card. In this case, precautions must be taken when entering the PIN code since this is very easy to spy over the shoulder of the user (attack known as "shoulder-surfing").

13.2.3 Biometrics

The biometric authentication has the advantage of checking the user's personal characteristics. The use of biometric data is now mandatory in numerous travel documents [53] and national ID programs. Here, the role of the electronic chip is to securely store user's biometric data face image being mandatory, whereas fingerprints and iris images are optional.

13.2.4 Three-Factor Authentication

Any combination of two among three authentication factors will miss at least one of the different security criteria. *Something-we-know* with *something-we-are* will miss privacy since no personal device implies the use of a database to centralize all biometric data. *Something-we-have* with *something-we-are* will miss a secret in the architecture since Biometrics are public data. *Something-we-have* with *something-we-know* will miss real user authentication since there is no proof of link between the user and his card/PIN code.

Three-factor authentication provides the highest security level in IT. Without being paranoid, some applications need to duplicate one factor in the authentication scheme: sometimes we need to show both ID card and Passport, we need to present both face and fingerprints, we need to enter the password to log in a system and then enter another password for the application we intend to use. For instance, the use of smart card, PIN code, fingerprints and facial recognition remains a three-factor authentication and not a four-factor authentication as we can sometimes read in press releases and marketing messages.

In today's digital world, most of communication channels are insecure since the first goal was to provide user convenience. When delivering a password or a biometric data, a particular attention must be paid to this communication channel to avoid very simple way to bypass authentication in the system. The use of cryptographic tools is mandatory to ensure the security of any three-factor authentication, the ultimate solution being to combine three-factor authentication with a Public Key Infrastructure (PKI). Nevertheless PKI being hard and costly to set up, manage and maintain, more simple solutions to provide secure communications over insecure channels [126] and to provide confidentiality and integrity of data [125] must be considered.

13.3 The Yescard / NoCard Issue

The YesCard is a smart card which has been maliciously modified to always answer with a positive authentication, whatever the biometric data it receives. This helps an attacker to enter in the system by presenting his own fingerprint and the biased smart card. This attack was popular few years ago in the banking area, exploiting a security flaw in ATM during off-line transaction.

Conversely, the NoCard is a smart card which has been maliciously modified to always answer with a negative authentication, whatever is the biometric data it receives. This provides denial of service for an authorized person to whom an attacker has replaced the card and then get some benefits from this situation (afterward, the attacker could impersonate the authorized user with a YesCard to enter in the system).

The Match-On-Card feature has the unique advantage of protecting the reference template of the user against capture and replay attacks (capture and replay attack with a matching candidate template is still feasible if no additional protection) by storing this reference template in a "safe". Once written at the enrollment, the smart card will never output this reference, only the candidate will be sent to the smart card to be internally compared with the reference. However, since the smart card takes the decision, the MoC feature opens the path for YesCard and NoCard. This widely used argument against Match-on-Card can be easily thwarted by the protocol described hereafter.

Firstly, we assume the use of a secure block cipher E (e.g. AES, 3DES) and a cryptographic key k shared between genuine smart cards and the system. The first idea was to use a challenge-response protocol to output the decision of the smart card: a/ if positive verification of candidate template T, the smart card send $r = E_k(c)$ (the response) where c (the challenge) is a random value given by the system together with the candidate template T b/ if negative, the smart card send any value different from r (c for instance, but then one can spy the result, $r \neq c$ is preferable). See Figure 13.1.

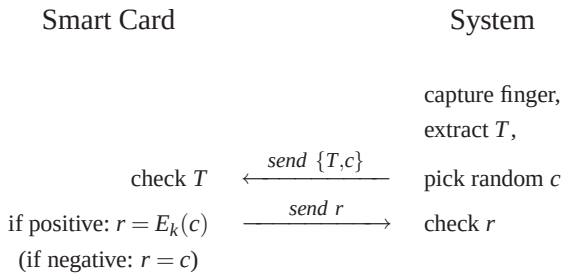


Figure 13.1: Protocol #1

However this only protects from the Yescard. Then we replace c by $c||b$, denoting the concatenation of c with a bit b where $b = 0$ if negative authentication or $b = 1$ if positive authentication. The smart card will then send $r = E_k(c||b)$. See Figure 13.2.

This protocol obviously protects from a non-authorized smart card to be a YesCard or a NoCard.

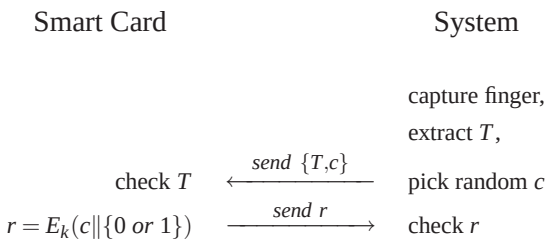


Figure 13.2: Protocol #2

13.4 The Oracle Issue

An oracle is a device or an algorithm to which we can submit questions and get answers, the oracle model is a powerful tool to evaluate the security of a system by estimating the average number of necessary queries to guess the content of the oracle. Of course, a smart card could be used as an oracle by a malicious system to guess a matching candidate T (see Figure 13.3).

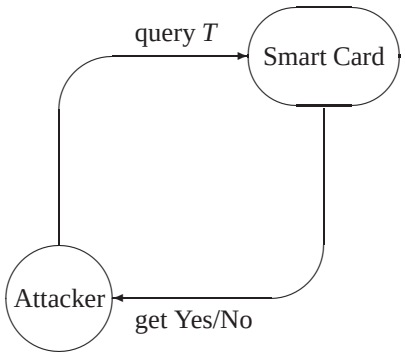


Figure 13.3: Smart Card as an oracle

We can thus enhance our protocol to resist it (even if a practical countermeasure could be the use of a try-counter of non-matching candidate T to turn off the card). The Protocol #2 does not protect against a malicious system which will send different T with always the same challenge c and analyze differences in answers (a practical countermeasure could be the comparison of the challenge received with a log table of previously used c to turn off the card). Moreover, classical side-channel attacks against smart cards could be used to find the value of the concatenated bit (which represents the decision) by carefully looking at the microprocessor operations (e.g. power consumption or processing time will differ between computation with $\|0$ or $\|1$), all other bits being known since the challenge c is transmitted in clear.

A simple way to protect the smart card against unauthorized system is to encrypt the couple $\{T,c\}$ under the shared key k. This also protects against side-channel attacks to retrieve the concatenated bit since all the other bits of the challenge c are no longer known. See Figure 13.4.

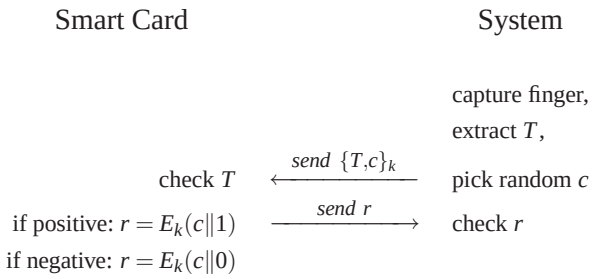


Figure 13.4: Protocol #3

13.5 Conclusion

We introduce the notion of *NoCard*, being as problematic as YesCard in the Biometrics domain and propose a protocol (Figure 13.4) that thwarts both attacks in a unique simple way. Moreover this protocol prevents the smart card from being used as an oracle by an unauthorized system to guess its biometric content. Our work has been published in [10] and proposed as a cost-effective solution within the current definition of the future European Citizen Card (ECC). However the chip targetted to be used in those cards being quite high-end, the consortium decided to use more complex and classical approach such as the one described in the next chapter.

CHAPTER 14

SBMOC - Secure Biometric Match-on-Card

Contents

14.1 Introduction to PIV - Personal Identity Verification - card

14.1.1 Framework

Authentication of an individual's identity is a fundamental component of physical and logical access control processes. When an individual attempts to access security-sensitive buildings, computer systems, or data, an access control decision must be made. An accurate determination of identity is needed to make sound access control decisions. A wide range of mechanisms are employed to authenticate identity, utilizing various classes of identity credentials. For physical access, individual identity has traditionally been authenticated by use of paper or other non-automated, hand-carried credentials, such as driver's licenses and badges. Access authorization to computers and data has traditionally been authenticated through user-selected passwords. More recently, cryptographic mechanisms and biometric techniques have been used in physical and logical security applications, replacing or supplementing the traditional credentials.

This USA PIV initiative specifies the architecture and technical requirements for a common identification standard for US Federal employees and contractors. The overall goal is to achieve appropriate security assurance for multiple applications by efficiently verifying the claimed

identity of individuals seeking physical access to federally controlled US government facilities and electronic access to US government information systems.

The PIV standard describes the card elements, system interfaces, and security controls required to securely store, process, and retrieve identity credentials from the card. The physical card characteristics, storage media, and data elements that make up identity credentials are specified in this standard. The interfaces and card architecture for storing and retrieving identity credentials from a smart card are specified in NIST Special Publication 800-73, Interfaces for Personal Identity Verification. Similarly, the interfaces and data formats of biometric information are specified in NIST Special Publication 800-76, Biometric Data Specification for Personal Identity Verification.

The PIV card as of today is a contact smart card with the following mandatory elements in the electronic chip:

- PIN code (the user always have to enter his PIN code to activate the card)
- Card Holder Unique ID (CHUID)
- Authentication data (one asymmetric key pair and corresponding certificate, using 1024-bit RSA and ECDSA)
- Two fingerprints

Depending on the sensitivity of the application, three security levels are defined:

- Some Confidence: only reads CHUID (what-you-have + what-you-know)
- High Confidence: fingerprint (one finger) authentication in unattended environment (what-you-have + what-you-know + what-you-are: three-factor)
- Very High Confidence: fingerprint (one finger) authentication in attended environment + PKI authentication (three-factor + cryptography)

14.1.2 Biometrics Implementation

Two fingerprint templates are stored on the PIV card. These templates must be compliant to ANSI INCITS 378 minutiae format (6 bytes per minutiae) and are readable upon authentication request to process the fingerprint comparison on the terminal side. The native scanning resolution of the device shall be 197 pixels per centimeter (classical 500 pixels per inch) in both the horizontal and vertical directions. The system will preferably use index fingers or thumbs and ANSI minutiae templates are prepared from images of the primary and secondary fingers. In order to improve the security level, the system may optionally request the authentication of both the primary and secondary fingers. A facial image (printed on the card body) is also digitally stored in the electronic chip for further reading and human-eye comparison between printed and stored images, no purpose of automated facial recognition here (for the moment).

14.1.3 Cryptography Implementation

PIV relies on US FIPS201 standards. FIPS201 employs cryptographic mechanisms to authenticate cardholders, secure information stored on the PIV Card, and secure the supporting infrastructure. FIPS201 and its supporting documents specify a suite of keys to be stored on the PIV Card for personal identity verification, digital signature generation, and key management. The PIV cryptographic keys specified in FIPS201 are:

- The asymmetric PIV authentication key, mandatory (RSA1024)
- A card authentication key for symmetric challenge-response, optional (2TDEA, CBC mode)
- An asymmetric digital signature key for signing documents and messages, optional (RSA1024)
- An asymmetric key management key, supporting key establishment or key transport, optional (RSA1024)
- A card management key to support card personalization and post-issuance updates, optional (2TDEA, CBC mode)

The cryptographic algorithms, key sizes, and parameters that may be used for these keys are specified in the standard. PIV Cards must implement private key computations for one or more of the algorithms identified in this section. Cryptographically protected objects specified in FIPS201 include:

- The X.509 certificates for each asymmetric key on the PIV Card (RSA1024)
- A digitally signed Cardholder Unique Identifier (SHA1 + RSA1024)
- Digitally signed biometric data (SHA1 + RSA1024)
- The *Security Object*, which is a digitally signed (RSA1024) hash table (SHA1) of all stored data

14.2 Evaluating the next generation PIV card

In order to enhance convenience for the end-user, the future generation of PIV should use contactless smart cards. Match-On-Card (with optional use of ISO19794-2 minutiae template - divides by 2 the template size, i.e. 3 bytes per minutiae) is also proposed to enhance privacy, whereas RSA2048 is targeted to enhance security. In 2007, NIST launched an RFP (Request For Proposal) to smart card manufacturers to assess the feasibility of such a card with some timing constraints: the overall authentication process should be completed in less than 1.5s if using RSA1024 and less than 2.5s if using RSA2048 (RSA public key is read in the card certificate, and used both for challenge-response authentication and symmetric session keys agreement).

The following sections describe our contribution to this topic and our support to NIST.

14.3 Security Framework

Currently, FIPS 201 permits biometric data to be released only across the contact interface of a PIV Card, and only after activation of the PIV Card through presentation of the cardholder's PIN. These restrictions achieve two security objectives: communication of biometric data occurs only over a trusted communication channel that is not easily subject to eavesdropping attacks (namely, the wired contacts inside the smart card reader); and the PIV cardholder implicitly attests to the legitimacy of the smart card reader, as they indicate by entering the PIN on the smart card reader keypad. FIPS 201 enables biometric authentication to occur without imposing a technical requirement for automatic authentication of smart card readers to PIV Cards. Such a requirement, it was believed, would add unacceptable key management costs (the PIV fingerprint object is digitally signed, and the signature can be used to verify authenticity and integrity of the data). This feasibility study evaluated the impact of contactless smart card, Match-on-Card and secure protocol on transaction performance, when the protocol meets these security objectives (SO):

- SO1: communication of biometric data shall occur only over a trusted channel that is not susceptible to eavesdropping attacks in the reader-to-card direction, nor spoofing or replay attacks in the card-to-reader direction
- SO2: communication of biometric data between the smart card and smart card reader shall occur only after the cardholder has indicated the reader is legitimate
- SO3: communication of biometric data from the smart card to the reader shall occur only after the cardholder has entered their PIN
- SO4: the approach should achieve the preceding security objectives without reader-to-smart-card authentication or associated key management infrastructure

These security objectives are aligned with the high-level security objectives of FIPS 201. They protect both the integrity of the biometric authentication transaction and the privacy of the cardholder's biometric data, whereas avoiding the potential cost of reader authentication key management.

Figure 14.1 depicts the basic principle of SBMOC: (1) establish a secure session, (2) smart card receives candidate template and process comparison and (3) smart card sends the *signed* OK/NOK decision.

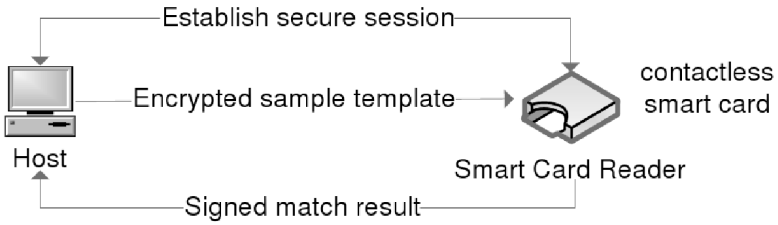


Figure 14.1: SBMOC principle

14.4 Our Protocol

The main challenge here is to overcome the problem of potential, and easy, Man-in-the-middle attack or replay attack inherent to contactless communication, whereas the current generation of PIV Cards bases its security against these threats on the difficulty to discreetly probe the card's contacts. We proposed the use of a card's asymmetric key pair to process card authentication and to agree on two session symmetric keys for biometric data encryption and MACed decision. Figure 14.2 summarizes the exchange of commands between the user/reader and the card. It describes in sequence (from 1 to 10) the command exchanges (in/out) and the main security data associated. On the left part of the figure, are described the internal reader processes during the authentication protocol. The card internal processes are described on the right part of the figure:

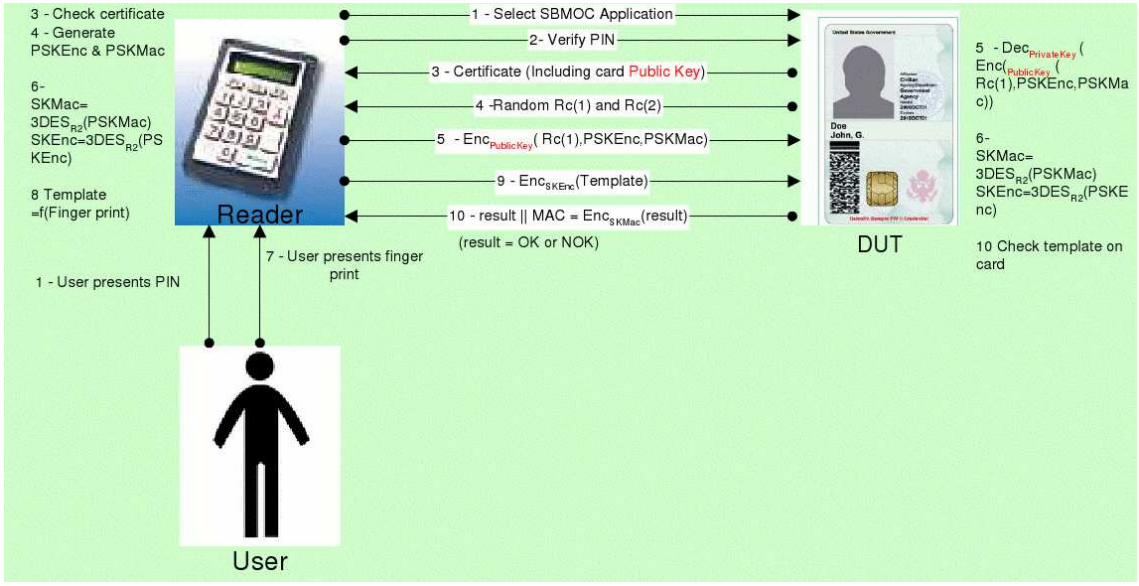


Figure 14.2: SBMOC framework

Table 14.1 summarizes the analysis of the authentication protocol against the NIST security objectives:

Objective ID	Objective description	Sequence(s) that answer the objective
SO1	Eavesdropping attacks Eavesdropping attacks Replay attacks	Sequences 5, 9 Sequences 4 to 9 Sequences 3 to 10
SO2	Reader is legitimated before biometric data transmission	Sequence 2*
SO3	PIN is verified before biometric data transmission	Sequence 2
SO4	No reader authentication and key management	Full protocol

*visual authentication only, by the user accepting to enter the PIN code

Table 14.1: Our protocol vs NIST Security Objectives

14.5 Our Implementation on Smart Card

For the smart card implementation of our protocol we used on-board key pair generation and the following APDU commands:

- Select SBMOC applet
  - This command is used to select the SBMOC application with a multi-application javacard
- Verify PIN
  - This command is used to verify the PIN code
- Read RSA Public Key
  - This command is used to retrieve the RSA public key from the Smart Card
- Write X509 Certificate
  - This command is used to write the X509 certificate in the Smart Card
- Enroll Biometric Data
  - This command is used to enroll biometric data, i.e. the reference fingerprint template
- Read X509 Certificate
  - This command is used to retrieve the X509 Certificate from the Smart Card
- Get Challenge
  - This command is used to receive a 24-Byte random (8-Byte Rc1, 16-Byte Rc2) generated by the Smart Card
- Send External Challenge
  - This command is used to send two 16-byte random PSKenc and PSKmac in the Smart Card to compute two 128-bit symmetric session keys for encryption and MAC
- Verify Biometric Data
  - This command is used to verify biometric data, i.e. compare the reference fingerprint template VS the deciphered candidate fingerprint template and send MACed decision

We used here the usual PIV electronic chip with the classical PIV application firmware on the top of a javacard operating system and we enhanced its features: Match-on-Card algorithm in programmable ROM, activation of the contactless communication interface, activation of the on-board key pair generation within the embedded cryptoprocessor.

Once delivered to the NIST with previously seen commands, each smart card must be activated and personalized before the testing campaign. This splits on two phases: (1) initialisation and (2) testing campaign.

Here is the card initialisation process (done once at card delivery):

- Terminal: selects SBMOC applet
- Card: requests PIN verification
- Card: on-board RSA key pair generation
- Terminal: reads RSA public key and generate the X509 certificate
- Terminal: writes X509 certificate within the card
- Terminal: captures reference fingerprint and writes the reference template within the card

This process consumes about 20s to 30s because of the on-board key pair generation.

Here is a user authentication process (normal use during card lifecycle):

- Terminal: selects SBMOC applet
- Card: requests PIN verification
- Terminal: reads X509 certificate within the card
- Terminal: gets challenge from the card
- Terminal: sends challenge to the card
- Terminal & Card: compute session keys
- Terminal: captures candidate fingerprint, sends encrypted candidate template
- Card: decrypts candidate template, compares, sends decision
- Terminal: verifies decision and MAC

14.6 Results

The timing metrics obtained from the SBMOC feasibility study showed that it is possible to securely perform biometric match-on-card operations over the contactless interface of a smart card within 2.5 seconds, even when using RSA2048. The study also showed that the amount of time required to complete an SBMOC operation is dependent on a number of factors, such as the cryptographic mechanisms used, the minutia count of the reference and candidate fingerprint templates, and the format of the fingerprint templates.

The feasibility study participants assess that some of the cards were constructed by adding firmware and data to PIV card stock that had already passed NIST and FIPS certifications. These remarks are further evidence that SBMOC is technically feasible through firmware extensions to existing smart cards.

Timing results with our protocol and card implementation are depicted in Figures 14.3 and 14.4:

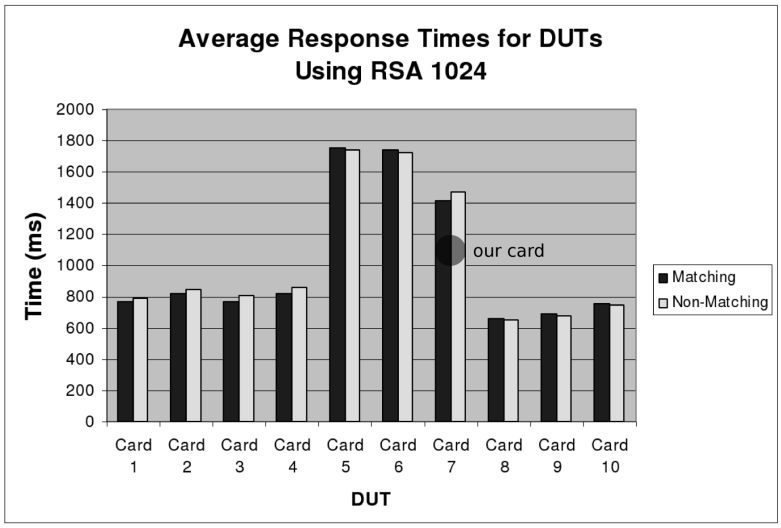


Figure 14.3: Timing results with RSA1024

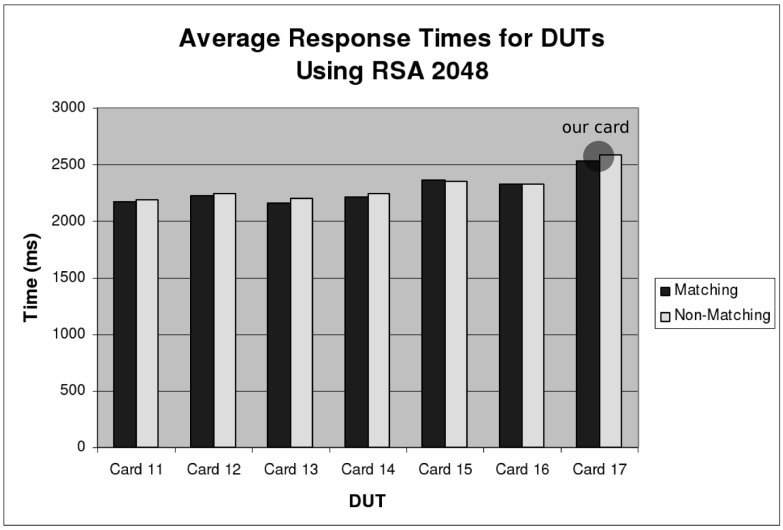


Figure 14.4: Timing results with RSA2048

14.7 Conclusion

The results have been published in NIST Interagency Report 7452 [27]. Four entities participated in the test, all demonstrated the feasibility of this targeted future generation of PIV Cards. Acknowledgement for our contribution appears under the name Gemalto on page 4 of the NIST document.

Our contribution was both the proposal of the SBMOC protocol and its implementation in a contactless smart card chip. For technical reasons we were obliged to use ANSI minutiae format instead of ISO minutiae format in our available smart cards, this justifies our not so good timing results because of the in-card need to decipher double-sized data. This is particularly sensible with RSA1024 cards, however we fully enter in the timing specifications. As for our protocol, NIST didn't disclose other three competitors approaches of the secure protocol; a different protocol may also justify differences in timing results.

PART V

Biometrics & Cryptography Interaction

CHAPTER 15
General Introduction & State-of-the-Art

Contents

15.1 Introduction

Personal biometric data are often being stored in large databases, moreover the same biometric data is being used in different applications, thus raising questions such as data security and data privacy. The cryptography toolbox is usually the answer to such questions, however dealing with biometric data will lead to some complications.

A major security issue with biometrics is the need, in most currently deployed systems, to compare the reference and the candidate both in clear forms. We are only able to compute a matching in the clear domain, but not in a hash domain, nor in a cipher domain, nor in any other secure domain. In opposition to usual password schemes where only the hash value of the password is stored and compared with the hash value of the candidate password, the inherent nature of biometrics makes it impossible to always capture the same digital data (i.e. bit per bit) due to differences linked to human interaction, sensors technologies, environmental conditions and so on at each authentication request, thus making it impossible to hash a biometric data in a deterministic way.

Beyond the impossibility to precisely reproduce biometric data at each capture, biometrics data

are not uniformly distributed: a lot of improbable values for the template bitstream (e.g. all minutiae concentrated in only one corner of the image, more than two eyes in a face...) is sensibly reducing the search space for the attacker. And uniform distribution is a key element, for instance, in cryptographic key security against attacks.

This topic, aka *crypto-biometrics*, is quite new in the research community. And unfortunately much hyped in the industry, often resulting in unrealistic implementation and uses cases due to misunderstanding of this complex interaction between cryptography and biometrics.

Before discussing about serious approaches coming from the research community in the following sections, we will depict these "too straightforward" applications coming from, let's say, marketing people as illustrated in figure 15.1 coming from the homepage of a vanished (about 2003-2006, rest in peace) start-up company website: no more passwords, no more keys, no more cards, oops!

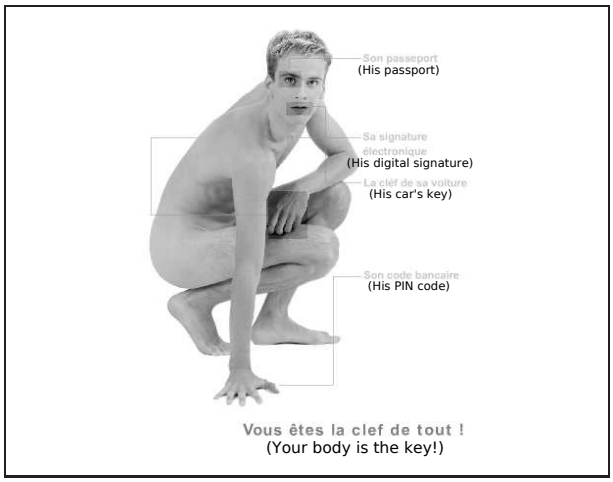


Figure 15.1: The crypto-biometrics utopia

Well, is this realistic?

Among these "too straightforward" ideas we may cite (even assuming the existence of a *deterministic* biometric solution):

1/ "my fingerprint is my password / is the secret key of a symmetric cryptographic system"

. regarding security issues with biometrics data discussed in part II, is it safe to let your secret key being a public data, seriously? And what about revocation?

2/ "my fingerprint is the private key of an asymmetric cryptographic system"

. same comment as above

3/ "my fingerprint is the public key of an asymmetric cryptographic system"

. for sure, no privacy issue here, but is it usefull? You will never have to use your own public key (apart from rarely verifying your own signature to ensure no impersonating attack was conducted during the signing process). However we will discuss

about few niche applications where this may be usefull (e.g. IBS, described later). It's important to note here that a fingerprint being used as a public key can no longer be reasonably used for biometric authentication.

Changing identity is even mandatory in few legal cases (e.g. protecting political refugees, crime witnesses). The biometric data must be protected. Once this first assumption accepted, the trivial idea is to encrypt this sensitive data, secure the key, and decipher only in a secured environment to process the matching. Then how to conveniently secure the key? Reverting the previous scheme, some solutions propose to protect the key with the biometric data (e.g. positive biometric recognition grants access the private key to process a digital signature) and in such a case the end user will wrongly have the feeling to directly sign with his fingerprint. So biometrics to protect cryptography or cryptography to protect biometrics? This is a kind of chicken & egg situation.

Less than a decade ago, the biometrics research community and the cryptography research community had quite different approaches to this interaction between biometrics and cryptography. *Biometricians* tended to pass security issues to cryptographers, whereas cryptographers tended to consider biometric data just as any other noisy data, applying an error-correcting code approach to "stabilize" biometrics [31]. Actually for about seven years now, open discussions between biometricians and cryptographers led to interesting cryptography and biometrics mixing schemes.

Here are the expected features of any serious *privacy-concerned* systems, with the notion of *secured representing sample*:

- The original biometric data can't be retrieved from its secured representing sample, even with a trapdoor (one-way)
- The *bio-protected* secret, if any, can't be retrieved without the trapdoor (i.e. the positive matching between the *alive* candidate biometrics and its secured representing sample)
- Each application must have its own secured derivation of the original representing sample
- The secured representing sample may be revoked and renewed
- The recognition process takes place in the secured domain
- The weak binary property of the biometric decision (i.e. yes, no) is well protected, hidden or ideally useless for the system

Maturity of discussions between these two parties (i.e. biometricians, cryptographers) highlights some bio-crypto primitives to develop: extract reproducible data from biometrics (fuzzy extractors), biometric hashing (one-wayness), cancelable biometrics (revocation & regeneration), homomorphic encryption schemes (matching in secured domain), template obfuscation (retrieve a biometric-based secret only with the related biometrics presentation) and so on. We will lightly describe these primitives in the following sections. An interesting recent book in the domain (and also about *Physical Unclonable Functions* -PUF-) is "Security with Noisy Data" by Tuyls *et al* [119].

15.2 Hamming Distance

Hamming Distance (HD) appears to be a straightforward method for measuring the closeness of digital data samples: HD is the Hamming Weight (HW, number of "1") of the exclusive-or operation between the two bitstrings. The lower the HD, the closer are the samples (i.e. when the compared bits at the same location have the same value, they result in a "0" value when XORing). Moreover, computing the Hamming Distance in a secured domain is trivial (e.g. One-Time Pad encryption). This distance is widely used in the different schemes depicted in the next sections.

15.3 Fuzzy Extractors

Coming from pure cryptographers, several notions enter in this category: Fuzzy Extractors and Secure Sketch by Dodis *et al* [34], Fuzzy Vaults by Juels *et al* [65], Fuzzy Identity-Based Cryptography derived from Identity-Based Cryptography [110].

The Fuzzy Extractor construction is addressing both error-tolerance for the entry data and uniformity of the output data: this is the process of extracting a uniformly random string R from a biometric input T (e.g reference template) in such a way that the system is able to exactly retrieve the same R with any biometric input T' enough close to T (figure 15.2). Less theoretical and more practical, the Secure Sketch construction is only addressing error tolerance and discloses some information about the entry data T by the needed publication of a parameter p at enrollment which will be recombined with any T' to exactly recover T at any verification request (figure 15.3). The parameter p could be, for instance, learned from multiple capture of the original biometric data (T, T_1, T_2 to T_n). Secure Sketch seems to be a stepping stone to achieve a practical Fuzzy Extractor.

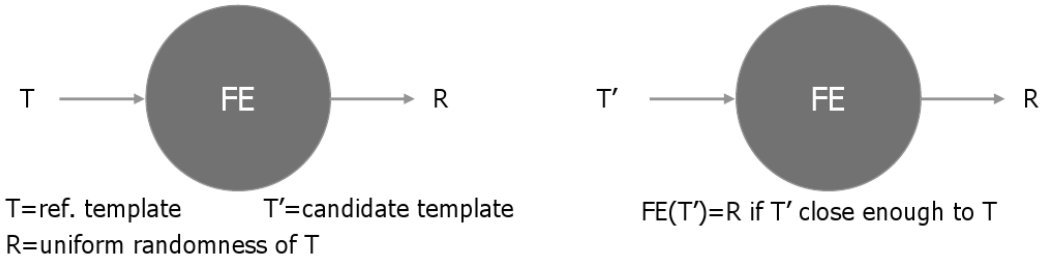


Figure 15.2: Fuzzy Extractor at Enrollment and Verification

Major inspiration for Fuzzy Extractor research, the Fuzzy Vault construction proposes Alice to put a secret in a vault and lock it with an unordered set A, then Bob could open the vault if and only if he proposes a set B that substantially overlap A. The given example based on a list of prefered movies is presumed to be adaptable to any noisy but close data, hence biometrics. This allows Alice and Bob to conveniently share a secret. Several research are adapting this notion of Fuzzy Vault to fingerprint data, for instance Uludag &Jain in [122, 120].

The notion of Identity-Based Encryption (IBE) and Identity-Based Signature (IBS), with its recent practical construction [15], where the public key is fixed and easily known (e.g. email

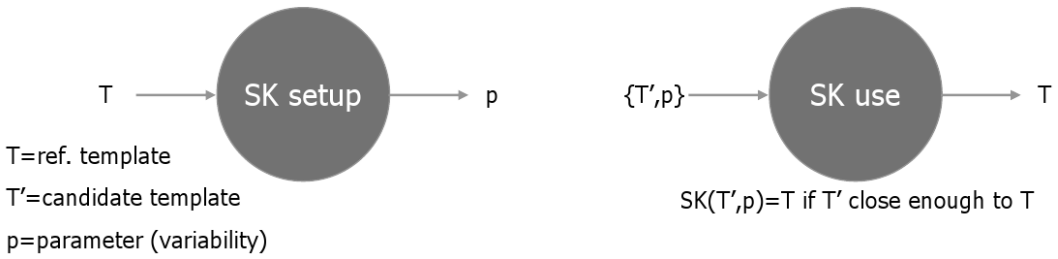


Figure 15.3: Secure Sketch at Enrollment and Verification

address) had extension such as fuzzy IBE [103] and fuzzy IBS [19] where the public key is the biometric data of the signer: the signer may actually prove he is the right signer just by presenting its biometrics to the signature verification system. These schemes differentiate from the usual fuzzy extraction in that sense they consider biometric data as the public key rather than trying to extract a stable secret from the biometric data.

15.4 Cancelable Biometrics

The cancelable biometrics concept has been introduced by Ratha *et al* at IBM Watson Research[92], recent implementations with faces and fingerprints have been discussed in [91, 94]. This approach proposes to apply a non-reversible transformation to biometric data in such a way that different captures of the same biometric data under the same transformation could be matched in the transformed domain, as depicted in figure 15.4. This allows different applications to use the same biometric data with different transformation parameters, with revocation ability and renewability. These transformations could be applied either at signal level (i.e. before extraction, e.g. face image hereunder) or feature level (i.e. after extraction, e.g. fingerprint minutiae template hereunder).

The non-reversibility of the transformation is claimed by the authors: retrieving the original data is computationally hard, randomly guessing the input data being as efficient as any other technique. Coming from pure biometricians, this interesting technique however lacks cryptographically proven irreversibility and general security (e.g. collision resistance). There is no real interaction with cryptography here.

15.5 Biometrics Hashing

As its name clearly suggests, this approach targets stabilizing the biometric data to apply classical cryptographic hash functions. Practical implementations have been introduced by Tuyls *et al* at Philips Research [118, 117]. This actually combines fuzzy extraction and salting (just as Unix password construction) techniques to stabilize the biometric data and then derive a unique identifier that could be revoked and renewed. The first step is using the so-called *helper data* to retrieve the original biometric, this helper data being constructed at enrollment, just as the p parameter of the secure sketch in fuzzy extractors. The scheme is depicted in figure 15.5, where W is the helper data, $F()$ is the salting and hashing function, *enc* and *dec* are the secure sketch

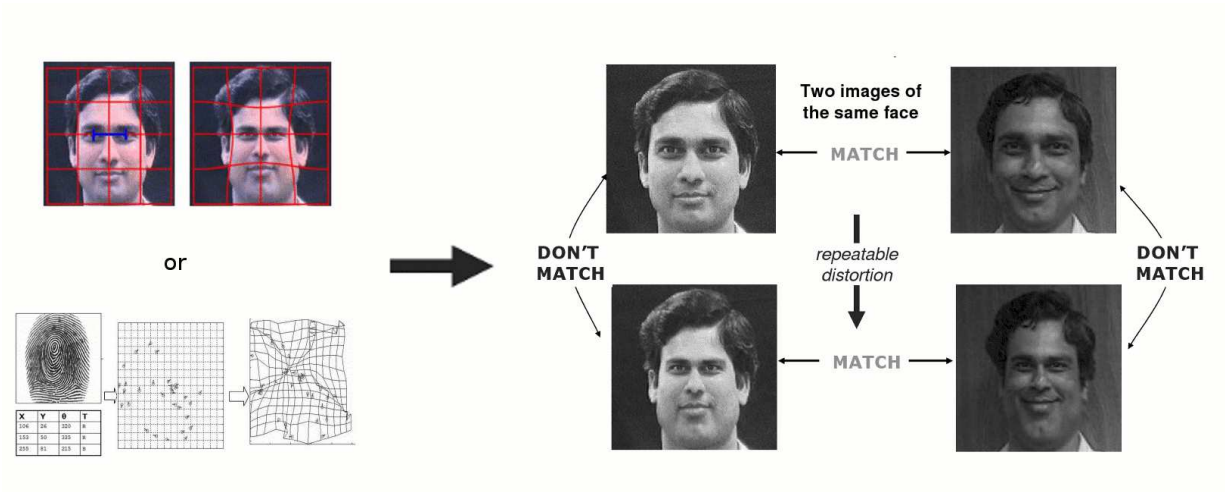


Figure 15.4: Transformation-based Cancelable Biometrics

functions, X^n and Y^n are two instances of the same biometric data (often referred to as T and T' in this dissertation).

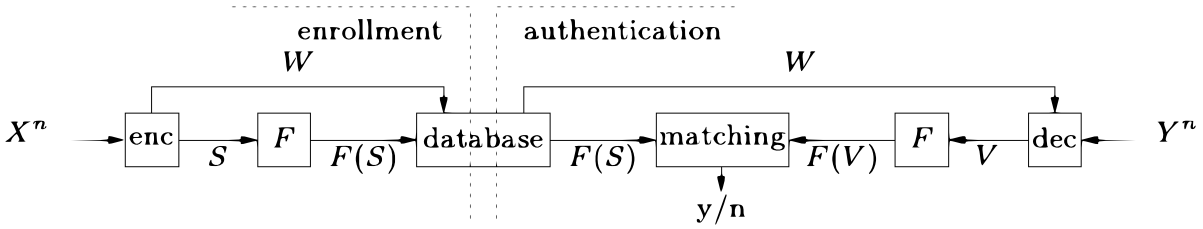


Figure 15.5: Hash-Based Template Protection Technique

Another interesting technique, particularly dedicated to point-pattern data, comes from Geometric Hashing [131], classically used for model-based object recognition applications and efficient search in large image databases. One practical implementation dedicated to realign two minutiae-based fingerprint representations is reported in [24], this could be a first step to extract and represent stable data from a biometric input.

15.6 Biotopes & Biotokens

Terry Boult *et al* have proposed their so-called *Biotopes* approach since 2007 [16, 105, 93] as well as their application in revocable biotokens. One particular interesting property here is the ability to build a *most private* version of a biotope that would only suit verification schemes and would be useless for identification schemes. In opposition to other protection schemes (decreasing matching performances by a factor of two in the best cases), the authors claim their scheme to even improve the performances of the underlying matching algorithm by an order of

magnitude of 30%. This approach is not straightforward and seems to really mix cryptography and biometrics, with demonstrated applications to face and fingerprints.

Interestingly, Scheirer and Boult also introduced one of the first papers regarding attacks and security analysis of crypto-biometrics systems and privacy-enhanced templates [106], proving their real position in the middle of cryptographers and biometricians.

15.7 Intricated Biometrics

In the Encyclopedia of Biometrics [70], Mainguet introduces the notion of intricated biometrics. Here both the biometric template and the secret are mixed in the same public template. This general concept (figure 15.6) have the two compulsory properties: (i) can retrieve neither the original biometrics nor the secret from the intricated template, (ii) the secure environment using the intrincated template and a fresh biometric capture is able to both verify identity and extract secret without the leakage of the final entropy bit yes/no.

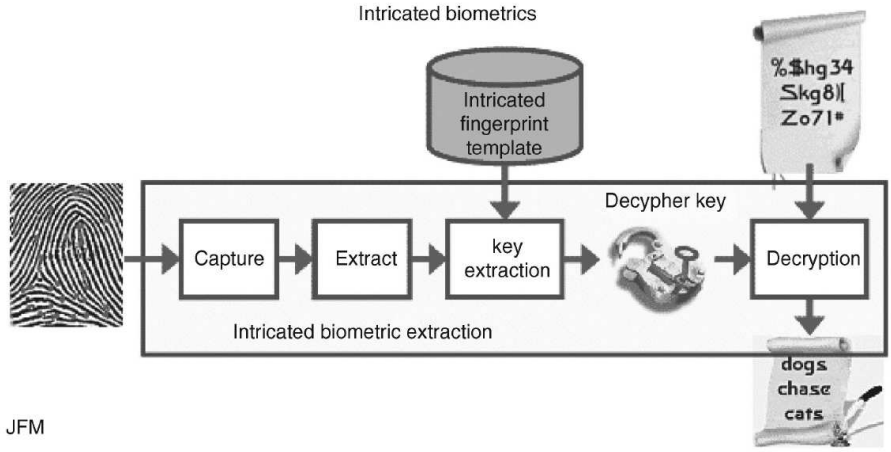


Figure 15.6: Intricated Biometrics

No implementation is given, however this construction could be achieved with concepts depicted in previous sections.

15.8 Homomorphic Encryption for Biometric Data

Coming from cryptographers, the idea is to adapt classical homomorphic encryption schemes to biometric data to reach the goal of being able to match data directly in the encrypted domain. Homomorphic encryption have the interesting property that an operation with cleartexts has a corresponding (different) operation with ciphertexts: the result with ciphertexts is equivalent to the encryption of the result with cleartexts. Known schemes are Goldwasser-Micali, Naccache-Stern or Paillier cryptosystems [46, 81, 85]. Application to biometrics is introduced by Bringer *et al* [18] and Schoenmakers and Tuyls [107].

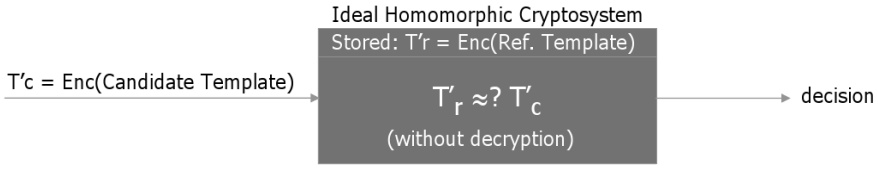


Figure 15.7: Ideal homomorphic encryption scheme for biometric data

15.9 Biometric Data Obfuscation

Still coming from cryptographers, the idea here is to apply information hiding techniques, such as steganography, to obfuscate the sensitive biometric data in other data or add some random (or not) noise to the template. Required property is non-invertibility (i.e. retrieving original biometric data or obfuscation key from the noisy template) without a trapdoor (e.g. the original biometric data or obfuscation key). *Obfuscation key* actually refers to the noise generating technique and parameters. A current technique applied especially to minutiae-based fingerprint templates is the use of so-called *chaff points*: false minutiae are added to the set of real minutiae extracted from the fingerint image [25, 66, 133], and a secret is the trapdoor to separate false from real minutiae. We particularly exploited this technique since 2002, patented by Barral *et al* in 2003, and published in 2004, to derive a secure and fast Match-on-Card technique suitable to very low-end processors [8]. This application will be fully depicted in the next chapter.

15.10 Use Cases

15.10.1 Pseudo-Identities

As previously discussed, one of the goals is to provide revocable and renewable identities, different for each application, so-called *pseudo-identities* studied in the European funded project "TURBINE" (stands for TrUsted Revocable Biometric IdeNtitiEs, www.turbine-project.eu). Requested properties of such pseudo-identities (PI) are:

- Non-invertibility: can't retrieve the original biometric data from the PI
- Collusion resistance: can't even retrieve the original biometric data from many PIs derived from it
- Revocation: a compromised PI could be dismissed
- Renewable: many PIs could be generated from the same biometric data without compromising it
- Cross-Matching resistance: different PIs from the same biometric data could not be matched
- Secure Domain Matching: at authentication stage, only the original biometric data is transformed to the PI domain for comparison

All above described techniques (apart from pure fuzzy extractors) are able to produce pseudo-identities. This is a clear advance in the domain of protecting personal biometric data, finding

its applications in all civil ID schemes, e.g. ID cards and passports. The TURBINE project is currently proposing a relevant standard initiative (Security Techniques - Biometric Template Protection) at ISO level [61].

15.10.2 Efficient Duplication Checking

Once one should be able to hash biometric data, the straightforward idea is to use well known Look-Up Tables techniques, coming from large databases management, for efficient search and comparison in biometric references. Even in 1-to-1 applications, the enrollment stage often needs a 1-to-n capability in order to detect duplications of identities in the same system. This AFIS-like feature usually requires the management of a large database (with its inherent privacy concerns), a powerfull server and could be seen, roughly speaking, as a n times 1-to-1 matching, hence not so efficient. Based on biometric-hashed values, this feature could be implemented at nearly no cost with indisputable efficiency: just a comparison of few hundreds of bits, only exact (i.e. bit per bit) match will raise an alarm.

15.10.3 Other Use Cases

These crypto-biometrics techniques could be of interest for any application that would need very small digital data to save storage space, communication time, communication bandwidth, communication security and that will use cost-effective embedded electronic devices to store and process these privacy-enhanced reference biometric data.

CHAPTER 16

Biometrics-Based Challenge-Response: BioEasy

Contents

16.1 Introduction

BioEasy introduces a new concept for highly secure biometrics verification on a smart card. It provides a unique alternative to match-on-card that allows fast and easy biometrics authentication on the smart card, without compromising security or memory space.

Secure: BioEasy delivers a simple yet highly secure way to deploy a biometric infrastructure, while still performing the verification on the actual smart card (rather than on the PC or reader) for maximal security and privacy. The technology uniquely externalizes heavy computations on the terminal, while still having the card handling the security issues. The only computation done by the smart card is simple comparison between two bit strings, hence working just as fast and easy as basic PIN code verification, but involving biometric data. In our prototype, it only takes a few tens of milliseconds (ms) in Java Card code to perform the bit string comparison, in comparison to several hundred ms for classical match-on-card algorithms in native code.

Principle: BioEasy is designed to perform biometrics authentication on the card without compromising security, speed or memory size. So how does it work? The smart card stores 2 types of data: First, the card stores a public data - a.k.a. “obfuscated template”, which is based on the end-users real minutiae and a set of added false minutiae generated by the enrolling authority. Second, the card stores a secret data - the “BioEasy code” - acting as the key to interpret the obfuscated template in order to distinguish between real and false data. The BioEasy code will never be communicated outside of the smart card. The enrollment process begins with the reader extracting minutiae from the enrolling user’s fingerprint. This extracted data is then

obfuscated by adding false minutiae. Next, for authentication, the card outputs the obfuscated template to the reader. The reader then utilizes the user's original fingerprint minutiae to determine what data is real/false, and based on that sends back a candidate response to the card. Finally, the BioEasy code stored on the card is used to validate whether the terminal has correctly interpreted the challenge, i.e. if the errors reported by the reader match the errors sent by the card. If the authentication is positive, the cardholder is permitted to access the system.

Privacy: In addition to the obfuscated minutiae, a "validity vector" (the BioEasy code) is stored in the protected memory of the smart card, informing the BioEasy program of the valid and false data. This validity vector, stored when the enrollment is performed, never leaves the memory of the smart card and thus protects the privacy of the user by not revealing the real minutiae.

Small footprint: since few computations have to be performed by the smart card, BioEasy can be implemented through a relatively small Java applet (less than one kilobyte), allowing additional applications to be added onto the card. For ultimate cost efficiency, this approach even suits a simple memory card with dedicated glue logic for the bitstring comparison.

Open platform: by operating on an open Java Card platform, an issuer can easily add/remove/update applications and data on the card, even after it has been issued - so called "post-issuance". Moreover, since few resources are used by the BioEasy applet, it also allows the issuing organization to introduce biometrics in Java-based cards that have already been deployed (given that there is memory available). However, for low-cost applications where post-issuance capabilities are not required (e.g. nationwide ID programs for countries with large populations), a simple memory card can be used to support the BioEasy authentication technology.

Ubiquity: BioEasy is suitable for any biometric technique, e.g. fingerprint, iris, face recognition etc., and can be used for the interoperability of all proprietary solutions known on the market. Moreover, this approach is an additional layer for any classical matching engine, not competing with, but rather complementing it. The BioEasy technology is ideal for applications where identity (e.g. passport, driving license) is involved. The main advantage is the possibility to have one biometric company doing the enrollment and another doing the verification, as long as they are using compatible templates and are in compliance with the open standards published by ISO.

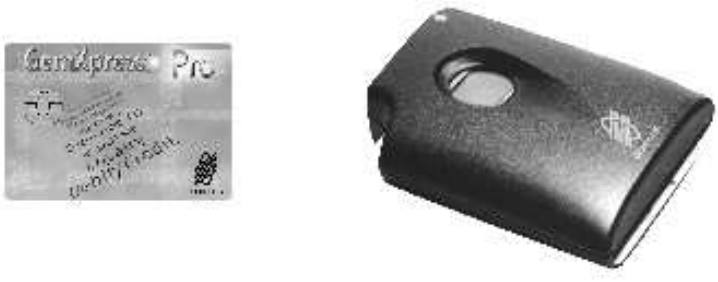


Figure 16.1: Smart Card and Fingerprint/Smart Card Combo Reader

16.2 Issues with Classical Match-on-Card

Generally, a fingerprint biometric authentication system comprises four main modules:

- A capture unit, which acquires the raw biometric fingerprint data IM of an individual (typically a bitmap of the finger's ridges).
- A feature extraction module $Extract()$ in which the acquired biometric data is processed to extract a feature-set $Extract(IM)$ that models IM. Typically $Extract(IM)$ is the position and orientation of minutiae.
- A matching module $Match()$ in which an extracted feature-set $Extract\ (IM_{cand}\ )$ can be compared to a reference pattern $Extract\ (IM_{ref}\ )$. This comparison process outputs a score $0 \leq Match\ (TP_{cand}\ ,\ TP_{ref}\ ) \leq 1$.
- A decision-making module in which the user's claimed identity is either accepted or rejected based on the matching score: if $Match\ (TP_{cand}\ ,\ TP_{ref}\ ) > T_h$ return accept else return reject. T_h is an application-dependent security parameter, often referred to as the threshold. When a matching score exceeds T_h, the two feature sets are declared as belonging to the same individual; otherwise, they are assumed to belong to two different individuals.

A biometric smart card has the capacity to store a reference template $Extract\ (IM_{ref}\ ) = TP_{ref}$ in EEPROM and perform both matching and decision-making when presented with a candidate template $Extract\ (IM_{cand}\ ) = TPcand$. Typically, an acceptance will "open" the card and permit access to some of its EEPROM files, enable the generation of a digital signature or debit a purse. It is customary to require that these steps take place in less than a second (convenience). When coded in a card a matching algorithm would use at least 2kB of RAM. The code would usually occupy 2kB to 12kB of ROM. The code complexity (matching usually involves many floating-point trigonometric operations) and RAM consumption are two decisive cost factors in the design of such solutions.

Problem Formulation

Let $Extract\ (IM_{ref}\ )$ and $Extract\ (IM_{cand}\ )$ be the representation of the reference template and candidate template, respectively. Here the representation $TP = \{m_1, m_2, m_3, ..., m_n\}$ is a feature vector (of variable length) whose elements are the fingerprint minutiae. Each minutia may be described by a number of attributes, including its location in the fingerprint image, orientation, type (e.g. ridge termination or ridge bifurcation), a weight based on the quality of the fingerprint image in the neighborhood of the minutia, and so on. Most common minutiae matching algorithms consider each minutia m as a triplet $m\{x, y, \theta\}$ that indicates the x, y minutia location coordinates and the minutia orientation θ (minutia type information being not sufficiently interoperable):

$$\begin{array}{lcllll} TP_{ref} & = & \{m_1, m_2, \cdots, m_n & \} & , & m_i = \{x_i, y_i, \theta_i\} \quad , \quad i = 1 \cdots, n \\ TP_{cand} & = & \{m'_1, m'_2, \cdots, m'_{n'} & \} & , & m'_i = \{x'_i, y'_i, \theta'_i\} \quad , \quad i = 1 \cdots, n' \end{array}$$

where n and n' denote the number of minutiae in TP_{ref} and TP_{cand} , respectively.

A minutia $m'_j \in TP_{cand}$ and a minutia $m_i \in TP_{ref}$ are considered matching, if the *spatial distance* (sd) between them is smaller than a given tolerance r_0 and the *direction difference* (dd) between them is smaller than an angular tolerance θ_0:

$$\begin{aligned} \mathrm{sd}(m'_j, m_i) &= \sqrt{(x'_j - x_i)^2 + (y'_j - y_i)^2} \leq r_0 \quad (1) \\ \mathrm{dd}(m'_j, m_i) &= \min(|\theta'_j - \theta_i|, 360^\circ - |\theta'_j - \theta_i|) \leq \theta_0 \quad (2) \end{aligned}$$

Equation (2) takes the minimum of $|\theta'_j - \theta_i|$ and $360^\circ - |\theta'_j - \theta_i|$ because of the circularity of angles (the difference between angles of 2° and 358° is only 4°). The *tolerance boxes* defined by r_0 and θ_0 are necessary to compensate for the unavoidable errors made by feature extraction algorithms and to take into account the small plastic distortions that cause the minutiae positions to change.

Aligning the two fingerprints is a mandatory step in order to maximize the number of matching minutiae. Correctly aligning two fingerprints requires *translation* (in x and y) and *rotation* (θ) to be recovered, and frequently involves other geometrical transformations:

- *scale* has to be considered when the resolution of the two fingerprints may vary (*e.g.* the two fingerprint images have been taken by scanners operating at different resolutions)
- other *distortion-tolerant* geometrical transformations could be useful to match minutiae in case one or both of the fingerprints is affected by severe distortions

In any case, tolerating a higher number of transformations results in additional degrees of freedom to the minutiae matcher: when a matcher is designed, this issue needs to be carefully evaluated, as each degree of freedom results in a huge number of new possible alignments which significantly increases the chance of incorrectly matching two fingerprints from different fingers, and also has a huge impact on processing time.

Let map$(.)$ be the function that maps a minutia $m'_j \in TP_{cand}$ into m''_j according to a given geometrical transformation; for example, by considering a translation of $[\Delta x, \Delta y]$ and a counterclockwise rotation δ around the origin [‡]:

$$\mathrm{map}_{\Delta x, \Delta y, \delta}(m'_j) = m''_j = \{x''_j, y''_j, \theta'_j + \delta\}$$

where

$$\begin{bmatrix} x''_j \\ y''_j \end{bmatrix} = \begin{pmatrix} \cos\delta & -\sin\delta \\ \sin\delta & \cos\delta \end{pmatrix} \begin{bmatrix} x'_j \\ y'_j \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

[‡]The origin is usually selected as the minutiae centroid (*i.e.* the average point); before the matching step, minutiae coordinates are adjusted by subtracting the centroid coordinates.

Let $\zeta(.)$ be an indicator function that returns 1 when the minutiae m''_j and m_i match according to Equations (1) and (2):

$$\zeta(m''_j, m_i) = \begin{cases} 1 & \text{if } \ \mathrm{sd}(m''_j, m_i) \leq r_0 \ \text{ and } \ \mathrm{dd}(m''_j, m_i) \leq \theta_0 \\ 0 & \text{otherwise} \end{cases}$$

The matching problem can be formulated as:

$$\underset{\Delta x, \Delta y, \delta, P}{\text{maximize}} \quad \sum_{i=1}^{n} \zeta(\mathrm{map}_{\Delta x, \Delta y, \delta}(m'_{P(i)}), m_i) \qquad (3)$$

where $P(i)$ is an unknown function that determines the pairing between TP_{ref} and TP_{cand} minutiae; in particular, each minutia has either exactly one mate in the other fingerprint or has no mate at all:

1. $P(i) = j$ indicates that the mate candidate of the $m_i \in TP_{ref}$ is $m'_j \in TP_{cand}$
2. $P(i) = \perp$ indicates that $m_i \in TP_{ref}$ has no mate in TP_{cand}
3. an $m'_j \in TP_{cand}$ such that $\forall i = 1, \cdots, n \quad P(i) \neq j$ has no mate candidate in TP_{ref}
4. $\forall i = 1, \cdots, n \ \ \forall k = 1, \cdots, n' \ \Rightarrow P(i) \neq P(k)$ or $P(i) = P(k) = \perp \ with \ i \neq k$
   (this requires that each minutia in TP_{cand} is associated with at most one minutia in TP_{ref})

Note that, in general, $P(i) = j$ does not necessarily mean that minutiae m'_j and m_i match in the sense of Equations (1) and (2), but only that they are the most likely pair under the current transformation.

Expression (3) requires that the number of minutiae mates be maximized, independently of how strict these mates are; in other words, if two minutiae comply with Equations (1) and (2), then their contribution to expression (3) is made independently of their spatial distance and of their direction difference.

Solving the minutiae matching problem (expression (3)) is ideal case when the correct alignment $(\Delta x, \Delta y, \delta)$ is known; in fact, the pairing (*i.e.* the function P) can be determined by setting for each $i = 1, \cdots, n$:

- $P(i) = j$ if $m''_j = \mathrm{map}_{\Delta x, \Delta y, \delta}(m'_j)$ is closest to m_i among the transformed candidate minutiae.
$$\left\{ m''_k = \mathrm{map}_{\Delta x, \Delta y, \delta}(m'_k) \mid k = 1, \cdots, n', \quad \zeta(m''_k, m_i) = 1 \right\}$$
- $P(i) = \perp$ if $\forall k = 1, \cdots, n', \quad \zeta(\mathrm{map}_{\Delta x, \Delta y, \delta}(m'_k), m_i) = 0$

To comply with constraint 4 above, each minutia m''_j already mated has to be marked, to avoid mating it twice or more. Figure 16.2 shows an example of minutiae pairing given a fingerprint alignment.

To achieve the optimum pairing (according to Equation (3)), a slightly more complicated scheme should be adopted: in fact, in the case when a minutia of TP_{cand} falls within the tolerance hypersphere of more than one minutia of TPref, the optimum assignment is that which maximizes the number of mates (refer to Figure 16.3 for a simple example).

The maximization in (3) can be easily solved if the function P (minutiae correspondence) is known; in this case, the unknown alignment $(\Delta x, \Delta y, \delta)$ can be determined in the least square sense. Unfortunately, in practice, neither the alignment parameters nor the correspondence function P are known and therefore, solving the matching problem is very hard. A brute force approach, that is, evaluating all the possible solutions (correspondences and alignments) is prohibitive as the number of possible solutions is exponential in the number of minutiae (the function P is more than a permutation due to the possible $\perp$ values). Hence heuristics are used.

In figure 16.2 minutiae of TP_{cand} mapped into TP_{ref} coordinates for a given alignment. Minutiae of TP_{ref} are denoted by $\odot$, whereas TP_{cand} minutiae are denoted by $\times$. Note that TP_{cand} minutiae are referred to as m'', because what is shown in the figure is their mapping into TP_{ref} coordinates. Pairing is performed according to the minimum distance. The dashed circles indicate the maximum spatial distance. The gray circles denote successfully mated minutiae; minutia m_1 of TP_{ref} and minutia m''_3 of TP_{cand} have no mates, minutiae m_3 and m''_6 cannot be mated due to their large direction difference.

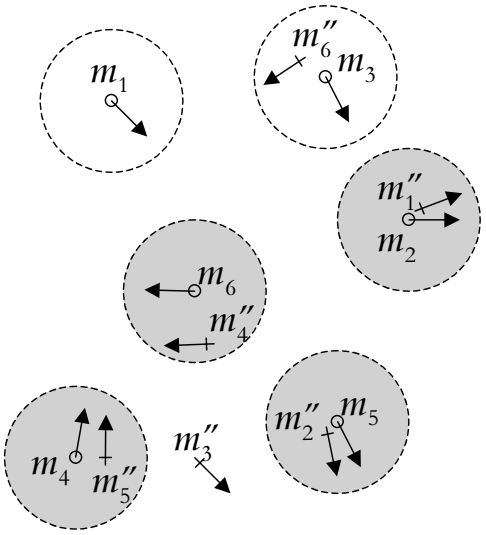


Figure 16.2: Mating

In figure 16.3, if m_1 were mated with m_2'' (the closest minutia), m_2 would remain unmated; however, pairing m_1 with m_1'', allows m_2 to be mated with m_2'', thus maximizing Equation (3).

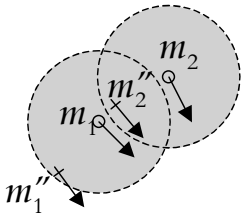


Figure 16.3: Mating with a Second-Closest

Algorithmic Complexity

This minutiae matching problem is part of graph theory in the mathematics, where a bipartite graph is a graph whose vertices can be divided into two disjoint sets U and V, such that every edge connects a vertex in U to one in V; that is, U and V are independent sets. The maximum bipartite matching in graph theory is known to be a NP-Hard problem in computational complexity theory, i.e. at least as hard as the hardest problems in NP (Non-deterministic Polynomial-time). Thus, classical fingerprint matching algorithms reduce this problem to only approximate the distance between U and V in regards of an adjustable threshold.

The following section provides a novel solution to this overall matching problem. The solution, called *Externalized Fingerprint Matching*, allows to implement the matching module $Match()$ even in simple (microprocessor-less) memory cards. This is particularly important for cost-effectively addressing very large markets (e.g. China, which has 1.3 billion inhabitants) and for deploying disposable biometric IDs such as visas, hotel room keys or visitor/subcontractor badges.

16.3 Externalizing the Fingerprint Matching

The new idea consists in adding *false minutiae* to TP_{ref} and *reversing the burden of proof* to have the *card challenge the reader* to find out, based on the acquisition coming from the scanner, which minutiae are genuine:

Figure 16.4: Fingerprint Scrambling with False Minutiae on Fingerprint Image

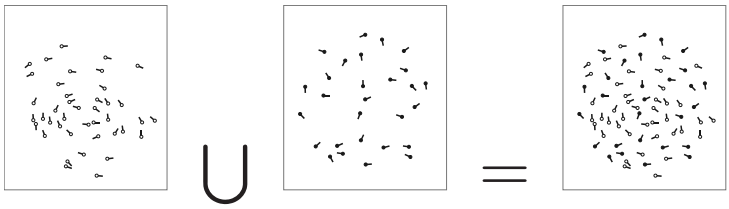


Figure 16.5: Fingerprint Scrambling with False Minutiae

In the simplified minutiae scrambling model, we let n be the number of minutiae in TP_{ref} and $N > n$ be the total number of minutiae in the resulting template t, that is, true and false minutiae. Again, we let d be a security parameter whose choice is discussed below. d is the minimum number of differences desired between u and u', and the maximum number of mismatches of minutiae for a biometric match.

Definition: A Biometric Scrambling Algorithm is an algorithm taking as input a set TP_{ref} of n minutiae and outputting a template t (called the *obfuscated template*) and a randomly distributed N-bit string u of weight n, such that the i-th minutia in t belongs to TP_{ref} iff $u_i = 1$.

$$Scrambling(TP_{ref}) = (t, u)$$

Definition: An Obfuscated Biometric Matching Algorithm is an algorithm taking as input a set (TP_{cand} , t) and outputting a candidate string u' to be compared with the enrolled u.

$$ObfMatching(TP_{cand}, t) = u'$$

Adversary Models

The security model is to avoid an attacker being able to quicly recover the original u or generate a candidate u', denoted as $Adversary(t) = u'$, only knowing t. However, our protocol does not protect from an attacker replaying a previously matching u'.

Assumption: Given only t the success probability of an attacker in finding u' such that $H(u' \oplus u) \leq d$ is $\varepsilon_{\text{guess}} + \varepsilon$, where ε is a negligible function of the parameters (N, d), and

$$\varepsilon_{\text{guess}} = 2^{-N} \sum_{i=0}^{d} \binom{N}{i}$$

Security Game

1. Pick a human at random
2. Extract TP_{ref}
3. Run the Scrambling Algorithm: $Scrambling(TP_{ref}) = (t, u)$
4. Run $Adversary(t) = u'$
5. Win if $H(u' \oplus u) \leq d$

Security: Given any adversary, $Pr(win) \leq \varepsilon_{\text{guess}} + \varepsilon$

FAR Game

1. Pick two humans at random
2. Extract TP_{ref} and TP_{cand}
3. Run the Scrambling Algorithm: $Scrambling(TP_{ref}) = (t, u)$
4. Run the Obfuscated Biometric Matching Algorithm: $ObfMatching(TP_{cand}, t) = u'$
5. Win if $H(u' \oplus u) \leq d$

We define the FAR being the probability to win this game.
If we have security, the FAR is smaller than the attacker's success probability :

$$\text{FAR} \leq 2^{-N} \sum_{i=0}^{d} \binom{N}{i} + \varepsilon$$

Neglecting the term ε, the following table lists various $\{N, d\}$ choices and their corresponding FARs.

$-\log_{10}(\text{FAR})$	2	3	3	4
N	10	20	26	30
d	2	3	5	5

Although a security level from 10^{-2} to 10^{-5} may sound quite ridiculous to cryptographers, this is equivalent to classical values in usual biometric matching solutions and numeric PIN codes. Moreover, as for PIN codes, a try counter is a classically used countermeasure against exhaustive search. For PIN codes, the usual value of the counter is three, whereas in biometrics the initial counter value would depend on the FRR of the system and is often between five and ten.

FRR Game

1. Pick a human at random
2. Extract TP_{ref} and TP_{cand} (this TP_{cand} being a new instance of the same fingerprint than TP_{ref} , but captured in different conditions)
3. Run the Scrambling Algorithm: $Scrambling(TP_{ref}\ ) = (t,u)$
4. Run the Obfuscated Biometric Matching Algorithm: $ObfMatching(TP_{cand}\ ,t) = u'$
5. Win if $H(u' \oplus u) \geq d$

We define the FRR being the probability to win this game.

Advanced Adversary Strategy

One may submit a random vector u' of weight k for optimal choice of k:
The correct fingerprint is characterized by the reference vector u whose length is $n+m$ and whose Hamming weight is n. Since u is unknown to the attacker we assume that it has a random distribution over the vectors of weight n.
Assume that the Hamming weight of u', the vector submitted by the attacker, is equal to k, where k is an integer. Letting $w = u' \oplus u$ we have $w = w_1 \vee w_2$ where $w_1 = u \wedge \neg u'$ and $w_2 = \neg u \wedge u'$. Let $i = H(w_1)$.
We have $H(u') = k = H(u) + H(w_2) - H(w_1)$, which gives $H(w_2) = i + k - n$, whereby $H(w) = H(w_1) + H(w_2) = 2i + k - n$. Since $H(u') = k$, the number of possible choices for w_2 is $\binom{k}{i+k-n}$ and the number of possible choices for w_1 is $\binom{m-k+n}{i}$.
The number of possible u vectors for a given integer i is therefore:

$$R(n,m,k,i) = \binom{k}{i+k-n} \times \binom{m-k+n}{i}$$

Summing over all possible i, we obtain the probability over u, which we denote $P(n,m,k,d)$, that the attack succeeds with a candidate u' of weight k:

$$P(n,m,k,d) = \frac{\sum_{i=0}^{(d-k+n)/2} R(n,m,k,i)}{\binom{m+n}{n}}$$

If $k-n$ is negative, we obtain the probability:

$$P(n,m,k,d) = \frac{\sum_{i=0}^{(d+k-n)/2} R'(n,m,k,i)}{\binom{m+n}{n}}$$

where

$$R'(n,m,k) = \binom{k}{i} \times \binom{m-k+n}{i-k+n}$$

Eventually, the success probability of an adversary is the maximum probability, over all possible k, that a candidate u' is accepted:

$$\varepsilon_{\text{guess}} + \varepsilon = \max_{k=0}^{m+n} P(n,m,k,d)$$

We then define the advantage of the adversary as being $Adv \leq \varepsilon_{\text{guess}} + \varepsilon$.

Our Scrambling Algorithm

$Input = TPref$
$Output = \{t,u\}$

1. Get TP_{ref} and sets $t \leftarrow \{\}$.
2. Generate a random k-bit string $u = u_1, \ldots, u_k$.
3. For $i = 1$ to k:
   (a) if $u_i = 1$, add to the template t a random (and not already selected) minutia from TP_{ref} .
   (b) if $u_i = 0$, add to t a random minutia: pick x,y at random, respecting image bounds (i.e. pixel height and width, and inside the convex hull of the real minutiae set) until minutia location is not too close to other existing minutia upon bound on spatial distance and pick θ at random with bounds determined upon angle data of surrounding minutiae for local coherence.
4. Deliver the data $\{t,u\}$.

Hypothesis

1. Secure, our advanced adversary is the best adversary
2. FRR is small, the probability to match many false minutiae in t to many real minutiae in TP_{ref} being small

Under the secure hypothesis, the FAR is smaller than the attacker's success probability :

$$\text{FAR} \leq \max_{k=0}^{m+n} P(n,m,k,d)$$

Letting FAR $= 10^{-e}$, typical $\{n,m\}$ values for $e = 5$ and $d = 0$ would be:
$\{6,17\},\{7,14\},\{8,12\},\{9,11\},\{10,10\},\{11,9\}$.
Variations in d affect the FAR as shown in the graphics below (Figure 16.6 shows the FAR for $d = 4$ and Figure 16.7 shows the FAR for $m = n$ and different d values):

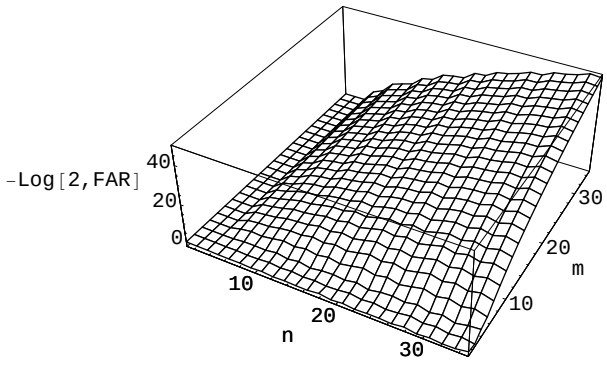


Figure 16.6: FAR for $d = 4$

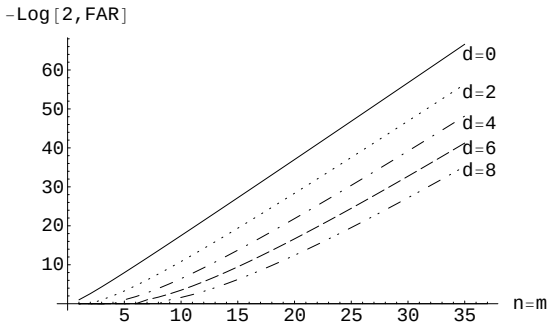


Figure 16.7: FAR for $m = n$ and different d values

Enrollment

The enrollment protocol is the following:
$Input = IM_{ref}$
$Output = A\ card$

1. Extract TP_{ref}
2. Run the Scrambling Algorithm: $Scrambling(TP_{ref}\ ) = (t,u)$
3. Sign the data $\{t,u,d\}$. Let σ be the signature of $\{t,u,d\}$
4. Deliver an identity card containing $\{t,u,d,\sigma\}$
   The card allows the free reading of t and d

Authentication

The authentication protocol is the following:
$Input = \{IM_{cand}\ ,t\}$
$Output = OK/NOK\ decision$

1. Receive from the scanner a fingerprint candidate image IM_{cand} and extract TP_{cand}
2. Read t from the card
3. Run the Obfuscated Biometric Matching Algorithm: $ObfMatching(TP_{cand}\ ,t) = u'$
4. Send u' to the card
5. The card computes $w = u \oplus u'$. If the Hamming weight of w (denoted $H(w)$) is smaller than d, the card outputs σ and u. At this step the card considers that the presented finger is legitimate
6. The terminal verifies σ with respect to the issuer's public-key, and if σ is correct and $H(u \oplus u') \leq d$ then the terminal considers that the scanned finger and the card match each other and are approved by the issuer

16.4 Our Implementation

The implementation may actually be seen as a Challenge-Response protocol using biometric information: the card sends a challenge to the terminal (the obfuscated template, the challenge being "find out chaff minutiae"). With a fresh capture of the (correct) candidate fingerprint, the terminal is able to compare the candidate minutiae set with the obfuscated one to find matching minutiae, deducing all other minutiae as being potentially chaff points, and send the response ("here are the real points, here are the fake points"). This proves the terminal captured the correct fingerprint of the authorized user. It is interesting to note that depending on the application (e.g. where the biometric authentication unlocks a functionality internally in the smart card) the case may be that no information is sent back to the terminal, limiting potential replay attacks.

The protocol was implemented as a Javacard applet on a Gemplus GemXpresso Pro smart card using Ikendi Software AG's minutiae extraction engine. For the reader terminal emulation, the demonstrator was using a Pentium III at 500 MHz and a Gemplus GemPC Touch 430 smart card reader with an embedded fingerprint sensor (silicon sensor using capacitive technology), shown in figure 16.8.

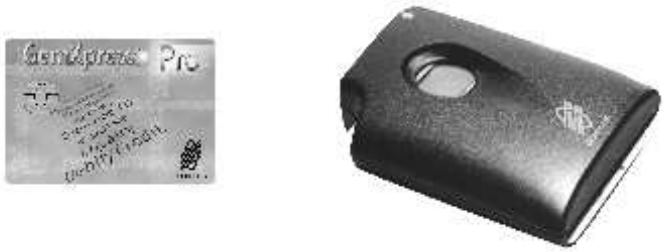


Figure 16.8: GemXpresso Pro and GemPC Touch 430

The entire fingerprint matching process takes less than a second. The applet's size is about 510 bytes and the card's processing time is 26 ms, that break-down as follows:

Protocol phase	**Duration**
The terminal asks permission to send u'	6 ms
The card prepares to receive u'	8 ms
The terminal sends u'	6 ms
The card compares u and u'	4 ms
The card returns true or false	2 ms

National Identity and Access Control to Facilities

In a typical national ID application, a law enforcement agent using a portable secret-less biometric reader must ascertain that a physically present individual is associated to a data string Q. In most cases, Q represents information such as the ID card number, the surname, given names, nationality, height, place of birth, date of birth, dates of issue and expiry, color of eyes, residence *etc.* In the sequel we assume that σ also signs Q.

Given that the portable reader is under the agent's total control (*i.e.* provides end-to-end security

from the capture unit to the decision taking and display module) the display of Q on the reader's screen provides the officer with a binding between the physically present individual and Q.

Note (as is the case with *all other* match-on-card protocols) that biometry alone cannot provide a binding between the ID (physical support) and the individual but only between Q (the information) and the individual. To provide *also* a binding between the ID and the individual the ID must be enriched with active digital signature or zero-knowledge capabilities.

Internet Login and On-Line Access

As is the case with passwords and other biometric protocols, one *cannot* require resistance against parties who witnessed a successful biometric identification session (or participated in it). However, we do require that a party who never witnessed a successful session will be unable to mimic the legitimate user and his card, even under the (very adversarial) assumption that the attacker managed to steal the user's ID card.

Given that the card will only reveal u when the interrogator proves to it that he knows already an extremely close approximation of u, the thief will not be able to retrieve TP_{ref} from the ID card [‡] and mimic the user's presence on the other side of the communication line.

As is the case with passwords and other biometric protocols, a remote user can always voluntarily 'delegate' (lend) his IM and σ to friends or colleagues. To prevent this and ascertain that the ID card is physically present at the other side of the line, the ID card must be enriched with active public-key capabilities. Note that even such a capability will never ascertain that the user did not voluntarily give the physical ID card plus IM to the friend or the colleague.

Access Control to Card Inner Data or Card Functions

In many settings, one wishes to bind the enabling of an *on-card* function to the user's presence. A typical example is an electronic purse where a debit function is activated only after successfully recognizing the user's IM. This provides an excellent protection against card theft and subsequent illegal debit.

Note that unless the capture unit is embedded in the card (such capture units are marketed by several suppliers today), a user can, again, voluntarily lend his IM to a friend (although in most cases debit operations are done in front of merchants). Other applications of access control to inner card data consist in accessing private files on a memory stick or medical data in health cards.

Conclusion

The above shows that although extremely economical (from an on-board resource consumption perspective), the protocol presented here provides equivalent functionalities to other match-on-card techniques in all typical use-cases.

[‡] Should the FAR be high (*e.g.* $\simeq 2^{-40}$), we recommend to protect the card against exhaustive search using a ratification counter. *e.g* lock the card after 20 successive unsuccessful fingerprint verifications.

16.5 Our Demonstrator

We have built a demonstrator out of our concept, described hereunder.
Figure 16.9 describes the enrollment procedure with the following steps:

1. capture the reference fingerprint and extract the minutiae set, denoted X here
2. obfuscate the minutiae set to build the obfuscated minutiae set, denoted Y here
3. store in the smart card the public Y along with its private BioEasy code, denoted S here. S will never leave the smart card after enrollment, apart for signature verification upon successful match.

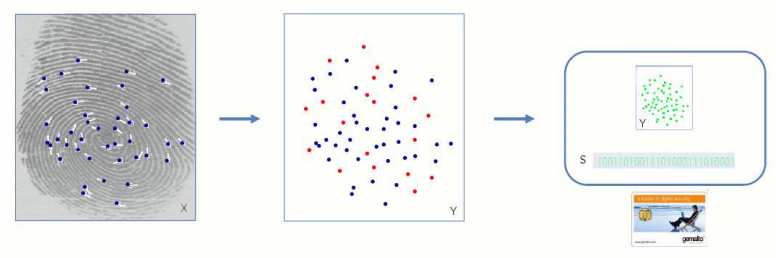


Figure 16.9: BioEasy Enrollment

Figure 16.10 shows the enrollment within the demo application. First image shows real extracted minutiae in green, second image shows added fake minutiae in red, third image shows the obfuscated minutiae set Y and the associated S is displayed at the bottom of the application.

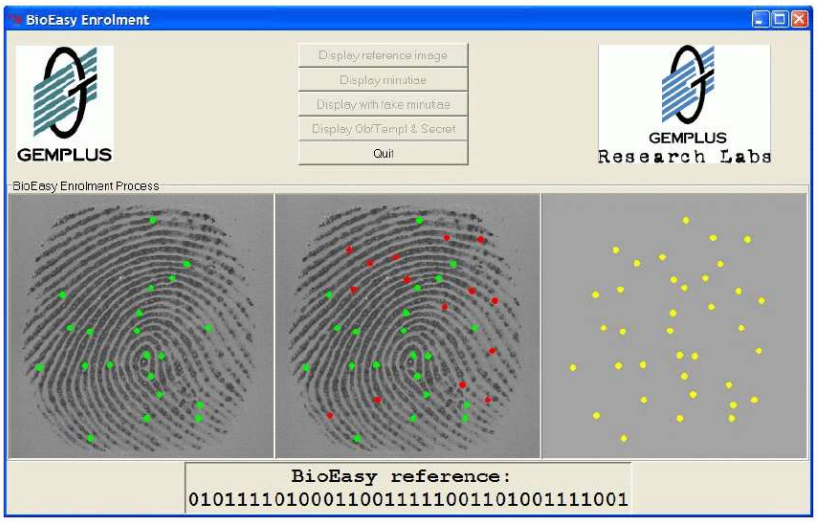


Figure 16.10: BioEasy Demo Application at Enrollment

Figure 16.11 describes the verification procedure with the following steps:

1. The terminal captures the candidate fingerprint and extracts the minutiae set, denoted X' here.
2. The terminal reads Y from the smart card. Even if we're facing a malicious terminal, Y is non-sensitive public data.
3. By the comparison of X' and Y the terminal is able to find out real and fake minutiae, and thus build a candidate binary string, denoted U here.
4. The terminal sends U to the smart card.
5. The smart card internally compares U with its private S and take a binary decision based on internal thresholds.
6. The smart card unlocks the internal application on positive verification - the terminal doesn't need the decision status.

or 6'. If the application needs to output the result, the smart card sends the binary decision OK/NOK.

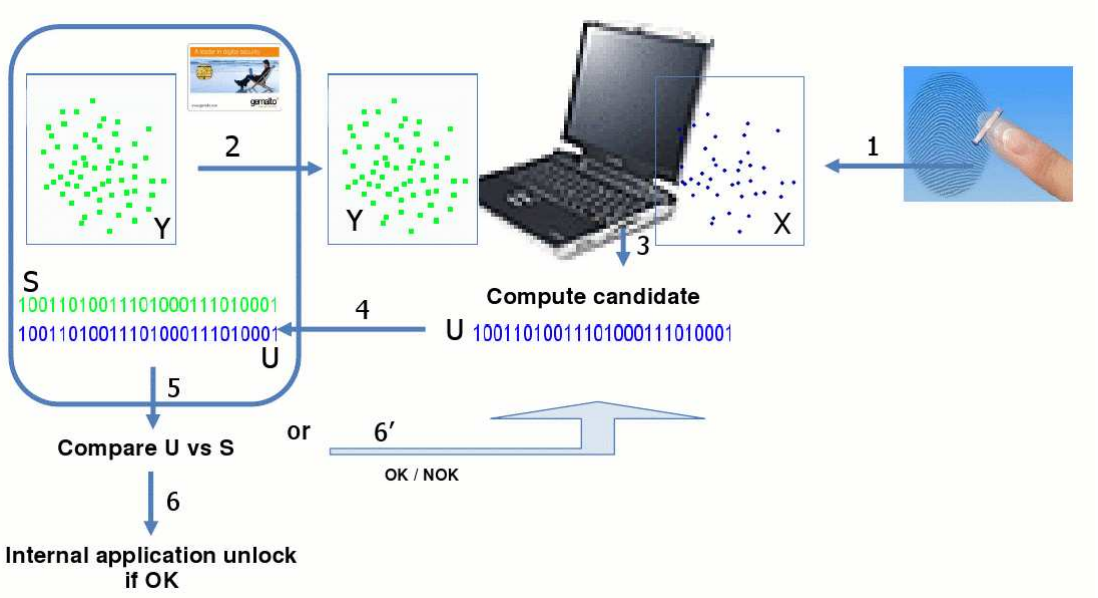


Figure 16.11: BioEasy Verification

Figure 16.12 shows a positive verification within the demo application. First image shows real extracted minutiae from the candidate fingerprint in blue (X'), second image shows Y read from the smart card in yellow, third image shows the matching results (green points are matching, blue points are non-matching ones from X', red points are non-matching ones from Y, thus may be considered as the fakes) and the associated candidate bitstring U is displayed at the bottom of the application.

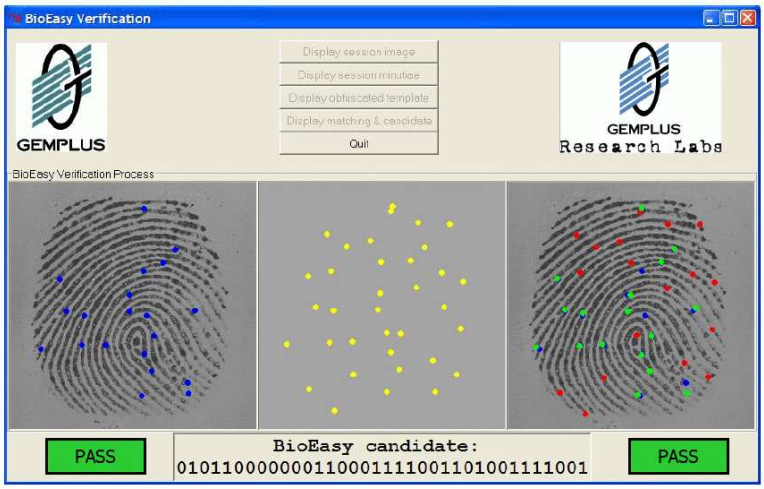


Figure 16.12: BioEasy Demo Application at Verification - Pass

Figure 16.13 shows a negative verification within the demo application. Same explanations as above.

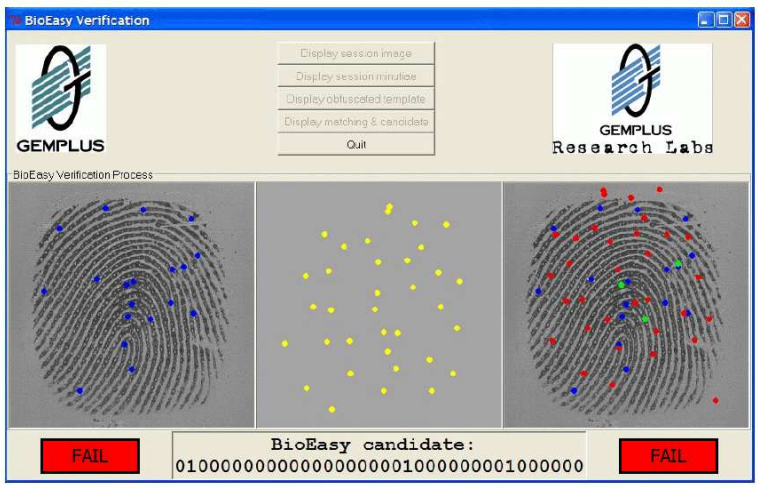


Figure 16.13: BioEasy Demo Application at Verification - Fail

16.6 Conclusion

Our work has been published in [8]. In [34], Dodis *et al* referred our work as interesting and complementary to theirs. The implementation of this approach has been rewarded as the most innovative software for smart cards during the annual Cartes'04 event in Paris and patented since early 2003. Very recently, with the hype of privacy-enhanced technologies for biometrics, the patent is constantly attacked but is still maintained, and would potentially give royalties in the long run.

General Conclusion & Future Research

General Conclusion

This thesis is a broad overview about fingerprint authentication, covering security and technical issues with sensors, computing platforms, algorithms, protocols and especially highlights the interaction with other information technologies. Because of our particular experience in the area, we focused our biometric research activities on smart cards and cryptography, beginning with hardware prospectives and naturally continuing with software aspects as introduced in Part I of our present work.

Privacy issues related to the public aspect of fingerprints and other biometric modalities in general clearly show the compulsory need to protect data in terms of integrity and authentication. In our opinion, confidentiality is less sensitive. The main threat related to the public aspect of fingerprint is the current lack of aliveness detection within fingerprint sensors. The maturity of certification standards in cryptography and smart cards fade onto the area of biometrics and dedicated certification procedures are currently under development. As depicted in Part II of our present work, security insurances will be mandatory in the coming years for large scale biometric systems.

The ubiquitous deployment of biometric authentication in embedded electronic devices paves the way to small and efficient biometric subsystems such as sensing, extraction, one-way transformation, matching. It also paves the way to the development of new secure protocols for data exchange between a wide range of mobile devices. The importance of the constant improvement of algorithms and protocols has been proved by the establishing of ambitious monitoring programs such as MINEX and SBMOC, as depicted in Part III and Part IV.

Biometric hardwares, algorithms and protocols may be considered mature enough to widespread in numerous applications. But, under certain conditions, the sensitivity of data and the security of current tools is still questionable on a cryptanalysis point of view. Ongoing programs such as TURBINE explore this brand new area, still in its infancy and open to a lot of improvements and fun for researchers. These stakes are introduced in Part V of our present work. Once again this proves the perfect complementarity between smart cards, cryptography and biometrics in opposition to mercantile or denigrating messages such as “forget your keys, cards and passwords: your fingerprint is THE key”, “Biometrics are useless to security” and so on.

Evaluation of claimed crypto-biometric solutions is not obvious. Claimed solutions come in the form of black boxes for an evaluation and IP issues result in the lack of technical information. An evaluator would need these detailled information to understand the approach, evaluate possible issues and reach a first level of confidence in the system.

Future Research

Covering a wide range of security issues with biometric data as user identifier, our dissertation shows the relative security level as of today and the opportunity for improvements in the near future. Our work is at the intersection of multiple disciplines such as embedded electronics (both hardware and software), biometrics, cryptography and security in general. The interaction with living persons enlarges the scope to biology and interesting future areas of research such as BCI (Brain-Computer Interfaces) and BioChips.

Brain-Computer Interfaces are usually developed for peoples with disabilities to enhance their control of prosthesis, vehicles or computers. A well-known cryptographer, Paul Van Oorschot, and his colleagues are proprosing "Pass-thoughts: Authenticating With Our Minds": just thinking about our password, we will be authenticated upon our secret and the way our brain is thinking about it! The ultimate melding of *what-you-know* and *what-you-are*, bringing the *secrecy* feature to biometrics [116].

Biochips are more and more used in medicine for quick and cost-effective diagnostics of glycemy, cholesterol, pregnancy at early stage, or any chemical/biological component presence detection in a living sample, with so-called *Lab-on-Chip*. This technology, at the intersection of biology and electronics, could be added to any silicon-based fingerprint sensor to detect, for instance, the presence of amino-acids from a finger to check aliveness.

We would like, in the near future, to enhance the cryptography-biometrics interaction part of this dissertation and strongly believe this will be a very sensitive subject to achieve a big step forward in security and reach conscientious user's expectations in terms of privacy protection. In our opinion, both P.Tuyls *et al* and T.Boult *et al* approaches are promising, these solutions being both practical and theoretically proven at the same time.

Five years ago, our proposal for a challenge-response (strange) vision of the fingerprint matching and its interaction with smart cards was not well understood and accepted by the biometrics community. Fortunately, biometricians finally understood the need to enlarge their scope to the global security system, whereas they had been historically accustomed to consider only the efficiency of their image processing and pattern recognition domain.

Finally, and despite many years of experience in this domain, we are still confident about the usability of biometrics as a good security element in combination with other security tools, and biometrics could even be privacy-protective if correctly handled.

Bibliography

[1] ANSI INCITS 378. *Information technology - Finger Minutiae Format for Data Interchange*, 2004.

[2] A. Antonelli, Raffaele Cappelli, Dario Maio, and Davide Maltoni. A new approach to fake finger detection based on skin distortion. In Zhang and Jain [136], pages 221–228.

[3] Fingerprint Duplication Archive. How to duplicate your fingerprints. http://www.journalofaestheticsandprotest.org/4/fingerprint/fingerprint.pdf [last access on 2008/06/15].

[4] ASFIP. Attack standardization for fingerprint system certification. https://tokyo.emse.fr/trac/asfip/ [last access on 2009/09/10].

[5] Julian Ashbourn. *Practical Biometrics - From Aspiration to Implementation*. Springer, 2004.

[6] Multiples authors. Biometric evaluation methodology supplement. Technical report, Common Criteria, 2002.

[7] Denis Baldisserra, Annalisa Franco, Dario Maio, and Davide Maltoni. Fake fingerprint detection by odor analysis. In Zhang and Jain [136], pages 265–272.

[8] Claude Barral, Jean-Sébastien Coron, and David Naccache. Externalized fingerprint matching. In Zhang and Jain [135], pages 309–315.

[9] Claude Barral and Assia Tria. Fake fingers in fingerprint recognition: Glycerin supersedes gelatin. In Cortier and *et al.* [28], pages 57–69.

[10] Claude Barral and Serge Vaudenay. A protection scheme for moc-enabled smart cards. *IEEE - Biometric Consortium Conference, 2006 Biometrics Symposium: Special Session on Research at the*, 2006.

[11] G. Bebis, T. Deaconu, and M. Georgiopoulos. Fingerprint identification using delaunay triangulation. In *Information Intelligence and Systems, 1999. Proceedings. 1999 International Conference on*, pages 452–459, 1999.

[12] Bir Bhanu and Xuejun Tan. Fingerprint indexing based on novel features of minutiae triplets. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(5):616–622, 2003.

[13] Wieslaw Bicz, Zbigniew Gumienny, and Mieczyslaw Pluta. Ultrasonic sensor for fingerprints recognition. volume 2634, pages 104–111. SPIE, 1995.

[14] Jean-Daniel Boissonnat, Olivier Devillers, and Samuel Hornus. Incremental construction of the delaunay triangulation and the delaunay graph in medium dimension. In *Symposium on Computational Geometry*, pages 208–216, 2009.

[15] Dan Boneh and Matthew K. Franklin. Identity-based encryption from the weil pairing. In Joe Kilian, editor, *CRYPTO*, volume 2139 of *Lecture Notes in Computer Science*, pages 213–229. Springer, 2001.

[16] Terrance E. Boult, Walter J. Scheirer, and Robert Woodworth. Revocable fingerprint biotokens: Accuracy and security analysis. In *CVPR*. IEEE Computer Society, 2007.

[17] Jonathan N. Bradley and Christopher M. Brislawn. The wavelet scalar quantization compression standard for digital fingerprint images. In *ISCAS'94*.

[18] Julien Bringer, Hervé Chabanne, Malika Izabachène, David Pointcheval, Qiang Tang, and Sébastien Zimmer. An application of the goldwasser-micali cryptosystem to biometric authentication. In Josef Pieprzyk, Hossein Ghodosi, and Ed Dawson, editors, *ACISP*, volume 4586 of *Lecture Notes in Computer Science*, pages 96–106. Springer, 2007.

[19] Andrew Burnett, Adam Duffy, Tom Dowling, and Nui Maynooth. A biometric identity based signature scheme. In *Proceedings of the Applied Cryptography and Network Security Conference*, 2004.

[20] R. Cappelli, D. Maio, A. Lumini, and D. Maltoni. Fingerprint image reconstruction from standard templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9):1489–1503, 2007.

[21] Raffaele Cappelli, Alessandra Lumini, Dario Maio, and Davide Maltoni. Can fingerprints be reconstructed from iso templates? In *ICARCV*, pages 1–6. IEEE, 2006.

[22] B. Charlot, F. Parrain, N. Galy, S. Basrour, and B. Courtois. A sweeping mode integrated fingerprint sensor with 256 tactile microbeams. *Journal-of-Microelectromechanical-Systems*, 13(4):636–44, 2004.

[23] Heeseung Choi, Raechoong Kang, Kyungtaek Choi, and Jaihie Kim. Aliveness detection of fingerprints using multiple static features. *World Academy of Science, Engineerind and Technology*, 28:157–162, 2007.

[24] Yongwha Chung, Daesung Moon, Sungju Lee, Seunghwan Jung, Taehae Kim, and Dosung Ahn. Automatic alignment of fingerprint features for fuzzy fingerprint vault. In Dengguo Feng, Dongdai Lin, and Moti Yung, editors, *LNCS - Information Security and Cryptology*, volume 3822 of *Lecture Notes in Computer Science*, pages 358–369. Springer, 2005.

[25] T. Charles Clancy, Negar Kiyavash, and Dennis J. Lin. Secure smartcard-based fingerprint authentication. In *WBMA '03: Proceedings of the 2003 ACM SIGMM workshop on Biometrics methods and applications*, pages 45–52, New York, NY, USA, 2003. ACM Press.

[26] Chaos Computer Club. How to fake fingerprints? http://www.ccc.de/biometrie/fingerabdruck_kopieren.xml?language=en [last access on 2008/06/15].

[27] D. Cooper, H. Dang, P. Lee, W. MacGregor, and K. Mehta. Secure biometric match-on-card feasability report - nistir 7452. Technical report, National Institute of Standards and Technology, 2007.

[28] Véronique Cortier and *et al.*, editors. volume 5458 of *Lecture Notes in Computer Science*. Springer, 2009.

[29] G. S. Cox and G. De Jager. A survey of point pattern matching techniques and a new approach to point pattern recognition. In *Proc. South African Symposium on Communications and Signal Processing*, pages 243–248, 1993.

[30] John Daugman. How iris recognition works. *IEEE Trans. Circuits Syst. Video Techn.*, 14(1):21–30, 2004.

[31] George I. Davida, Yair Frankel, Brian J. Matt, and RenÃ© Peralta. On the relation of error correction and cryptography to an off line biometric based identification scheme. In *Proceedings of the Workshop on Coding and Cryptography, Paris, France*, pages 129–138, 1999.

[32] Stéphanie Delaune and Florent Jacquemard. A theory of dictionary attacks and its complexity. In *CSFW*, pages 2–15. IEEE Computer Society, 2004.

[33] W. Diffie and M.E. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22:644–654, 1976.

[34] Yevgeniy Dodis, Leonid Reyzin, and Adam Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In Christian Cachin and Jan Camenisch, editors, *EUROCRYPT*, volume 3027 of *Lecture Notes in Computer Science*, pages 523–540. Springer, 2004.

[35] Josep Domingo-Ferrer, David Chan, and Anthony Watson, editors. *Smart Card Research and Advanced Applications, Proceedings of the Fourth Working Conference on Smart Card Research and Advanced Applications, CARDIS 2000, September 20-22, 2000, Bristol, UK*, volume 180 of *IFIP Conference Proceedings*. Kluwer, 2000.

[36] M. Drahansky. Experiments with skin resistance and temperature for liveness detection. In *Intelligent Information Hiding and Multimedia Signal Processing, 2008. IIHMSP '08 International Conference on*, pages 1075–1079, Aug. 2008.

[37] M. Drahansky, R. Notzel, and W. Funk. Liveness detection based on fine movements of the fingertip surface. In *Information Assurance Workshop, 2006 IEEE*, pages 42–47, June 2006.

[38] John W. Eaton, David Bateman, and Soren Hauberg. *GNU Octave Manual, Version 3*. Network Theory Ltd, 2008.

[39] David C. Feldmeier and Philip R. Karn. Unix password security - ten years later. In Gilles Brassard, editor, *CRYPTO*, volume 435 of *Lecture Notes in Computer Science*, pages 44–63. Springer, 1989.

[40] Vincent Fleury and Tomoko Watanabe. Morphogenesis of fingers and branched organs, how collagen and fibroblasts break the symmetry of growing biological tissue. *Les Comptes Rendus de l'Academie des Sciences, Paris*, 325:571–583, 2002.

[41] Vincent Fleury and Tomoko Watanabe. About the equilibrium shape of fibred structures and biological shapes. *Les Comptes Rendus de l'Academie des Sciences, Paris*, 327:663–677, 2004.

[42] Steven Fortune. A sweepline algorithm for voronoi diagrams. In *Symposium on Computational Geometry*, pages 313–322, 1986.

[43] Futronic. Futronic's live finger detection(lfd) technology. http://www.futronic-tech.com/download/LFD_fact_sheet.pdf [last access on 2009/09/10].

[44] FVC2006. http://bias.csr.unibo.it/fvc2006/ [accessed 2009/11/14].

[45] Robert S. Germain, Andrea Califano, and Scott Colville. Fingerprint matching using transformation parameter clustering. *IEEE Comput. Sci. Eng.*, 4(4):42–49, 1997.

[46] Shafi Goldwasser and Silvio Micali. Probabilistic encryption. *J. Comput. Syst. Sci.*

[47] P. Grother and W. Salamon. Performance of fingerprint match-on-card algorithms evaluation plan - nistir 7485. Technical report, National Institute of Standards and Technology, 2007.

[48] P. Grother and *et al.* Performance and interoperabili 378 fingerprint template - nistir 7296. Technical report, National Institute of Standards and Technology, 2006.

[49] P. Grother and *et al.* Performance of fingerprint match-on-card algorithms - nistir 7477. Technical report, National Institute of Standards and Technology, 2008, 2009.

[50] Gaël Hachez, François Koeune, and Jean-Jacques Quisquater. Biometrics, Access Control, Smart Cards: A not so simple combination. In Domingo-Ferrer et al. [35], pages 273–288.

[51] David A. Hall, Jason Ptacek, and Michael Snyder. Protein microarray technology. *Mechanisms of Ageing and Development*, 128(1):161 – 167, 2007.

[52] ICAO. Annex 1 - Use of Contactless Integrated Circuits. Technical report, May 2004. Available at http://www.icao.int/mrtd/download/documents/Annexs.pdf.

[53] ICAO. Biometrics deployment for Machine Readable Travel Documents. Technical report, May 2004. Available at http://www.icao.int/mrtd/download/documents.

[54] ICAO. PKI for Machine Readable Travel Documents offering ICC read-only access. Technical report, Oct. 2004. Available at http://www.icao.int/mrtd/download/documents/TR-PKIy

[55] Innovatrics. Id_demo. http://www.innovatrics.com/products/iddemo/ [last access on 2008/06/15].

[56] International Business Machines Corp. The consideration of data security in a computer environment. *IBM, Data Processing Division*, 1968.

[57] ISO/IEC 19794-2. *Information technology - Biometric data interchange formats - Part 2: Finger minutiae data*, 2005.

[58] ISO/IEC 19794-3. *Information technology - Biometric data interchange formats - Part 3: Finger pattern spectral data*, 2006.

[59] ISO/IEC 19794-4. *Information technology - Biometric data interchange formats - Part 4: Finger image data*, 2005.

[60] ISO/IEC 19794-8. *Information technology - Biometric data interchange formats - Part 4: Finger pattern skeletal data*, 2006.

[61] ISO/IEC 24745. *Information Technology - Security Techniques - Biometric Template Protection (Committee Draft)*, 2009.

[62] Anil Jain, Ruud Bolle, and Sharath Pankanti. *Biometrics - Personal Identification in Networked Society*. Kluwer Academic Publishers, 1999.

[63] Anil K. Jain, Yi Chen, and Meltem Demirkus. Pores and ridges: High-resolution fingerprint matching using level 3 features. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(1):15–27, 2007.

[64] Jia Jia, Lianhong Cai, Kaifu Zhang, and Dawei Chen. A new approach to fake finger detection based on skin elasticity analysis. In Lee and Li [68], pages 309–318.

[65] A. Juels and M. Sudan. A fuzzy vault scheme, 2002.

[66] Alper Kanak and Ibrahim Sogukpinar. Fingerprint hardening with randomly selected chaff minutiae. In Walter G. Kropatsch, Martin Kampel, and Allan Hanbury, editors, *CAIP*, volume 4673 of *Lecture Notes in Computer Science*, pages 383–390. Springer, 2007.

[67] Pierre-Olivier Ladoux, Christophe Rosenberger, and Bernadette Dorizzi. Palm vein verification system based on sift matching. In *ICB'09*.

[68] Seong-Whan Lee and Stan Z. Li, editors. *Advances in Biometrics, International Conference, ICB 2007, Seoul, Korea, August 27-29, 2007, Proceedings*, volume 4642 of *Lecture Notes in Computer Science*. Springer, 2007.

[69] Stan Z. Li and Anil K. Jain. *Handbook of Face Recognition*. Springer, 2005.

[70] Stan Z. (Ed.) Li. *Encyclopedia of Biometrics*. Springer, 2009.

[71] Jean-François Mainguet.
http://pagesperso-orange.fr/fingerchip/ [last access on 2009/04/15].

[72] Jean-François Mainguet, Marc Pégulu, and John B. Harris. Fingerprint recognition based on silicon chips. *Future Generation Comp. Syst.*, 16(4):403–415, 2000.

[73] Jean-Francois Mainguet, Wei Gong, and Anne Wang. Reducing silicon fingerprint sensor area. In *ICBA'04*.

[74] Davide Maltoni, Dario Maio, Anil Jain, and Salil Prabhakar. *Handbook of Fingerprint Recognition*. Springer, 2003.

[75] T. Mansfield and J. Wayman. Best practices in testing and reporting performance of biometric devices, npl report cmsc 1402. Technical report, National Physical Laboratory, 2002.

[76] Tsutomu Matsumoto. Gummy and conductive silicone rubber fingers. In Zheng [137], pages 574–576.

[77] Tsutomu Matsumoto, Hiroyuki Matsumoto, Koji Yamada, and Satoshi Hoshino. Impact of artificial gummy fingers on fingerprint systems,. *Optical Security and Counterfeit Deterrence Techniques IV, proceedings of SPIE*, vol. 4677:pages 275–289, 2002.

[78] Keith Mayes and Konstantinos Markantonakis. *Smart Cards, Tokens, Security and Applications*. Springer, 2008.

[79] A. Menezes, P. Van Oorschot, and S. Vanstone. *Handbook of Applied Cryptography*. The CRC Press series on discrete mathematics and its applications. CRC-Press, 1997.

[80] Y.S. Moon, J.S. Chen, K.C. Chan, K. So, and K.C. Woo. Wavelet based fingerprint liveness detection. *Electronics Letters*, 41(20):1112–1113, Sept. 2005.

[81] David Naccache and Jacques Stern. A new public key cryptosystem based on higher residues. In *ACM Conference on Computer and Communications Security*, pages 59–66, 1998.

[82] I. Nakanishi, Y. Yorikane, Y. Itoh, and Y. Fukui. Biometric identity verification using intra-body propagation signal. In *Biometrics Symposium, 2007*, pages 1–6, Sept. 2007.

[83] NEUROtechnology. Verifinger sdk. http://www.neurotechnology.com/vf_sdk.html [last access on 2008/06/15].

[84] Philippe Oechslin. Making a faster cryptanalytic time-memory trade-off. In Dan Boneh, editor, *CRYPTO*, volume 2729 of *Lecture Notes in Computer Science*, pages 617–630. Springer, 2003.

[85] Pascal Paillier. Public-key cryptosystems based on composite degree residuosity classes. In *EUROCRYPT*, pages 223–238, 1999.

[86] Sujan T. V. Parthasaradhi, Reza Derakhshani, Larry A. Hornak, and Stephanie A. C. Schuckers. Time-series detection of perspiration as a liveness test in fingerprint devices. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, 35(3):335–343, 2005.

[87] Giuseppe Parziale and Albert Niel. A fingerprint matching using minutiae triangulation. In Zhang and Jain [135], pages 241–248.

[88] Salil Prabhakar, Sharath Pankanti, and Anil K. Jain. Biometric recognition: Security and privacy concerns. *IEEE Security & Privacy*, 1(2):33–42, 2003.

[89] Denis Praca and Claude Barral. From smart cards to smart objects: the road to new smart technologies. *Computer Networks*, 36(4):381–389, 2001.

[90] Wolfgang Rankl and Wolfgang Effing. *Smart Card Handbook*. John Wiley & Sons, Ltd, 2003.

[91] N. Ratha, J. Connell, R.M. Bolle, and S. Chikkerur. Cancelable biometrics: A case study in fingerprints. In *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, volume 4, pages 370–373, 2006.

[92] N. K. Ratha, J. H. Connell, and R. M. Bolle. Enhancing security and privacy in biometrics-based authentication systems. volume 40, pages 614–634, Riverton, NJ, USA, 2001. IBM Corp.

[93] Nalini K. Ratha and Venu (Eds.) Govindaraju. *Advances in Biometrics*. Springer, 2008.

[94] N.K. Ratha, S. Chikkerur, J.H. Connell, and R.M. Bolle. Generating cancelable fingerprint templates. volume 29, pages 561–572, 2007.

[95] P.V. Reddy, A. Kumar, S.M.K. Rahman, and T.S. Mundra. A new method for fingerprint antispoofing using pulse oxiometry. In *Biometrics: Theory, Applications, and Systems, 2007. BTAS 2007. First IEEE International Conference on*, pages 1–6, Sept. 2007.

[96] The Register. Get your german interior minister's fingerprint here. http://www.theregister.co.uk/2008/03/30/german_interior_minister_fingerprint_appropriated/ [last access on 2008/06/15].

[97] M. Reichmanis, A.A. Marino, and R.O. Becker. Laplace plane analysis of skin impedance: a preliminary investigation. *Journal of Electrochemical Society*, 125:1765–1768, 1978.

[98] R.L. Rivest, A. Shamir, and L.M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21:120–126, 1978.

[99] N. Rokbani and A. Alimi. Fingerprint identification using minutiae constellation matching. In *IADIS Virtual Multi Conference on Computer Science and Information Systems*, pages 157–162, 2005.

[100] Arun Ross, Jidnya Shah, and Anil K. Jain. Towards reconstructing fingerprints from minutiae points. *SPIE 2nd Conf. on Biometrics Tech. for Human Ident.*, 5779:60–80, 2005.

[101] Arun Ross, Jidnya Shah, and Anil K. Jain. From template to image: Reconstructing fingerprints from minutiae points. *IEEE Trans. Pattern Anal. Mach. Intell.*, 29(4):544–560, 2007.

[102] Robert K. Rowe. Biometrics based on multispectral skin texture. In Lee and Li [68], pages 1144–1153.

[103] Amit Sahai and Brent Waters. Fuzzy identity-based encryption. In Ronald Cramer, editor, *EUROCRYPT*, volume 3494 of *Lecture Notes in Computer Science*, pages 457–473. Springer, 2005.

[104] Marie Sandström. Liveness detection in fingerprint recognition systems. Master thesis, Linköping university, Sweden, 2004.

[105] Walter J. Scheirer and Terrance E. Boult. Bipartite biotokens: Definition, implementation, and analysis. In Massimo Tistarelli and Mark S. Nixon, editors, *ICB*, volume 5558 of *Lecture Notes in Computer Science*, pages 775–785. Springer, 2009.

[106] W.J. Scheirer and T.E. Boult. Cracking fuzzy vaults and biometric encryption. In *Biometrics Symposium, 2007*, Sept. 2007.

[107] Berry Schoenmakers and Pim Tuyls. Efficient binary conversion for paillier encrypted values. In Serge Vaudenay, editor, *EUROCRYPT*, volume 4004 of *Lecture Notes in Computer Science*, pages 522–537. Springer, 2006.

[108] Stephanie Schuckers. Clarkson university engineer outwits high-tech fingerprint fraud. www.yubanet.com/cgi-bin/artman/exec/view.cgi/38/28878 [last access on 2008/06/15].

[109] M. M. Schulz, H. D. Wehner, W. Reichert, and M. Graw. Ninhydrin-dyed latent fingerprints as a dna source in a murder case. *Journal of Clinical Forensic Medicine*, 11(4):202 – 204, 2004.

[110] Adi Shamir. Identity-based cryptosystems and signature schemes. In *CRYPTO*, pages 47–53, 1984.

[111] G. S. Sodhi and J. Kaur. Fingermarks detection by eosin-blue dye. *Forensic Sci Int*, 115(1-2):69–71, Jan 2001.

[112] G. S. Sodhi and J. Kaur. Powder method for detecting latent fingerprints: a review. *Forensic Sci Int*, 120(3):172–176, Sep 2001.

[113] Colin Soutar. Biometric system security. http://www.bioscrypt.com/assets/security_soutar.pdf [last access on 2009/09/10].

[114] Michael Still. *The Definitive Guide to ImageMagick*. Apress, 2006.

[115] Douglas Stinson. *Cryptography - Theory and Practice - 2nd edition*. Chapman & Hall / CRC, 2002.

[116] Julie Thorpe, P.C. van Oorschot, and Anil Somayaji. Pass-thoughts: Authenticating with our minds. Cryptology ePrint Archive, Report 2005/121, 2005.

[117] Pim Tuyls, Anton H. M. Akkermans, Tom A. M. Kevenaar, Geert Jan Schrijen, Asker M. Bazen, and Raymond N. J. Veldhuis. Practical biometric authentication with template protection. In Takeo Kanade, Anil K. Jain, and Nalini K. Ratha, editors, *AVBPA*, volume 3546 of *Lecture Notes in Computer Science*, pages 436–446. Springer, 2005.

[118] Pim Tuyls and Jasper Goseling. Capacity and examples of template-protecting biometric authentication systems. In Davide Maltoni and Anil K. Jain, editors, *ECCV Workshop BioAW*, volume 3087 of *Lecture Notes in Computer Science*, pages 158–170. Springer, 2004.

[119] Pim Tuyls, Boris Skoric, and Tom (Eds.) Kevenaar. *Security with Noisy Data*. Springer, 2007.

[120] U. Uludag and Anil Jain. Securing fingerprint template: Fuzzy vault with helper data. In *Computer Vision and Pattern Recognition Workshop, 2006. CVPRW '06. Conference on*, pages 163–163, June 2006.

[121] Umut Uludag and Anil K. Jain. Attacks on biometric systems: a case study in fingerprints. In *Security, Steganography, and Watermarking of Multimedia Contents'04*.

[122] Umut Uludag, Sharath Pankanti, and Anil K. Jain. Fuzzy vault for fingerprints. In *AVBPA'05*.

[123] European Union. European data protection directive. http://en.wikipedia.org/wiki/Data_Protection_Directive [last access on 2009/12/01].

[124] Ton van der Putte and Jeroen Keuning. Biometrical fingerprint recognition: Don't get your fingers burned. In Domingo-Ferrer et al. [35], pages 289–306.

[125] Serge Vaudenay. *A Classical Introduction to Cryptography: Applications for Communications Security*. Springer, 2005.

[126] Serge Vaudenay. Secure communications over insecure channels based on short authenticated strings. In Victor Shoup, editor, *CRYPTO*, volume 3621 of *Lecture Notes in Computer Science*, pages 309–326. Springer, 2005.

[127] Stephen P. Wargacki, Linda A. Lewis, and Mark D. Dadmun. Understanding the chemistry of the development of latent fingerprints by superglue fuming. *J Forensic Sci*, 52(5):1057–1062, Sep 2007.

[128] James Wayman, Anil Jain, Davide Maltoni, and Dario Maio. *Biometric Systems - Technology, Design and Performance Evaluation*. Springer, 2005.

[129] J.H. Wegstein. A computer oriented single fingerprint identification system. *National Bureau of Standards, NBS Technical note 443*, Jan. 1968.

[130] J.H. Wegstein. Matching fingerprints by computers. *National Bureau of Standards, NBS Technical note 466*, Jul. 1968.

[131] Haim J. Wolfson and Isidore Rigoutsos. Geometric hashing: An overview. *Computing in Science and Engineering*, 4(4):10–21, 1997.

[132] John Woodward, Nicholas M. Orlans, and Peter T. Higgins. *BioMetrics: Identity Assurance in the Information Age*. Broché, 2003.

[133] Wenhua Xu and Minying Cheng. Cancelable voiceprint template based on chaff-points-mixture method. *Computational Intelligence and Security, International Conference on*, 2:263–266, 2008.

[134] Wei-Yun Yau, Hoang-Thanh Tran, Eam Khwang Teoh, and Jian-Gang Wang. Fake finger detection by finger color change analysis. In Lee and Li [68], pages 888–896.

[135] David Zhang and Anil K. Jain, editors. *Biometric Authentication, First International Conference, ICBA 2004, Hong Kong, China, July 15-17, 2004, Proceedings*, volume 3072 of *Lecture Notes in Computer Science*. Springer, 2004.

[136] David Zhang and Anil K. Jain, editors. *Advances in Biometrics, International Conference, ICB 2006, Hong Kong, China, January 5-7, 2006, Proceedings*, volume 3832 of *Lecture Notes in Computer Science*. Springer, 2006.

[137] Yuliang Zheng, editor. *Advances in Cryptology - ASIACRYPT 2002, 8th International Conference on the Theory and Application of Cryptology and Information Security, Queenstown, New Zealand, December 1-5, 2002, Proceedings*, volume 2501 of *Lecture Notes in Computer Science*. Springer, 2002.

APPENDIX A

List of Publications

Journals with peer review

2001 Denis Praca, Claude Barral
From Smart Cards to Smart Objects: The Road to New Smart Technologies (Computer Networks n^{o}36,2001) [89]

2008 Claude Barral, Assia Tria
Fake Fingers in Fingerprint Recognition: Glycerin supersedes Gelatin (SICS'08, Formal to Practical Security - LNCS 5458 p57-p69) [9]

Conferences with peer review and proceedings

2004 Claude Barral, Jean-Sébastien Coron, David Naccache
Externalized Fingerprint Matching (ICBA 2004 / LNCS 3070) [8]

2006 Claude Barral, Serge Vaudenay
A Protection Scheme For MoC-Enabled Smart Cards (BSYM'06, IEEE Xplore online) [10]

Conferences/Workshops **(with abstract peer review and slides in proceedings)**

2003 Claude Barral & Cédric Cardonnel - Half-day initiation on Biometrics and Smart Cards (eSmart'03, Sophia-Antipolis, France)

2004 Claude Barral - BioEasy (Cartes'04, Paris, France)

2005 Claude Barral - A Cryptographic Approach of Match-on-Card (Identech'05, Marseille, France)

2006 Claude Barral - About the duality "Biometrics vs. Password" (World e-ID'06, Sophia-Antipolis, France)

2006 Claude Barral - A Biometric Challenge-Response Protocol (eSmart'06, Sophia-Antipolis, France)

2007 Claude Barral - Forum Sécurité Atena "Cryptographie & Biométrie" (INT Evry, France)

2007 Claude Barral - Workshop COST2101 - Biometrics for Identity Documents and Smart Cards (Lausanne, Switzerland)

2007 Claude Barral, Louis Goubin & Pascal Paillier - The Fuzzy Interaction between Cryptography & Biometrics (eSmart'07, Sophia-Antipolis, France)

2009 Claude Barral, Sébastien Sanaur, Assia Tria & Abdel Yakoub - ASFIP: **A**ttack **S**tandardization for **FI**nger**P**rint systems certification (eSmart'09, Sophia-Antipolis, France)

Technical Reports

VulnBio [‡] **(Funded by French DCSSI -National Agency for IT Security-, confidential)**

2008 Claude Barral, Assia Tria
Etat de l'art de la fabrication de faux doigts

2008 Claude Barral, Axel Bonness, Assia Tria
Fabrication de Leurres

2008 Claude Barral
La détection du vivant appliquée à la sécurisation des capteurs d'empreintes digitales

2009 Claude Barral, Philippe De Choudens
Grille de cotation des attaques sur les systèmes biométriques

2009 Claude Barral, Philippe De Choudens
Rapport d'analyse de vulnérabilité - Tests et méthodologies

ASFIP [††] **(Funded by French ANR -National Agency for Research-, restricted)**

2008 Sébastien Sanaur, Claude Barral
(under Claude Barral's supervision, responsible of work package #1)
Etat de l'art - Création d'empreintes digitales physiques

2008 Stéphane Revelin
(under Claude Barral's supervision, responsible of work package #1)
Etat de l'art - Création d'empreintes digitales synthétiques

2008 Alain Thiébot
(under Claude Barral's supervision, responsible of work package #1)
Etat de l'art - Détection du vivant dans les systèmes biométriques à base d'empreintes digitales

[‡] **Vuln**érabilités des systèmes **Bio**métriques
[††] **A**ttack **S**tandardization for **FI**nger**P**rint systems certification

2009 Abdel Yakoub, Claude Barral
Ingéniérie de génération de fausses empreintes digitales pour l'évaluation des capteurs biométriques

2009 Claude Barral
Résultats des tests d'attaques sur un échantillon de capteurs biométriques représentatifs

2009 Claude Barral
Méthodologies de tests pour l'évaluation des capteurs biométriques

2009 Abdel Yakoub, Claude Barral
Nouvelles méthodologies de réalisation de leurres

Micro-PackS (Micro-Packaging & Security Labs)

2009 Claude Barral
Minex II - Overall Results Analysis & positioning of Micro-Packs/Gemalto Match-on-Card algorithm

Patents

2003 Claude Barral, Jean-Sébastien Coron, David Naccache, Cédric Cardonnel
Biometric identification method and device adapted to verification on chip cards.
EP1634220 / WO2004109585 / FR2855889 / US2005011946 /US7433501

2007 Claude Barral
Method and device for automatic authentication of a set of points.
EP1990757 / WO2008141872

2009 Claude Barral
Biometrics-based secure true random number generator.
ongoing registration

APPENDIX B
Curriculum Vitae

CURRICULUM VITAE

CLAUDE BARRAL

ADDRESS

n^{o}6 La Bastide Samat
13119 Saint Savournin, France
Phone: +33.4.42.32.36.92
Mobile Phone: +33.6.86.83.19.55
Email: claude.barral@gmail.com
Homepage: http://www.linkedin.com/in/cbarral

PERSONAL DETAILS

Gender: Male
Date of birth: 27th of October, 1968
Place of birth: Marseille, France
Present Citizenship: French
Marital Status: Married, 3 children (born 1997, 2001 & 2006)

EDUCATION

1984–1988	Baccalaureat (E-levels) : Mathematics and Technology Lycée polyvalent Antonin Artaud, Marseille, France
1988–1990	Diplôme Universitaire de Technologie GEII : Electronics Institut Univ. de Technologie, St Jérôme, Marseille, France (international level equivalence: L -Bachelor-)
1993–1994	Diplôme d'Etude Supérieure Technique : Signal Processing Institut de Promotion Supérieure du Travail, Marseille, France (international level equivalence: M -Master-)
1995	Diplôme d'Etude Supérieure Technique : Electronics Conservatoire National des Arts & Métiers, Marseille, France (continuing education) (international level equivalence: M -Master-)
1999	Engineering Degree in Electronics Conservatoire National des Arts & Métiers, Marseille, France (continuing education)

Since 07/2004	Ph.D. student at the Swiss Federal Institute of Technology (EPFL) in Lausanne, Switzerland. (continuing education)
	Thesis title: *Biometrics & Security: Issues when combining Fingerprints, Smart Cards and Cryptography* - Supervisor: Prof. Serge Vaudenay
	Doctoral School completed: Cryptography, Advanced Cryptography, Management of Innovation, Scientific Communication (all passed)
	Ph.D. defense on June 14th

WORKING EXPERIENCE

03/1992–01/1993	Technical Assistant at Commissariat à l'Energie Atomique, Cadarache, France
07/1994–09/1994	Technical Assistant at Eurocopter, Marignane, France
10/1994–01/1995	Technical Assistant at Merlin-Gérin Provence, Lamanon, France
02/1995–01/1998	Technical Assistant at Napac Méditerranée, Aix-en-Provence, France
02/1998–12/2005	Research Engineer at Gemplus Card International, Gémenos, France
since 01/2006	Research Scientist at Gemalto, La Ciotat, France (part-time since 03/2008)
since 03/2008	Scientist/Lecturer at Ecole des Mines and CEA-Leti, Gardanne, France (part-time)

EXPERTISE

Electronic Design
Smart Cards and Portable Personal Devices
Biometrics and Personal Authentication
Cryptography and Communication Security
Scientific Communication
Standardization Activities and Funded Projects
Industrial and Academic Relationship

Language Knowledge

French	native
English	written, spoken and read fluently
German	high school basics

Interests

Music	Bass Guitar player in a band
Running	Long distance, Marathon
Others	Soccer, Swimming, Diving, Squash, Mountain Biking

Publications

From Smart Cards to Smart Objects: the road to new smart technologies (Computer Networks n^{o}36, 2001)

Externalized Fingerprint Matching (ICBA 2004 / LNCS 3070)

A Protection Scheme For MoC-Enabled Smart Cards (BSYM'06, IEEE Xplore online)

Fake Fingers in Fingerprint Recognition: Glycerin supersedes Gelatin (SICS'08, LNCS 5458)

"La biométrie faciale se laisse berner"
Expert's Point of View (01 Informatique n^{o}1985, March 2009)

Lectures

since 2003	Gemplus & Gemalto, La Ciotat, France
2005, 2006	Ecole Supérieure d'Ingénieur de Luminy, Marseille, France
2005 → 2007	ENSIMAG, Grenoble, France
2005 → 2010	Ecole Centrale, Marseille, France
2007	Institut Supérieur d'Optique, Saint-Etienne, France
2008, 2009	Ecole Supérieure des Mines de Saint-Etienne, Gardanne, France

2009	Telecom & Management Sud Paris (ex-INT), Evry, France
2010	Institut Supérieur d'Electronique et du Numérique, Toulon, France

PATENTS

2003	Biometric identification method and device adapted to verification on chip cards EP1634220 / WO2004109585 / FR2855889 / US2005011946 /US7433501
2007	Method and device for automatic authentication of a set of points EP1990757 / WO2008141872
2009	Biometrics-based secure true random number generator ongoing registration

AWARD

2004	Innovation award at Cartes'04 with BioEasy (Secure & Lightweight Fingerprint Recognition in a Smart Card)

ORAL COMMUNICATIONS

2000	From Smart Cards to Smart Objects (GDC'00, Montpellier, France)
2003	Half-day initiation on biometrics and smart cards (eSmart'03, Sophia-Antipolis, France)
2004	Externalized Fingerprint Matching (ICBA'04, Hong Kong, China)
2004	BioEasy (Cartes'04, Paris, France)
2005	A Cryptographic Approach of Match-on-Card (Identech'05, Marseille, France)
2006	About the duality "Biometrics vs. Password" (World e-ID'06, Sophia-Antipolis, France)
2006	A Biometric Challenge-Response Protocol (eSmart'06, Sophia-Antipolis, France)
2006	A Protection Scheme for MoC-Enabled Smart Cards (Biometric Symposium 2006, Baltimore, United States)
2007	Forum Sécurité Atena "Cryptographie & Biométrie" (INT Evry, France)
2007	Workshop COST2101 - Biometrics for Identity Documents and Smart Cards (Lausanne, Switzerland)
2007	The Fuzzy Interaction between Cryptography & Biometrics (eSmart'07, Sophia-Antipolis, France)

2009 ASFIP: **A**ttack **S**tandardization for **FI**nger**P**rint system certification (eSmart'09, Sophia-Antipolis, France)

Last update: June 21, 2010