Sécurité des systèmes biométriques : révocabilité et protection de la vie privée

Rima Ouidad Belguechi

ECOLE NATIONALE SUPÉRIEURE D'INFORMATIQUE

Ecole Nationale Supérieure d'Informatique

Thèse de doctorat

Par

Rima Ouidad Belguechi

pour obtenir le

Doctorat de l'Ecole Nationale Supérieure d'Informatique
Spécialité : Informatique

Sécurité des systèmes biométriques : révocabilité et protection de la vie privée

SOUTENUE LE 30 JUIN 2015 DEVANT LES MEMBRES DU JURY :

*M^lle^*. BENATCHBA Karima	Professeur, ESI	(Présidente)
M. DJEDI Noureddine	Professeur, Université de Biskra	(Examinateur)
M. BALLA Amar	Professeur, ESI	(Examinateur)
*M^me^*. HAMAMI Latifa	Professeur, ENP	(Examinatrice)
M. ROSENBERGER Christophe	Professeur des Universités, ENSICAEN	(Directeur de thèse)

Pour
Mohammed et Nabiha
Mes tendres parents
Qui m'ont toujours eue dans leur poche

Résumé

En référence à la sécurité informatique, la biométrie concerne l'utilisation des caractéristiques morphologiques ou comportementales pour déterminer ou vérifier l'identité d'un utilisateur.
Récemment, des discussions sur la sécurité des systèmes biométriques ont émergé. Le stockage des données de référence pose de sérieux problèmes de sécurité et d'invasion de vie privée : manipulation d'informations sensibles, reconstruction de la biométrie d'origine à partir du modèle stocké, construction d'un échantillon biométrique falsifié, utilisation secondaire des informations biométriques (surveillance, discrimination, etc.) ou l'impossibilité de révoquer l'identifiant biométrique lorsqu'un vol d'identité à eu lieu.
La sécurité du modèle biométrique est l'une des tâches les plus cruciales dans la conception d'un système biométrique sécurisé. Considérant la modalité d'empreintes digitales, nous proposons dans cette thèse deux types de solution à ce problème. La première au niveau algorithmique et la seconde au niveau architectural.
Dans l'approche fonctionnelle ou algorithmique, nous traitons des schémas de protection des modèles biométriques. Il s'agit d'un nouveau concept dont le but est de générer une biométrie révocable en appliquant des transformations, idéalement à sens unique. Plusieurs schémas de biométrie révocable ont été proposés dans la littérature, mais pour l'heure, des efforts sont attendus pour améliorer leur fiabilité. Un schéma de biométrie révocable est une chaine de traitement qui inclut les phases d'extraction des caractéristiques, de transformation et de comparaison. Toutes ces phases sont traitées dans cette thèse. Principalement, nous nous intéressons aux descripteurs de texture d'empreintes digitales. Un premier schéma révocable, en utilisant une description de la texture globale de l'empreinte est proposé. Pour améliorer les résultats, ce schéma est étendu aux minuties. Une approche de transformation par projection aléatoire est ensuite opérée.
L'une des difficultés est d'évaluer correctement le schéma de biométrie révocable généré. Nous proposons un modèle d'évaluation basé sur un ensemble de métriques quantitatives, pour mesurer les critères de sécurité et de protection de vie privée souhaités.

Dans la seconde solution, nous proposons d'utiliser une architecture fermée pour le système de vérification biométrique. Les cartes à puce sont utilisées pour une meilleure gestion des données d'authentification de l'utilisateur. Un système de biométrie révocable avec un algorithme de comparaison sur la carte est proposé. Un tel système offre des avantages combinés de révocabilité et de confidentialité du modèle biométrique. Nous utilisons une JavaCard que nous gérons conformément à la norme PKCS15 pour plus d'interopérabilité.
Nous proposons ensuite d'étudier les possibilités de menaces de vie privée dans l'application des passeports biométriques. Nous concluons par le fait que la biométrie révocable serait souhaitable pour améliorer la protection des données biométriques contenues dans la puce du passeport.

Summary

In reference to information security, biometrics concerns the use of morphological or behavioral characteristics to determine or verify identity of an user.
Recently, discussions on security of biometric systems have emerged. The storage of biometric data arises serious privacy and security issues : exposure of sensitive information, possibility of raw data reconstruction, possibility of spoof attack, secondary use of information (function creep or cross-matching) or the impossibility to revoke template in case of identity theft.
So, template security is one of the most crucial issues in designing a secure biometric system. Considering the fingerprint modality, we propose in this thesis two types of solution. The first one, at the algorithmic level and the second at the architectural level.
In functional or algorithmic approach, we deal with template protection schemes. This is a new concept which aims to generate a cancelable biometrics by applying transformations, ideally a one way functions. Several schemes have been proposed in the literature, but efforts are expected to improve their reliability. The processing chain includes a feature extraction, a transformation and a comparison phases. All these steps are discussed in this thesis. Primarily, we focus on texture descriptors. A cancelable system using a description of the global texture is provided. To improve results, this scheme is extended to minutiae. A BioHashing approach is then carried out.
One difficulty is to properly analyze the cancelable system. We propose an evaluation model based on a set of metrics to measure the criteria of security and privacy.

In the second solution, we propose a closed biometric system. Smartcards are used for a better management of cancelable constructs. A chip matching algorithm is then proposed. Such a system offers combined benefits of revocability and confidentiality. A PKCS15 compliant JavaCard is then proposed.
We also study privacy issues in biometric passports. We conclude that cancelable biometrics is desirable to improve the protection of biometric data contained in the passport chip.

Remerciements

Même si ces remerciements se prolongeraient à l'infini, ils me seraient sans doute insuffisants pour exprimer toute ma gratitude à mon directeur de thèse, le Pr. **Christophe Rosenberger**. Voulant travailler sur des thématiques nouvelles liées à la biométrie, j'ai beaucoup cherché, ici et ailleurs, un encadrant pour mes idées. Le Pr. Christophe Rosenberger a tout de suite su discerner le tumulte résonant qui était dans ma tête en étoffant ce sujet de thèse qui me tient fortement à cœur. La réussite du chercheur dépend spécialement de ses capacités à s'ouvrir sur le monde ou à échanger ses idées, en m'acceptant dans son équipe *monétique et biométrie*, il m'a offert la chance d'enrichir et de consolider ma formation. Je souhaite aussi le remercier pour sa totale disponibilité, ses remarques acérées, sa patience et la pleine confiance qu'il m'a accordée tout le long de nos années de collaboration.

Aux moments les plus opportuns, les plus décisifs, il a su me guider et me conseiller efficacement. Qu'il me soit permis de remercier mon co-directeur de thèse, le Pr. **Samy Ait-Aoudia**, pour cela et pour le niveau de pertinence et de rigueur ajouté à ce travail. C'est aussi, mon enseignant, qui a solidement contribué à ma formation, ma reconnaissance lui est grandement acquise.

Les remerciements qui suivent iront au Dr. **Estelle Cherrier**, la force pérenne qui a bien voulu co-encadrer cette thèse. Qu'il me soit permis de saluer son plein investissement et sa bienveillance à mon égard. C'est avec honneur que j'ai pu bénéficier de ses connaissances et de ses remarques et critiques aiguisées. Je la remercie, particulièrement, pour son empathie dont elle a fait preuve depuis que nous travaillons ensemble.

Je tiens à remercier le Pr. **Karima Benatchba** d'avoir accepté de présider mon jury de thèse. Je suis également reconnaissante aux professeurs **Noureddine Djedi**, **Amar Balla** et **Latifa Hamami** d'avoir accepté d'examiner et d'honorer de leur point de vu ce travail.

Je remercie, les fiers gardiens de mon pays, tous ceux qui ont contribué à faciliter le déroulement de cette thèse, tous ceux qui m'ont permis d'accomplir sereinement mes différents stages, **Mr. Azzedine Benyahia**, en particulier. A mon tour, j'espère perpétuer ma servitude pour contribuer au développement de mon pays.

Je remercie tous les référées, des différentes conférences et journaux, qui ont pu lire, corriger, critiquer ou revoir mes travaux. Je remercie particulièrement, le Dr. **Patrick**

Lacharme d'avoir pu travailler avec lui. Je remercie les chercheurs prédécesseurs, source d'inspiration, des recherches actuelles.

Mes autres remerciements iront aux personnes de mon cœur. Papa, maman, à la vue du monde, je vous exprime ma gratitude et ma reconnaissance. Garants de mon bonheur, sans vous, jamais je ne serai parvenue, avec vous et pour vous, j'espère continuer.
Baba, grand baba, mon premier père, je te remercie parceque je te ressemble. Ton soutien et ta présence son chers à mon cœur.
Je remercie mon frère protecteur **Sofiane**, c'est une chance de toujours t'avoir à mes côtés, me soutenant, m'écoutant et me conseillant.
Je finis par remercier mon époux, mon petit trésor **Khadidja** d'avoir su au-delà son bas-âge m'accompagner vers l'achèvement de cette thèse.

Table des matières

Introduction générale

« Ce ne sont pas les murs qui protègent la citadelle, mais l'esprit de ses habitants »

Thucydide Ve siècle av. J.C

Contexte

Les réseaux informatiques à grande échelle (internet, réseaux mobiles, réseau interbancaire, etc.) se sont consolidés pour constituer une plateforme puissante qui a révolutionné le monde de la communication et des échanges dématérialisés. Les services commerciaux et bancaires, le gouvernement en ligne, le partage de connaissances, les soins de santé à distance sont d'autant d'applications sensibles qui émergent sur ces réseaux. La sécurité des transactions résultantes est donc primordiale.

Dans ce contexte, l'authentification des utilisateurs de ces services est l'un des défis les plus importants. Dans ce monde numérique, à cause du nombre important des cas d'usurpation d'identité, l'étape d'authentification est souvent considérée comme le maillon faible de la sécurité informatique. En général, la protection du contenu est réalisée en utilisant des protocoles cryptographiques standardisés, bien connus et établis. Pour l'authentification d'un individu, le mot de passe est de loin la méthode la plus répandue en dépit de son manque évident de sécurité (perceur de mot de passe, vol par écoute, etc.) et de sa facilité d'usage très limitée lorsque l'utilisateur souhaite accéder à une multitude de services. D'autres solutions d'authentification existent comme les OTPs (mots de passe à usage unique), les certificats numériques, les jetons cryptographiques ou les cartes électroniques. Bien qu'assez sophistiqués, ces mécanismes ne peuvent garantir, avec précision, le lien direct entre l'utilisation du service et l'utilisateur actuel étant donné qu'ils peuvent être volés, perdus, partagés ou bien même dupliqués.

La biométrie qui sert à reconnaitre, de façon automatique, une personne sur la base de ses

caractéristiques physiologiques ou comportementales se présente comme une alternative d'authentification à très fort potentiel. Contrairement aux méthodes d'authentification citées pécédemment, seule cette technologie garantit un lien fort entre l'authentifiant et l'utilisateur.

Un système biométrique peut être considéré comme un système de traitement de signal avec une architecture de reconnaissance de formes. Il capte le signal biométrique, le traite puis extrait un ensemble de caractéristiques représentatives appelées modèle biométrique (*biometric template*) qui seront ensuite comparées au modèle préalablement stocké sur une base de données.
Cette technologie possède de nombreux avantages. Premièrement, il existe un lien étroit entre l'utilisateur et son identifiant. Par exemple, il n'est pas possible de perdre son empreinte comme cela pourrait être le cas pour un jeton ou de la partager avec une autre personne, ce qui constitue un rempart efficace contre le problème de répudiation. D'autre part, la biométrie offre un certain confort d'utilisation. En effet, au lieu de se souvenir de mots de passe aussi diversifiés que complexes, il est plus pratique pour un utilisateur de s'authentifier en mettant son doigt sur un capteur ou de faire une capture de visage. Enfin, l'unicité et la permanence de la modalité biométrique considérée, offrent une manière performante et fiable d'authentifier les personnes. Cependant, comme toute technologie sécuritaire, il est impératif d'évaluer ses risques potentiels et de faire en sorte que la sécurité souhaitée soit bien obtenue. Plusieurs études [RCB01, BCR02, Adl07] font le constat de problèmes dans les systèmes biométriques et qui sont soit *(i)* inhérentes à la technologie biométrique soit *(ii)* intentionnellement causées par des attaques adverses. Principalement, les vulnérabilités liées à la biométrie peuvent se résumer comme suit :

1. L'usurpation d'identité : le fraudeur peut essayer de se faire passer pour un utilisateur légitime. Par exemple, en compromettant le modèle biométrique stocké dans la base de données, il lui est possible de reconstruire un signal artificiel proche du signal d'origine capable de passer avec succès le seuil de décision de vérification.
2. L'irrévocabilité : le principal inconvénient de la biométrie est qu'en cas d'abus ou de compromission du modèle biométrique, celui-ci ne peut être, en général, ni révoqué, ni remplacé, ni mis à jour.
3. La violation de la vie privée : l'utilisation du corps humain comme outil d'identification et sa sauvegarde dans les bases de données, parfois même au-delà des frontières, posent un vrai problème d'éthique. De plus, la biométrie faisant associer des données personnellement identifiables et donc sensibles [Cav08], sa collecte, son stockage et son utilisation doivent être régis par des juridictions légales. Alors que l'unicité de la biométrie est vue comme un avantage, elle peut aussi être considérée comme une

possibilité de profilage et de surveillance d'une personne, menaçant ainsi sa liberté individuelle. Ainsi, la mise en place d'un système biométrique doit être fondée sur un fort impératif de préservation de la vie privée.

Au vu des risques présentés, un système biométrique doit avant son déploiement être soumis à différentes contraintes de sécurité et de protection de la vie privée. Il devient indispensable d'assurer la sécurité des systèmes biométriques et de protéger l'identifiant biométrique avec de solides contre-mesures.

Cette thèse s'intéresse principalement à la protection du modèle biométrique, qui présente des menaces de violation de vie privée, potentiellement envahissantes. Le but est d'apporter des solutions approuvées qui puissent réconforter les appréhensions dans les systèmes biométriques sans pour autant en diminuer les fonctionnalités. Le prochain paragraphe explique plus en détails nos objectifs de recherche.

Objectifs de recherche

La prise en compte des problématiques de sécurité dans les systèmes biométriques est d'actualité dans l'industrie et la recherche, mais pour l'heure, les moyens mis en œuvre ne sont pas toujours suffisants. S'agissant du problème de manipulation de données d'ordre privé, des commissions et organismes on fait leur apparition comme la CNIL (Commission Nationale d'Informatique et des Libertés Individuelles) en France. En Algérie, il n'existe pas encore d'institution crédible capable de juger la légalité d'une opération de surveillance biométrique.

Ces autorités de protection de données doivent garantir que le déploiement des systèmes biométriques est fait de telle sorte que l'accès aux données biométriques soit limité aux personnes autorisées, sous certaines conditions, avec un contrôle explicite sur l'usage final. Cependant, au lieu d'être simplement basé sur des principes de bonnes pratiques, il serait plus judicieux de garantir la préservation de la vie privée au niveau de la conception du système biométrique, avant même son déploiement. Aujourd'hui, un nouvel axe de recherche est apparu, dont le but est d'intégrer la protection de la vie privée comme contrainte fonctionnelle du système biométrique. Il s'agit de rechercher des solutions techniques (et non uniquement juridiques), qui compléteraient avec la technologie biométrique, afin de protéger les données de référence. Toutefois, il a été constaté qu'en améliorant la préservation de la vie privée, les performances de reconnaissance avaient tendance à se dégrader de façon notable. Les objectifs de sécurité et de vie privée sont souvent contradictoires et l'un des défis majeurs est de garantir une somme positive entre les deux.

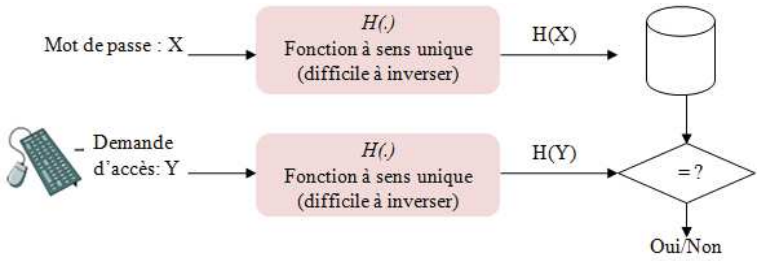


FIGURE 0.1 – Protection des mots de passe par les fonctions de hachage

La solution naturelle pour assurer la protection de données sensibles est l'utilisation d'outils cryptographiques en garantissant leur confidentialité. On pourrait chiffrer la donnée ou bien appliquer le schéma classique de protection des mots de passe reposant sur les fonctions de hachage (fonctions à sens unique) comme présenté sur la figure 0.1. Cependant, à cause de la variabilité du signal biométrique, la cryptographie classique s'avère non efficace, car dans tous les cas, la comparaison se ferait dans l'espace en clair et le modèle serait de nouveau exposé.

Les techniques les plus prometteuses de protection du modèle biométrique reposent sur des approches de transformations où on sauvegarderait dans la base de données une version transformée du modèle d'origine. La comparaison se fera alors dans ce nouveau domaine. Afin de garantir la préservation de la vie privée, il faudrait que le modèle transformé ou bien sécurisé soit :

1. *Non inversible*, ce qui mesure la difficulté de remonter au modèle d'origine à partir du modèle transformé et donc cela préservera de la divulgation des données personnelles.
2. *Diversifié*, ce qui empêchera de lier les modèles transformés entre eux même s'ils appartiennent à la même personne. Cela empêchera le suivi des individus entre plusieurs bases de données.

En même temps, il faudrait garantir les exigences de sécurité suivantes :

1. Des performances de reconnaissances fiables, de telle sorte que le modèle transformé ne devrait pas dégrader les performances du système.
2. Une méthode de révocabilité et de remplacement du modèle transformé à partir du même trait biométrique, pour pallier le problème de compromission du modèle de référence.

Le verrou scientifique identifié consiste à trouver une approche de transformation qui puisse garantir ces exigences en présence d'un signal biométrique naturellement variable. **C'est**

dans cette problématique que s'inscrit l'objectif global de cette thèse. A ce stade, il est important de noter que l'approche de transformation proposée dépend fortement de la représentation du modèle biométrique car :

- Différentes modalités vont donner lieu à différentes représentations du modèle biométrique : continu ou discret, ordonné ou non ordonné, vecteur ou ensemble, etc.
- La modélisation du bruit qui s'ajoute à l'information biométrique, représentant la variabilité liée à la capture ou à la donnée biométrique, diffère remarquablement d'une modalité à une autre. Le capteur, l'environnement, les conditions d'acquisition, et l'algorithme d'extraction des caractéristiques influent sur la nature du bruit inclus dans le modèle final.

Il est donc important de raffiner notre objectif de départ en sélectionnant, dès maintenant, la modalité biométrique étudiée. Dans cette thèse, nous ciblons la reconnaissance par empreintes digitales pour les raisons suivantes :

- L'empreinte digitale domine le marché de la biométrie aussi bien sur un plan privé que public ou gouvernemental. Le rapport 2009-2014 fournit par l'International Biometric Group le confirme [gro13].
- L'empreinte digitale est considérée comme invasive en termes de violation de la vie privée. En effet, les modalités biométriques ne comportent pas les mêmes risques de violation de vie privée. Une étude faite dans [Thi03] montre que le visage et l'empreinte digitale sont des modalités hautement risquées à cause de leur facilité d'accès (capture à l'insu de la personne ou récupération sur supports ou avec une caméra). L'empreinte est une modalité à trace et le visage peut facilement être pris par une caméra. De plus, leur haute compatibilité avec les bases de données existantes (i.e. le système EURODAC, le système US-VISIT, etc.) peut encourager le suivi et la surveillance des individus. L'iris et la rétine sont considérées comme moyennement risquées alors que la signature, la voix et la dynamique de frappe sont considérées comme faiblement risquées.

C'est donc la protection des empreintes digitales qui constitue la question principale de notre recherche. Pour arriver à cet objectif, nous étudions dans cette thèse les deux aspects suivants qui sont différents mais complémentaires :

Aspect I : proposition d'un schéma algorithmique de biométrie révocable pour la protection du modèle d'empreintes digitales.
Il s'agit de proposer une approche de transformation révocable qui réunirait les exigences de sécurité et de préservation de vie privée citées précédemment. L'approche de transformation inclut la phase d'extraction des caractéristiques à partir de l'empreinte digitale et

la génération d'un modèle transformé révocable. Elle inclut aussi la phase de vérification dans le domaine de la transformée ainsi que le module de révocation. Nous focalisons sur cet aspect en traitant des questions suivantes :

- Comment le modèle biométrique est-il représenté ? Quel est le schéma de transformation appliqué ? Quelle référence sera effectivement stockée ?
- Comment l'alignement géométrique des empreintes a-t-il été traité ? Est-ce qu'un pré-alignement est nécessaire ?
- Comment le modèle peut-être révoqué ?
- Comment évaluer la sécurité du nouveau système biométrique ? Comment estimer la précision du système de reconnaissance ? Comment estimer sa robustesse face à la préservation de la vie privée et aux attaques possibles ?

Aspect II : Gestion décentralisée du modèle d'empreinte digitale.
Dans le contexte de la gestion et de la protection des données personnelles et sensibles, il est, comme nous l'avons vu, important de s'intéresser au stockage de ces données. Alors que les bases de données centrales présentent différents risques d'administration, d'accès illicite ou de compromission, l'utilisation de dispositifs portables et sécurisés améliore nettement la préservation de la vie privée. Nous estimons, dans cette thèse, intéressant d'exploiter aussi cet aspect des choses où la solution de protection est aussi au niveau architecturel et non complètement au niveau algorithmique. Cet intérêt est justifié par la prolifération des dispositifs sécurisés comme les cartes à puce, qui sont de plus en plus utilisées comme moyen d'authentification pour diverses applications (d'ordre gouvernemental, bancaire ou autre) et à partir de plusieurs points d'accès : distributeur de billets ou internet par exemple. Nous nous intéressons à la gestion décentralisée des données biométriques sous les deux volets suivants :

1. D'abord, nous nous intéressons à la protection des données d'empreintes digitales en utilisant les cartes à puce ouvertes comme celles programmées en JavaCard. Le but est de proposer des cartes d'identité personnelles intégrant nos contributions en biométrie révocable. Un tel système combine les avantages de la biométrie révocable avec la sécurité connue des cartes à puce.

2. Ensuite, nous étudions une application déjà existante qui est celle du passeport biométrique. Sans doute, il s'agit là de l'application la plus visible de la biométrie à grande échelle. Dès lors que le document de transport contient des données biométriques, il peut être considéré comme un dispositif portable et donc décentralisé. On s'intéresse de plus près à la sécurité de ces données et à la possibilité de menace de vie privée contenue dans ce type d'application. Il nous incombe ensuite de faire des

propositions pour améliorer les manques existants.

Principales contributions

Nous remarquons que de nombreuses approches de protection des empreintes digitales existent, cependant, le déploiement d'une solution respectueuse de la vie privée qui maintiendrait les performances de reconnaissance reste un sujet de recherche ouvert. En outre, il faut noter que l'absence, pour le moment, d'un processus rigoureux et standardisé pour l'évaluation de ces méthodes a des répercussions directes sur la confiance dans ces systèmes de protection. Dans beaucoup de travaux de recherche, nous constatons que les propositions ne sont pas toujours suffisamment résistantes aux différentes attaques possibles. En ce qui concerne la biométrie révocable, un point faible est lié à la gestion de la clé spécifique utilisée comme paramètre dans les fonctions de transformation. Si cette clé est connue de l'imposteur, on observe une diminution de la performance en termes de taux de fausse acceptation. Dans certains cas, si le modèle transformé est lui aussi compromis, le processus de réversibilité devient possible. Dans cette thèse, nous avons abordé le problème de protection des empreintes digitales par les contributions qui peuvent être présentées comme suit :

1. A la lumière de notre état de l'art sur les méthodes de protection du modèle biométrique, notre intérêt s'est porté vers les fonctions de transformation non inversibles, dans le but de fournir un **système biométrique révocable**. La révocabilité étant un critère important aussi bien pour assurer la sécurité du système (en faisant des mises à jour régulières du modèle de référence) que pour la préservation de la vie privée (garantir la non traçabilité et la confidentialité de ces informations personnelles).

   Beaucoup d'algorithmes de protection nécessitent un descripteur biométrique de longueur fixe. Les attributs de texture de l'empreinte peuvent être utilisés dans ce contexte. Nous commençons notre étude par comparer différents descripteurs de texture récents comme les filtres de Gabor ou l'analyse LBP (Local Binary Pattern). Nous sélectionnons le descripteur approprié en considérant différentes contraintes comme l'efficacité ou la taille du descripteur. Les propriétés statistiques des réponses de Gabor ont été choisies comme meilleur descripteur parmi ceux étudiés. Nous proposons ensuite un schéma révocable d'empreintes digitales à base de ce descripteur.

   Afin d'améliorer les résultats, la seconde proposition est de considérer un modèle de transformation appliqué sur le modèle des minuties. Notre proposition se base sur la création d'un descripteur d'empreintes basé minuties qui considère aussi bien des

caractéristiques de texture, locales aux minuties que des caractéristiques de second degré qui expriment les relations de voisinage entre minuties. Le module de comparaison d'empreintes est assez puissant car il implique une méthode d'appariement local consolidée au niveau global par l'information de voisinage. Nous décidons ensuite de protéger ce modèle par la fonction du BioHashing [TN04].

Le BioHashing est un système de protection basé sur une transformation définie par un secret. Si le secret n'est pas compromis, il fournit une très bonne solution compte tenu de la sécurité et de la préservation de la vie privée. Néanmoins, il est bien connu par la communauté de recherche que le BioHashing classique a des résultats réduits si la clé utilisateur est compromise [LN06, TKL07]. Le BioHashing est principalement basé sur un changement d'espace, en projetant un vecteur sur une matrice. L'idée est donc de considérer un BioHashing sophistiqué appliqué sur un descripteur complexe plutôt que sur un simple vecteur. L'introduction de ce descripteur devient un facteur pour améliorer les performances du BioHashing face à plusieurs scénarios d'attaque comme : le vol de clé ou l'attaque par traçabilité (linkage attack). Concernant le problème pertinent d'alignement, nous avons d'abord considéré une solution explicite basée sur la détection du point core. Toutefois, étant donné qu'il est difficile de localiser de manière fiable ce point de référence dans toutes les images d'empreintes, nous avons amélioré la solution initiale : *(i)* en proposant une méthode de validation du point core détecté, *(ii)* en créant un descripteur de la minutie auto-aligné dans le cas où le point retourné serait non validé. Nous obtenons ainsi un système qui rivalise avec les récents développements de la littérature.

2. En raison de l'absence d'une méthode unifiée pour évaluer les algorithmes de protection, nous avons proposé un cadre d'évaluation commun, constitué d'un ensemble de métriques pour évaluer la sécurité et la préservation de la vie privée dans les systèmes à biométrie révocable. Ce cadre nous a permis d'évaluer notre travail.

3. Un algorithme de vérification d'empreintes digitales révocables pour carte à puce est proposé. Il permet de stocker le modèle de référence et l'algorithme d'appariement sur la carte. Les méthodes de protection conventionnelles utilisent la base de données centrale pour stocker le modèle transformé. Une fois la clé utilisateur compromise, l'imposteur peut exploiter le taux de fausse acceptation pour passer l'étape d'authentification.
   Le système proposé se base sur une authentification à 3 facteurs (clé + carte + biométrie) ce qui constitue une protection robuste contre l'attaque par vol de clé.

Pour son utilisation dans un contexte applicable, la carte développée est compatible avec le standard PKCS15 [PKC00b]. La nouveauté de ce travail est la proposition d'une carte avec une comparaison embarquée exploitant la solution de biométrie révocable compatible avec les normes internationales.
Une autre motivation pour la conception d'une telle solution est due aux attaques possibles contre les systèmes à biométrie révocable. Comme il est difficile de prouver formellement la sécurité, le système peut être sujet à une possible attaque quand cela a été initialement ignoré par l'étape d'évaluation. Cela fut le cas pour de nombreuses propositions où un inverse ou un pseudo-inverse du modèle original a pu être estimé à partir du modèle transformé. De telles attaques sont généralement réalisables dans le cas où plusieurs modèles transformés révoqués sont révélés en même temps que les paramètres de l'algorithme. Si nous maintenons le modèle transformé totalement confidentiel, ces attaques peuvent être évitées. Le système intégré proposé garantit cette exigence.

4. Nous abordons dans cette partie le problème de violation de vie privée dans l'architecture actuelle du passeport électronique pour le stockage et la transmission des données biométriques. Nous proposons une nouvelle solution combinant protocoles cryptographiques et biométrie révocable pour garantir la protection des données d'empreintes digitales contenues dans le passeport.

Organisation de la thèse

Ce manuscrit de thèse est organisé selon les quatre chapitres suivants :

- **Le chapitre 1** étaie l'état de l'art. Il aborde principalement les différents travaux qui s'accentuent autour des exigences de révocabilité et de confidentialité de la donnée d'empreinte digitale. Dans une approche de *privacy par design*, l'accent est mis sur les méthodes de protection biométriques.
- **Le chapitre 2** présente la méthode d'évaluation proposée pour les méthodes de protection biométriques. Un ensemble de métriques est proposé pour analyser les approches par transformation révocable.
- **Le chapitre 3** présente nos différents schémas révocables d'empreintes digitales.
- **Le chapitre 4** met en relief les avantages de le gestion décentralisée de la donnée biométrique en proposant un système MatchOnCard.
- Une conclusion générale fait le point de cette recherche et liste les différentes perspectives de cette thèse.

Positionnement du problème & Travaux existants

Chapitre 1

Positionnement du problème & Travaux existants

Cette thèse adresse la problématique générale de la protection du modèle d'empreintes digitales impliqué dans les systèmes d'authentification biométrique. Ce chapitre passe en revue les propriétés d'un système de reconnaissance par empreintes digitales, met l'accent sur le contexte du problème et fait un tour d'horizon sur les travaux existants et les principales problématiques de recherche qui restent ouvertes.

Sommaire

1.1 Généralités sur la reconnaissance par empreintes digitales

1.1.1 Empreintes digitales et biométrie

L'existence de plusieurs modalités biométriques, impose le problème de choix entre elles lors de la conception d'un système biométrique. Il ne suffit pas de comparer les performances des diverses technologies (empreintes, visage, main, etc.) pour les distinguer,

Techniques	Avantages	Inconvénients
Empreintes digitales	Coût, ergonomie moyenne, facilité de mise en place, taille du capteur réduite, fiabilité	Acceptabilité moyenne, possibilités d'attaques (moulage)
Forme de la main	Bonne acceptabilité	Système encombrant, coût, caractère non discriminant par rapport aux membres de la même famille
Visage	Coût, peu encombrant, bonne acceptabilité, authentification à distance	Problème des jumeaux, sensibilité au déguisement, possibilité de discrimination (éthnique, religieuse, etc.)
Rétine	Fiabilité, pérennité	Coût, Acceptabilité faible, installation difficile
Iris	Fiabilité	Acceptabilité faible, contrainte d'éclairage
Voix	Fiabilité moyenne	Vulnérabilité aux attaques
Signature	Ergonomie	Dépendant de l'état émotionnel de la personne, problème de fiabilité
Dynamique de frappe sur un clavier	Ergonomie, ne nécessite pas de capteur	Dépendant de l'état physique de la personne

TABLE 1.1: Avantages et inconvénients des différentes technologies biométriques

il faut aussi tenir compte d'autres critères telles que la facilité de leur usage ou leur acceptabilité. Il est donc important de comprendre que, dans le choix d'une modalité biométrique, différents facteurs doivent être pris en compte. Ces facteurs ont été discutés par le CLUSIF [1] dans le tableau 1.1.

Comparativement aux autres technologies, et comme indiqué sur le tableau 1.1, l'empreinte digitale se distingue comme étant une modalité bien approuvée techniquement, pas très coûteuse et parmi les plus fiables en terme d'erreur à la reconnaissance. De nos jours, les empreintes sont largement utilisées et reconnues comme méthode d'identification fiable et mature.

1. https ://www.clusif.asso.fr

1.1.2 Description des empreintes digitales

Une empreinte digitale est le dessin formé par les lignes de la peau des doigts. Elles sont uniques et immuables, elles ne se modifient donc pas au cours du temps (sauf par accident) mis à part leur qualité qui peut se dégrader. Une empreinte digitale est constituée d'un ensemble de lignes localement parallèles formant un motif (ridge pattern) unique pour chaque individu. On distingue *les crêtes*, ce sont les lignes en contact avec une surface au touché et *les vallées*, ce sont les creux entre deux crêtes. A l'intérieur de ce motif, il y a un très grand nombre d'éléments qui nous différencient les uns des autres. Ces caractéristiques sont formées par le flux des crêtes formant l'empreinte. La figure 1.1 illustre un exemple de ces caractéristiques.

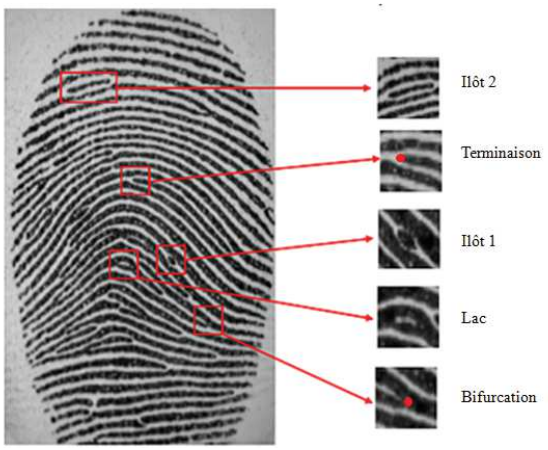


FIGURE 1.1 – Quelques caractéristiques des empreintes digitales

Ces éléments sont à leur tour découplés en deux familles : *les minuties* et *les singularités*. La minutie est l'arrangement particulier des lignes papillaires (crêtes et vallées) à l'origine de l'individualité des empreintes. Les minuties peuvent être de différents types comme le montre la figure 1.2, mais en pratique, deux types seulement sont utilisés, à savoir *les terminaisons* (le point où la crête se termine) et *les bifurcations* (le point de carrefour de plusieurs crêtes). Cela s'explique par le fait que les autres types sont des combinaisons de terminaisons et de bifurcations. Il existe deux points de singularités (figure 1.3) : *le core* et *le delta*. Le delta est localisé à la confluence de trois différentes crêtes. Le core est le point de courbure maximale.

Selon le motif (le nombre et la localisation des points delta et core), nous pouvons répertorier trois grandes familles d'empreintes comme le montre la figure 1.4 :

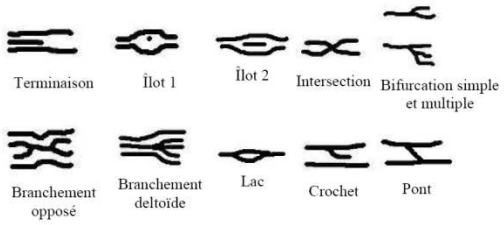


FIGURE 1.2 – Les différents types de minuties

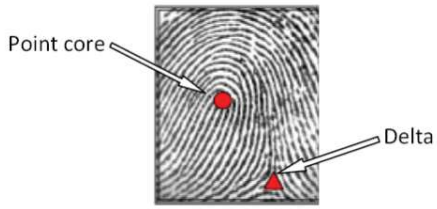


FIGURE 1.3 – Les singularités dans une empreinte

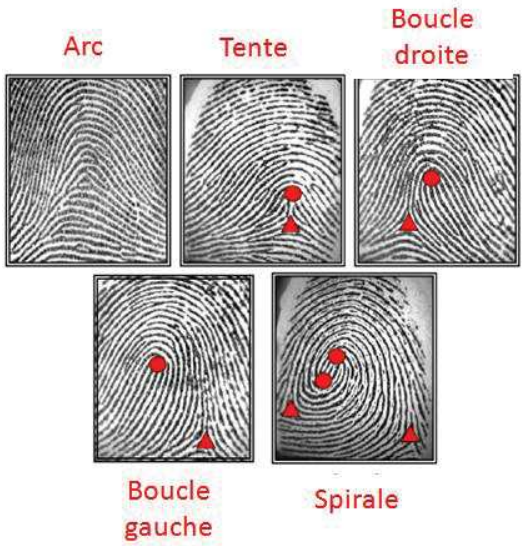


FIGURE 1.4 – Différents types d'empreintes digitales

- Les arcs ou les tentes.
- Les boucles à droite ou à gauche (loop).
- Les spirales (whorl).

Ces trois types d'empreintes regroupent 95% des doigts humains : 30% pour les spirales, 60% pour les boucles et 5% pour les tentes. A une très grande résolution, nous pouvons observer d'autres caractéristiques qui sont considérées très discriminantes, à savoir, *les pores* comme le montre le figure 1.5.

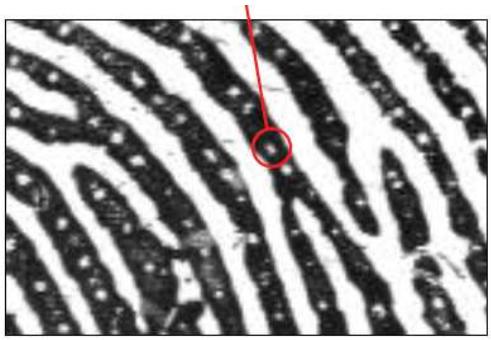


FIGURE 1.5 – Les caractéristiques visibles sur une grande résolution de 1000 dpi : les pores de la peau

1.1.3 Propriétés des images d'empreintes digitales

Le traitement des empreintes digitales s'appuie sur les propriétés intrinsèques qui caractérisent ce type d'image :

- Une empreinte est une alternance de crêtes et de vallées qui évoluent dans une direction précise. Un bloc de l'image bien défini possède une direction locale constante.
- Sur un bloc, les niveaux de gris des crêtes et des vallées constituent une forme sinusoïdale le long de la direction normale à l'orientation locale des crêtes (voir la figure 1.6). Une approximation de cette onde peut être donnée par l'équation 1.1 avec (f, θ) les paramètres d'orientation et de fréquence, respectivement. Seuls les points singuliers ou les minuties constituent une discontinuité de cette forme présumée.

$$I(x, y) = A2\pi f \cos(x \cos\theta + y \sin\theta) \tag{1.1}$$

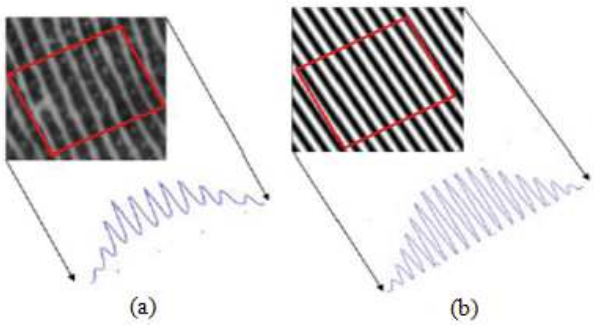


FIGURE 1.6 – Projection des crêtes sur une fenêtre orientée. (a) un échantillon d'empreinte (b) une empreinte synthétique

– Les valeurs des niveaux de gris atteignent leur maximum local le long de la direction normale aux crêtes.

Ces propriétés peuvent être matérialisées par des représentations spécifiques sous forme d'images appelées *images intrinsèques*. Cela inclut :

– L'image d'orientation : l'image d'orientation O représente l'orientation locale des crêtes sur chaque point de l'image.
– L'image de fréquence : la fréquence locale des crêtes indique la distance moyenne entre les crêtes sur chaque point de l'image.
– Le masque des régions : le masque indique l'ensemble des régions où apparaissent des crêtes. Il est aussi appelé masque région fond. D'autres masques sont capables de distinguer les régions exploitables parmi celles qui sont très endommagées et donc irrécupérables.

1.1.4 Architecture d'un système d'authentification par empreintes digitales

Un système automatique de vérification (authentification) d'empreintes digitales est une chaîne de traitement qui se scinde en deux étapes : *(i)* enrôlement ou enregistrement et *(ii)* authentification comme le montre le figure 1.7.

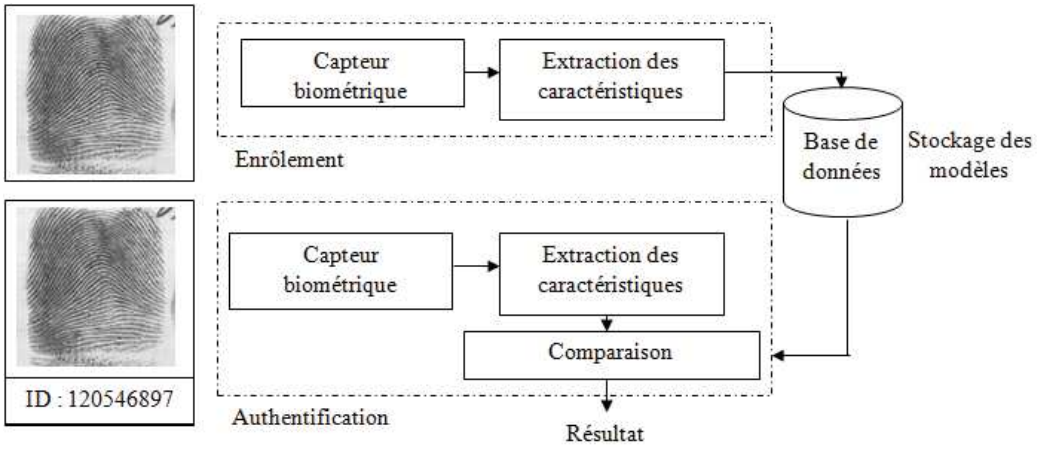


FIGURE 1.7 – Architecture d'un système d'authentification par empreintes digitales

Durant l'enrôlement, le trait biométrique de l'utilisateur est capturé (acquisition) et les caractéristiques sont extraites et sont sauvegardées dans une base de données comme **modèle de référence**. Durant l'authentification, le même trait biométrique de l'utilisateur est de nouveau capturé et les caractéristiques sont extraites et sont comparées avec celles dans la base de données pour calculer leur correspondance. L'enregistrement spécifique

choisi pour la comparaison est déterminé en fonction de l'identité annoncée par l'utilisateur. Les modules principaux de ce processus sont explicités dans ce qui suit :

Phase de capture

Le but de cette étape est de former l'échantillon biométrique sous la forme d'une image numérique en utilisant un dispositif spécial appelé *capteur*. Aujourd'hui, bon nombre de capteurs d'empreintes existe. Ils se distinguent, notamment par : leur technologie, leur coût, leur qualité d'acquisition, leur facilité d'intégration (téléphone, ordinateur portable, etc.) ou leur capacité à détourner les moulages d'empreintes.

Comme noté dans [TWW04], la qualité de l'image générée est un facteur prédictif de la performance du module de comparaison. Néanmoins, cette qualité ne dépend qu'en partie du capteur. Elle dépend aussi de l'empreinte en elle-même (distorsions élastiques, sueur, humidité, saleté, état de l'épiderme et accidents, etc.) et des conditions d'acquisition (empreintes latentes, coopération du sujet, etc.).
Selon Cappelli *et al.*[CFM08], la qualité du capteur dépend de plusieurs facteurs comme : la surface d'acquisition, la résolution (aujourd'hui autour de 500 ppi équivalente à 19.69 pixels par millimètre), la quantification des niveaux de gris (256 niveaux par exemple) ou la précision géométrique. Etudier la corrélation entre la qualité du capteur et les performances de reconnaissance est un point important qui permet de définir les exigences minimales des capteurs utilisés dans les applications biométriques.
D'un autre côté, nombreux travaux [LJY02, SKK01] s'intéressent à l'évaluation de la qualité de l'image d'empreinte digitale acquise. Une étude qui peut être intéressante pour les phases restantes du processus. Dans la mesure NFIQ présentée par le NIST [TWW04], on estime la qualité de l'image par blocs de régions à partir d'informations sur le contraste ou sur le flux d'orientation des crêtes. Dans chaque région, une valeur de qualité est assignée (qui est entre 0 et 4). Cette valeur permet de déterminer le degré de fiabilité des caractéristiques extraites subséquemment dans le module d'extraction.

Phase d'extraction des caractéristiques, et représentation

Une empreinte apparaît comme une surface alternée de crêtes et de vallées parallèles sur la plupart des régions. Différentes caractéristiques permanentes ou semi-permanentes tels que les blessures ou les coupures sont aussi présentes sur l'empreinte. Il est nécessaire de définir une représentation invariante appelée **gabarit** ou **modèle**. Cette représentation peut être globale prenant en compte toute l'image ou, locale c'est-à-dire constituée d'un ensemble de composantes dérivée chacune d'une région restreinte sur l'empreinte. La représentation finale, peut alors appartenir à l'une des catégories suivantes :

Image Dans cette représentation, c'est toute l'image qui est considérée comme une représentation possible. La comparaison est réalisée par corrélation. La corrélation entre les deux images $I_1(x, y), I_2(x, y)$, est donnée dans le domaine spatial par :

$$I_c(k, l) = \sum_x \sum_y I_1(x + k, y + l) I_2(x, y) \tag{1.2}$$

Le corrélateur établit la correspondance par la recherche de la magnitude des pics dans l'image de corrélation I_c. L'exactitude de cette corrélation se dégrade avec les transformations de l'image comme les phénomènes de translation ou de rotation. Un autre inconvénient est en relation avec la taille conséquente de l'image à sauvegarder durant l'enrôlement.

Représentation en minuties Les détails des minuties constituent la représentation la plus populaire de toutes les représentations existantes. Elles répondent efficacement au problème de taille posé précédemment. Les minuties peuvent être appariées en considérant le problème comme un problème d'appariement des primitives points (point pattern matching). Dans le cas idéal, nous supposons plusieurs propriétés :

1. La correspondance entre les deux ensembles de minuties est totale.
2. Il n'y a aucune déformation telles que la translation ou la rotation entre elles.
3. Chaque minutie présente sur une empreinte est exactement localisée.

Alors, la vérification d'empreinte devient une tâche insignifiante. Cependant, dans la pratique, la qualité de l'empreinte rencontrée durant la vérification est très **incertaine**, elle varie sur une grande portée :

1. Il peut ne pas y avoir un chevauchement suffisant entre les deux empreintes à comparer. Cela est particulièrement présent pour les capteurs de petite surface.
2. Il y a une translation relative, une rotation et des déformations non-linéaires des minuties dans l'image d'entrée par rapport au descripteur (modèle de réference).
3. Quelques minuties peuvent manquer dans l'image d'entrée et de fausses minuties peuvent apparaître.
4. Il peut y avoir des erreurs de localisation (position ou orientation) par l'extracteur, particulièrement sur les images de très mauvaise qualité.

Le processus de détection automatique des minuties est un processus extrêmement critique, particulièrement pour les empreintes de mauvaise qualité. Le processus traditionnel d'extraction des minuties suit les étapes suivantes tel qu'illustré sur la figure 1.8 :

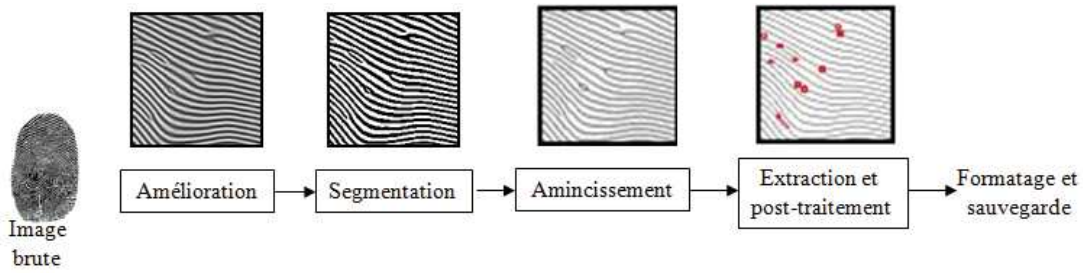


FIGURE 1.8 – Processus usuel d'extraction des minuties

– **Amélioration (image enhancement)** : le but de cette étape est d'améliorer la qualité des régions récupérables dans l'image. Les procédures d'amélioration orientées pixel (comme la légalisation d'histogramme, la normalisation, le filtrage ou l'adoucissement des frontières) améliorent la lisibilité de l'empreinte mais ne sont pas suffisantes pour traiter ce type d'images car ils n'agissent pas sur la structure globale des crêtes.

  En effet, le bruit dans une image d'empreintes s'exprime par une cassure dans le flux directionnel des crêtes. Généralement, une image d'empreinte digitale contient les trois catégories de régions suivantes (figure 1.9) : *(i)* région bien définie (les crêtes et les vallées sont visibles pour une extraction possible), *(ii)* région récupérable (les crêtes et les vallées sont corrompues mais un algorithme d'amélioration peut les récupérer), *(iii)* région irrécupérable (les régions sont très touchées par le bruit). Un algorithme d'amélioration a comme but de récupérer la région d'intérêt

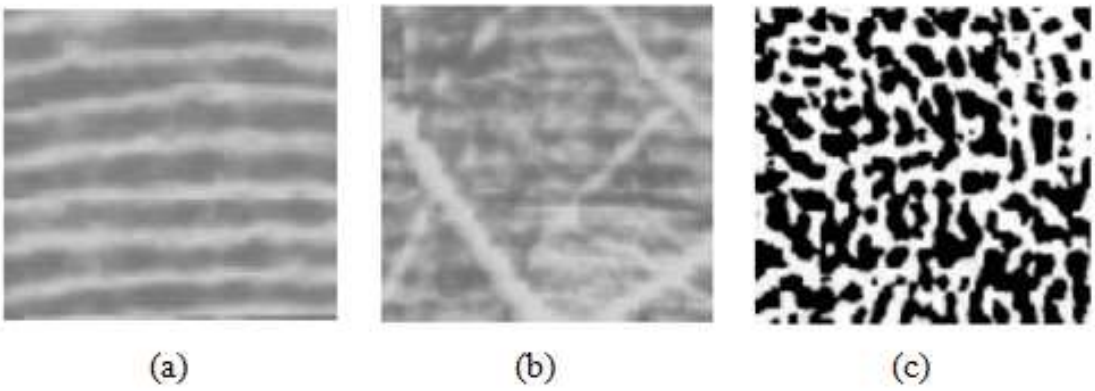


FIGURE 1.9 – Différentes qualités de régions sur une empreinte digitale

et de l'améliorer et de masquer la région irrécupérable. Pour ce faire, les meilleurs résultats ont été obtenus en utilisant des filtres contextuels où les paramètres du filtre

sont localement adaptés. Ces paramètres dépendent essentiellement de la fréquence et de la direction locales des crêtes. Un exemple de ces filtres sont les filtres de Gabor [HWJ98] ou les filtres anisotropes [LPY+12]. Ils opèrent comme des filtres passe-bandes, en augmentant le contraste entre crêtes et vallées (filtrage de différentiation) dans la direction normale à l'orientation des crêtes tout en effectuant un adoucissement dans la direction des crêtes (filtrage passe-bas) pour combler les impuretés et lier les trous.

– **Segmentation** : l'image en niveaux de gris est convertie en image binaire pour distinguer les crêtes des vallées. A cause de son caractère non stationnaire, une binarisation adaptative est souvent préférée. Le seuil de binarisation T est déterminé localement en considérant les propriétés du voisinage local [JHPB97, RCJ95, DTL98]. Généralement, cette étape de binarisation fournit de bons résultats à condition qu'elle soit appliquée à des images de bonne qualité ou après une phase d'amélioration. La figure 1.10 montre un résultat de binarisation sans étape d'amélioration. Nous remarquons sur cette figure que la binarisation n'a pas réussi à distinguer les crêtes humides qui ont tendance à se raccorder, ni a éliminer les discontinuités dans les crêtes sèches. Ces problèmes vont ensuite se propager dans le reste du processus.

– **Amincissement** : l'image binaire, en utilisant des opérateurs morphologiques, est soumise à une étape d'amincissement (l'épaisseur des lignes de crêtes est réduite à un pixel). Quelques algorithmes comme le MINDTCT [GWMW02], développé par le NIST (National Institute of Standards end Technology) pour le FBI (Federal Bureau of Investigation), ne requièrent pas cette étape.

– **Extraction et post-traitement** : Un simple calcul du nombre de connexions d'un pixel crête sur l'image amincie peut informer si le pixel concerné est une minutie ou bien non. Un post-traitement s'avère toujours utile pour éliminer les fausses alarmes. Celui-ci est généralement basé sur le choix d'heuristiques structurelles.

– **Formatage et sauvegarde** : les minuties requièrent une représentation très compacte qui dépasse rarement le $1KO$. Chaque minutie peut être décrite par un nombre d'attributs telles que la position (x, y), l'orientation et d'autres informations susceptibles d'aider à l'appariement comme son type. Cependant, la plupart des algorithmes considèrent seulement sa position et son orientation (figure 1.11).
Même si elle n'est pas toujours nécessaire, l'interopérabilité entre différents systèmes de reconnaissance peut s'avérer importante. Cela concerne principalement

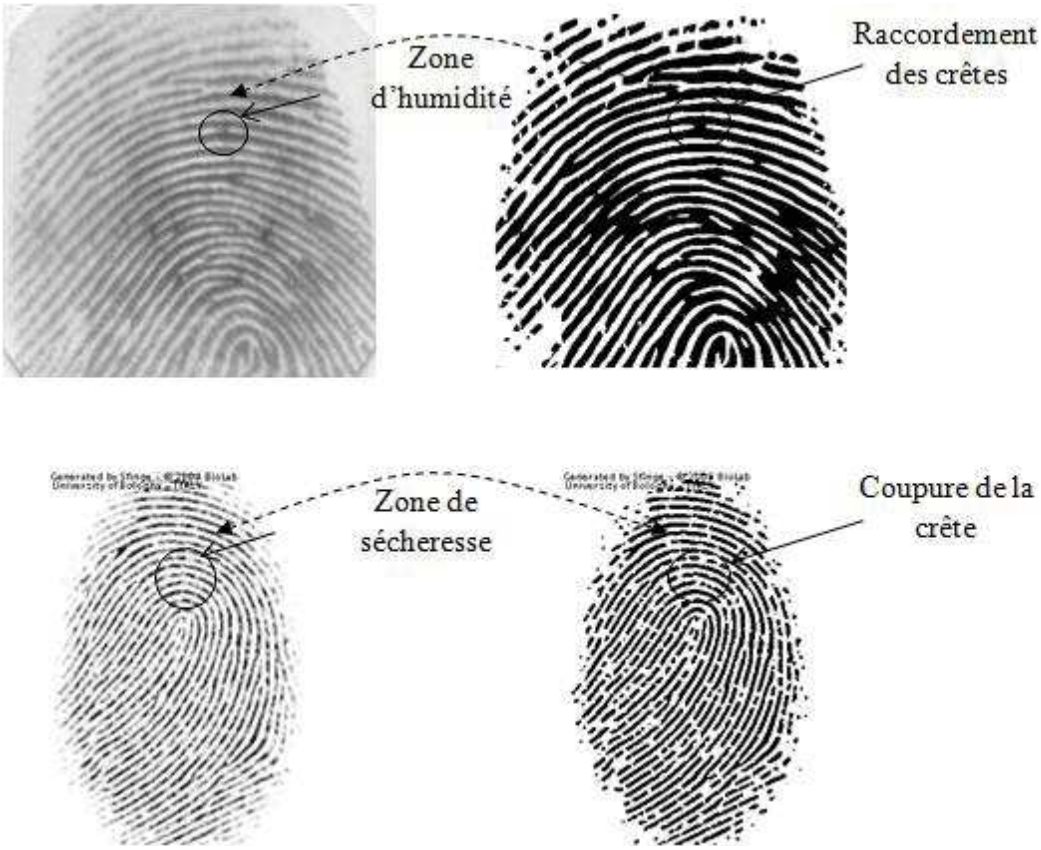


FIGURE 1.10 – Problèmes de binarisation sur des empreintes sèches ou humides [Bel06]

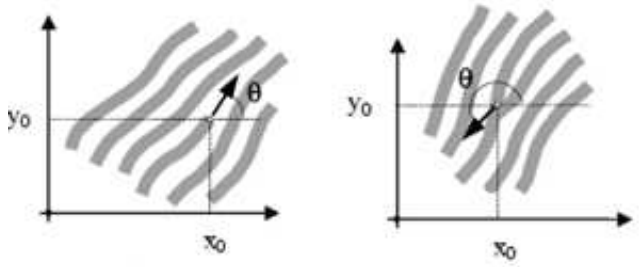


FIGURE 1.11 – Les caractéristiques principales des minuties

les applications de gestion d'identité à grande échelle comme les programmes de surveillance aux frontières où plusieurs sous-modules ont besoin de communiquer. Il est donc important de définir des standards pour spécifier un système de coordonnées unique ainsi que pour le formatage des minuties. Cela permettra principalement l'échange d'informations entre les modules d'extraction et de comparaison. Comme standard pour la représentation des minuties, nous citons : ANSI/NIST-ITL [MN07], INCITS 378-2004 [AI08], ISO/IEC 19794-2 [ISO08].

Les descripteurs de crêtes (ridge-based representation) Lorsque la surface de l'empreinte devient petite ou que la qualité de l'image en entrée devient trop endommagée, le nombre de minuties ne peut pas être suffisant pour effectuer l'appariement. Pour pallier ce problème, certains travaux [JPHP00] s'intéressent à extraire d'autres caractéristiques du motif des crêtes, comme : l'orientation locale, la fréquence locale, la forme de la crête ou l'information de texture. Cet appariement devient dans ce cas plus approprié même si ces performances ne sont pas aussi bonnes que l'appariement basé sur les minuties.

Phase de comparaison

La mise en correspondance entre deux images d'empreintes diffère suivant la représentation sélectionnée : image, minuties ou descripteur de crêtes. Pour ses nombreux avantages, l'appariement basé minuties reste la méthode la plus populaire (distinctive et réduite en taille principalement). Néanmoins, cette tâche n'est pas aussi simple à cause des différences qui puissent exister entre deux empreintes du même doigt, séparément acquises. La figure 1.12 illustre entre autres les problèmes de non-chevauchement et de distorsions géométriques. La prise en charge du problème d'alignement est une tâche obligatoire. Il s'agit de trouver les inconnues : $\triangle x, \triangle y, \theta$ et s, pour combler les problèmes de translation, de rotation et de changement d'échelle.

Une recherche gourmande en considérant toutes les correspondances possibles est de complexité factorielle. Pour réduire cette complexité, les algorithmes d'appariement adoptent différentes heuristiques. La figure 1.13 résume globalement les approches d'appariement existantes dans la littérature. Il existe deux grandes familles de méthodes d'appariement : **l'appariement global** et **l'appariement local**.

L'appariement global nécessite de retrouver explicitement les paramètres de transformation (translation, rotation et changement d'échelle). L'approche par la transformée de Hough [RKCJ96] est la méthode la plus représentative de l'appariement global. Il s'agit d'estimer les paramètres d'alignement durant le processus d'appariement en discrétisant l'espace de recherche. Ces algorithmes, **sans pré-alignement explicite**, sont fiables en terme de performance mais leur complexité de calcul est assez importante. L'idée est donc d'opérer un pré-alignement avant l'appariement effectif où les paramètres de ce pré-alignement sont sauvegardés dans la base de données en même temps que le modèle. Le pré-alignement nécessite l'ajout d'autres caractéristiques que les minuties. Il s'agit d'*information supplémentaire* qui va aider à faire cette phase d'enregistrement des minuties *(minutiae registration)*.

FIGURE 1.12 – Différences entres empreintes (chaque ligne montre une paire d'empreinte du même doigt)

Dans **l'approche absolue**, le pré-alignement est opéré pour chaque image indépendamment des autres. La méthode M82 du FBI est l'approche la plus populaire de cette technique. Plus robuste que l'approche précédente, **l'alignement relatif**, d'une image en entrée I se fait en fonction de chaque modèle T dans la base de données. Ce pré-alignement repose généralement sur la détection des singularités comme le point core [ZW02]. Néanmoins, il existe d'autres approches qui utilisent l'image d'orientation [YA05] ou l'image des crêtes [JHPB97].

Dans **l'appariement local**, il s'agit de comparer deux empreintes suivant la structure locale des minuties [TK03, CFM10]. Les structures locales sont caractérisées par des

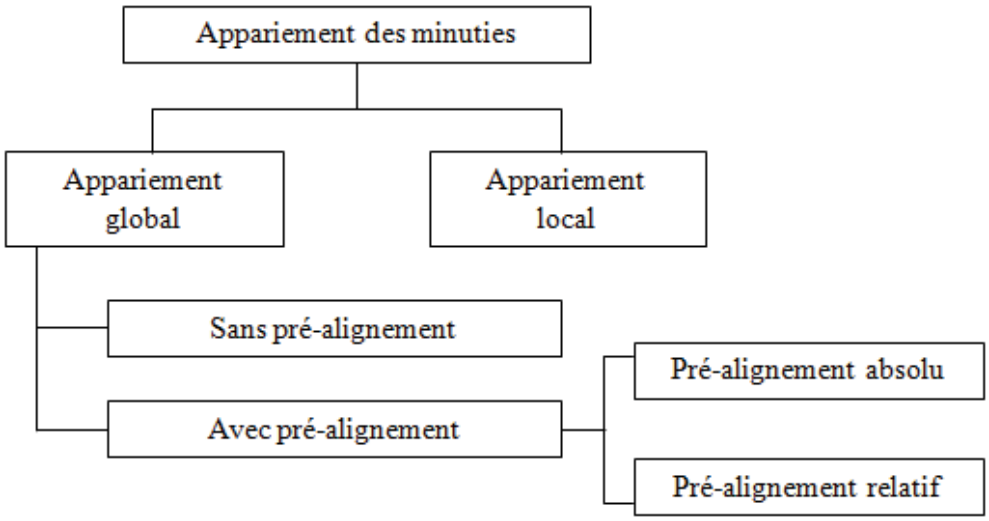


FIGURE 1.13 – Classification des méthodes d'appariement

attributs invariants ce qui élimine l'étape de pré-alignement. Ces approches sont donc plus rapides, plus robustes aux distorsions que les approches globales mais généralement moins distinctives (car elles relâchent les relations spatiales entre les minuties sur le plan global qui sont extrêmement discriminantes). Des pistes intéressantes reposent sur des stratégies hybrides où l'appariement local est consolidé par la prise en compte d'informations sur le plan global [Fen08]. Pour une lecture plus ample du problème d'appariement, le lecteur pour se référer au livre de Maltoni *et al.* [MMJP09].

1.1.5 Evaluation des systèmes d'authentification biométrique

Avec l'utilisation de plus en plus répondue de la biométrie, le besoin d'évaluer et de comparer les différents systèmes entre eux devient impératif. Malheureusement, jusqu'à présent, il n'existe pas encore de méthode systématique et standardisée pour certifier et garantir la fiabilité des systèmes biométriques. Les points à évaluer se présentent généralement sous trois critères :

1. L'évaluation de la performance du système, qui mesure les taux d'erreur du système ainsi que son efficacité.
2. L'évaluation de la sécurité et du degré de préservation de la vie privée, qui mesure la robustesse du système aux différentes attaques.
3. L'évaluation de l'usage, qui mesure l'acceptabilité et le taux de satisfaction des utilisateurs.

Dans cette section, on ne s'intéressera pas à l'évaluation de l'usage. Pour plus de détails, le lecteur peut se rapporter à [EA11].

Evaluation de performance des systèmes d'authentification biométriques

Il existe dans la littérature de nombreuses métriques pour quantifier la performance du système. On ne s'intéressera dans cette section qu'aux mesures des taux d'erreur et aux courbes de performance. Pour plus de détails, le standard ISO/IEC 19795 [ISO06a] est entièrement consacré à la prise en charge du problème d'évaluation de performance.

- **Les mesures des taux d'erreur**
  - *Le taux d'échec à la capture* (*Failure to Acquire Rate, FTA*) qui est la proportion des tentatives de captures pour lesquelles le système ne peut pas détecter un échantillon biométrique.
  - *Le taux d'échec à l'enrôlement* (*Failure To Enroll Rate, FTER*) qui mesure la proportion des individus pour lesquels le système ne peut pas créer de modèle biométrique.
  - *La fausse acceptation* (*False Acceptance, FA*) lorsque le système déclare l'individu comme étant légitime alors que c'est un imposteur.
  - *Le faux rejet* (*False Rejection, FR*) lorsque le système refuse un individu alors qu'il s'agit d'un utilisateur légitime.
  - *Le taux des fausses acceptations* (*False Acceptance Rate, FAR*) qui mesure la proportion des fausses acceptations par rapport au nombre total des transactions imposteurs.
  - *Le taux des faux rejets* (*False Rejection Rate, FRR*) qui mesure la proportion des faux rejets par rapport au nombre total des transactions légitimes.
  - *Le taux d'égale erreur* (*Equal Error Rate, ERR*) qui indique le taux d'erreur lorsque le système est configuré de manière à avoir le FAR égal au FRR.
  - *Le Zéro FRR* qui est défini comme le plus faible FAR lorsqu'aucun faux rejet ne survienne.
  - *Le Zéro FAR* qui est défini comme le plus faible FRR lorsqu'aucune fausse acceptation ne survienne.

Pour qualifier la fiabilité d'un système biométrique, l'EER est généralement le plus utilisé. Plus il est faible, plus le système est performant. Néanmoins, il est tout aussi intéressant de considérer le Zéro FAR qui, en général, est plus intéressant pour les cas pratiques.

- **Les courbes de performance**
  - *La distribution intra-classe/inter-classe* : pour évaluer la performance d'un système de vérification, on doit calculer les scores à partir d'un large nombre de comparaisons entre des gabarits d'un même sujet. On obtient alors *la distribution intra-classe* (*genuine distribution*). Il faut aussi collecter les scores des com-

paraisons entre des gabarits appartenant à des sujets différents pour obtenir *la distribution inter-classe* (*impostor distribution*). La distribution intra-classe/inter-classe se présente comme sur la figure 1.14. Le seuil de décision du système est ensuite choisi parmi les scores possibles suivant le niveau de sécurité souhaité.

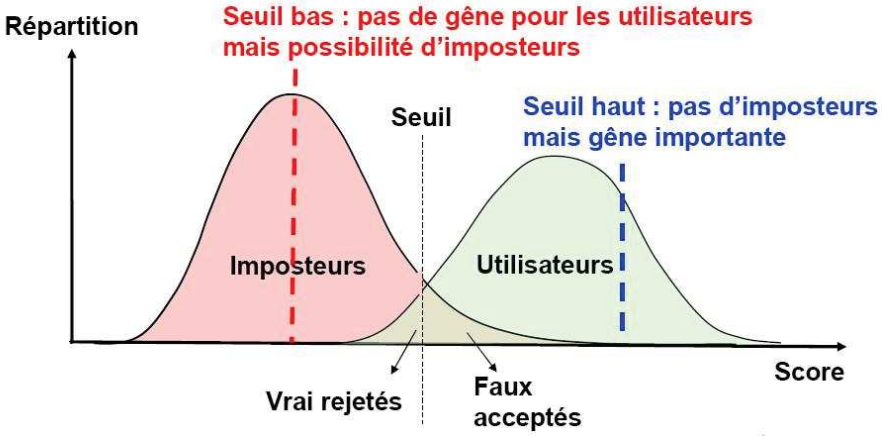


FIGURE 1.14 – La distribution inter-classe/intra-classe

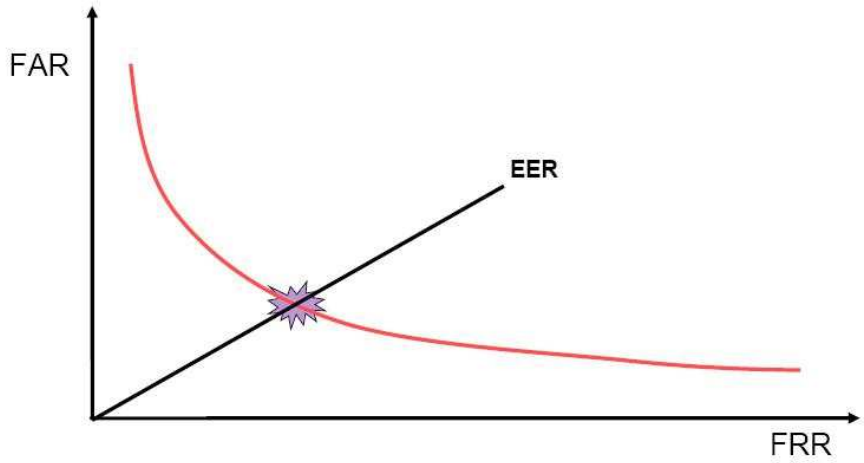


FIGURE 1.15 – Exemple de la courbe *DET*

- *La courbe réceptrice des caractéristiques* (ou *la courbe ROC*) qui est la plus couramment utilisée pour représenter les performances du système. Elle représente l'évolution du FAR en fonction du FRR suivant les différents seuils de décision possibles. Au lieu de *ROC*, parfois le terme *DET* (détection d'erreur Tradeoff) est utilisé. Dans ce cas, le terme *ROC* est réservé pour représenter les taux de vrais

rejets ($1 - FRR$) contre les taux de fausses acceptations (FAR). Cette courbe est utile pour donner une représentation globale sur le comportement du système. Un exemple de cette courbe est donné sur la figure 1.15.

Evaluation de la sécurité des systèmes biométriques

Il est bien connu que la méthodologie principale pour évaluer la sécurité dans les systèmes d'information est basée sur la notion de critères communs **CC**. Ces critères communs font l'objet d'une standardisation ISO 15408 [ISO09]. A l'état actuel des choses, ces critères ne suffisent pas pour évaluer complètement les systèmes biométriques. Ils doivent, avant tout, prendre en compte les particularités de ces systèmes. Récemment, un groupe de travail a suggéré une méthode d'évaluation basée sur les critères communs [Gro02]. Pour ce faire, deux points importants par rapport auxquels un système biométrique a besoin d'une considération spéciale ont été identifiés :

- L'analyse des vulnérabilités.
- Le test de performance.

Un modèle d'attaques possibles basé sur 15 points de vulnérabilités a été identifié. Les auteurs concluent qu'un système biométrique peut être évalué sous les mêmes CCs que n'importe quel autre IT système tout en prenant en compte des objectifs de sécurité (target security) spécifiques au domaine biométrique. Entre autres, cela concerne la protection des données de l'utilisateur et le respect de sa vie privée.

Une autre proposition sur la standardisation de l'évaluation de la sécurité dans les systèmes biométriques a été adressée par la norme internationale ISO/IEC FCD 19792 [ISO06b]. Le rapport présente une vue d'ensemble des vulnérabilités possibles. Il inclut aussi la phase de test du respect de la vie privée avec le processus d'évaluation.

Sur le plan académique, plusieurs travaux concernant la modélisation des attaques existent. Un travail pionnier a été présenté en 2001 par Ratha *et al.* [RCB01]. Ils identifient les points de vulnérabilités sur chaque module dans l'architecture générique du système d'authentification. La figure 1.16 illustre ces points d'attaque possibles.

I. Buhan dans sa thèse [Buh08], modélise les attaques possibles par un arbre à trois niveaux en se basant sur les 17 points de vulnérabilités identifiés dans les travaux de Bolle *et al.* [BCP+03]. Récemment, Henniger *et al.* [HSK10] présentent un modèle d'évaluation de la sécurité dans un système de vérification d'empreintes digitales en se basant sur les critères communs mais en développant plus en détails les niveaux de vulnérabilités.

Jain *et al.* [JNN08] utilisent une représentation par arête de poisson (a fishbone model) pour modéliser les points de vulnérabilités dans les systèmes biométriques en se basant sur deux types d'attaques : les attaques à zéro effort et les attaques adverses. Dans les attaques adverses, l'abus peut être causé lorsque l'infrastructure du système biométrique n'est

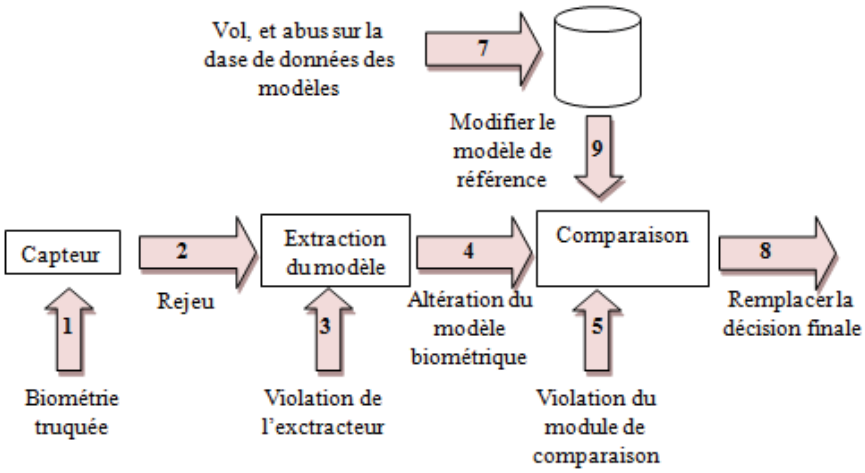


FIGURE 1.16 – Les points de vulnérabilités d'un un système biométrique suivant le modèle de Ratha *et al.* [RCB01]

pas sécurisée (matériel, logiciel ou canaux de communication). Maltoni *et al.* [MMJP09] adoptent une méthode légèrement différente, en se posant la question pratique sur la façon dont la défaillance pourrait être déclenchée ? Ils identifient principalement deux types de défaillance : le déni de service et l'intrusion. Dans le déni de service, l'utilisateur légitime est refusé par le système. Par contre, l'intrusion se réfère à un accès illégitime au système. Pour des raisons de sécurité, ils concentrent leur attention sur le risque d'intrusion dans lequel un attaquant doit d'abord obtenir les données biométriques puis essayer de les injecter dans le système biométrique. En se focalisant sur l'empreinte digitale, les auteurs exposent des procédés d'obtention des données d'empreintes digitales et des procédés pour les injecter dans le système.

Pour leur généralité, nous détaillons les points d'attaques de Ratha *et al.* [RCB01] illustrées sur la figure 1.16 :

- ***Point 1 :*** Attaque sur le capteur en présentant une modalité biométrique truquée. Par exemple, Matsumoto *et al.* [MMYH02] attaquent 11 différents systèmes de vérification d'empreintes digitales avec des moulages de doigts en gélatine. Une empreinte résiduelle sur du verre est utilisée pour créer le doigt artificiel. Tous les 11 systèmes ont accepté ce doigt avec une probabilité de 67%. En 2008, Galbally *et al.* [GCL+08] démontrent une autre attaque très nuisible. L'attaque est effectuée en présentant au capteur un doigt truqué généré à partir d'une image d'empreinte, qui elle, a été reconstruite à partir du modèle des minuties, volé de la base de données. Cette menace met en avant l'importance de protéger les données biométriques à l'intérieur des bases de données.
- ***Points 2,4,6,8 :*** Attaque sur les canaux de communication. Si aucune mesure de sécurité n'est prise en charge, l'attaquant peut intercepter (Eavesdropping), modifier

et insérer les données biométriques (Man-in-the-Middle) ou rejouer les mêmes données biométriques (Replay). Il peut aussi manipuler la décision issue du module de comparaison en la compromettant.

– ***Points 3,5 :*** Attaque sur les modules de traitement. Un des plus grands risques de sécurité informatique concerne l'injection de programmes malicieux qui peuvent ensuite contrôler le comportement du module initial (comme fournir le modèle ou le score souhaités).
– ***Point 7 :*** Attaque sur les modèles. L'attaquant peut capturer le modèle de référence, le substituer, le modifier et ainsi il peut compromettre la base de données.

Discussion

L'attaque sur le modèle biométrique peut être très préjudiciable car elle implique l'exposition d'une information personnelle et sensible ainsi que le vol d'identité. Dans la section qui suit, nous étudions l'importance de sécuriser le modèle biométrique. Nous présentons ses points de vulnérabilités et les menaces qui peuvent en découler aussi bien concernant la violation de la vie privée que les risques d'intrusion.

1.2 Modèle biométrique : vulnérabilités et menaces

Nous exposons dans cette section les éléments concernant l'exposition du modèle biométrique face aux menaces qui concernent aussi bien les risques de violation de la vie privée que les risques d'usurpation d'identité. Les niveaux de vulnérabilités deviennent plus élevés lorsqu'il s'agit de systèmes biométriques à grande échelle, de bases de données centralisées et de biométrie à distance.

1.2.1 Les risques de violation de la vie privée

D'après Solove [Sol08], il est assez complexe de définir la notion de vie privée. Nous formalisons une définition dans laquelle la vie privée est relative à la protection des individus à l'égard du traitement des données à caractère personnel et sensible. En se référant à quelques juridictions comme la directive européenne de protection des données (Dir95/46/EU), nous nous basons sur les définitions suivantes, qui concernent les données à caractère personnel :

Définition 1. *Les données personnelles sont toutes les informations relatives à une personne physique identifiable.*

Définition 2. *Une personne identifiable est une personne qui peut être identifiée, directement ou indirectement, en particulier par référence à un numéro d'identification ou à un ou plusieurs facteurs spécifiques à son identité physique, physiologique, psychique, économique, culturelle ou sociale.*

Le modèle biométrique, sous toutes ces formes, comme le modèle des minuties est une donnée personnelle car elle identifie son propriétaire. Par exemple, dans [PPJ02], les auteurs estiment la probabilité de faire correspondre deux empreintes digitales différentes, à partir des minuties, à $5.5x10^{-59}$. Cette probabilité très basse montre que les minuties peuvent identifier les personnes de façon unique. De plus, le déploiement des systèmes biométriques pour reconnaitre les personnes appuie le fait que le modèle biométrique est une donnée personnelle. En outre, les données biométriques sont généralement considérées comme sensibles impliquant des menaces sur la vie privée. Nous reprenons maintenant ces risques en se focalisant sur la modalité empreinte digitale :

- La donnée biométrique (en particulier basée image) peut exposer des informations sensibles telles que des informations sur la santé, l'origine raciale ou ethnique des individus. Cette information peut ensuite servir comme base à une discrimination injustifiée [MM08].
- Comme indiqué par Schneier [Sch99], la donnée biométrique est un identificateur non secret. A titre d'exemple, l'empreinte digitale est une modalité à trace. Par conséquent, la collecte et l'utilisation des données biométriques sans le consentement de la personne concernée rend cette information très sensible.
- La représentation compacte des données biométriques sous forme de modèle et non sous son aspect brute constitue aussi une information sensible. Récemment, plusieurs travaux de recherche ont montré la reconstruction du signal initial à partir du modèle de référence [CLMM07b, CLMM07a, FJ09].
- Le caractère unique des caractéristiques biométriques permet à une personne malveillante de relier un même utilisateur inscrit sur différentes bases de données permettant ainsi le suivi et le profilage des individus. Par exemple, le test MINEX'04 [2] confirme la possibilité de croiser les bases de données stockant les modèles des minuties. En effet, à cause du caractère interopérable de certaines bases de données et de leurs architectures orientées service, il suffirait de mettre le modèle biométrique sous le format spécifié et de le soumettre à la base de données en question. A titre indicatif, des programmes tels que US-VISIT pour le contrôle des frontières aux Etats-unis ou le système européen EURODAC tentent d'interfacer plusieurs bases de données entre elles ce qui génère une véritable crainte de la communauté civile.

2. http ://www.nist.gov/itl/iad/ig/minex04.cfm

Cette même frayeur est aussi à l'origine de l'opposition des usagers contre plusieurs projets gouvernementaux de conservation des données biométriques comme démontré dans [Dom04].

– Le détournement d'usage [MM08]) est un autre risque de violation de vie privée. Par exemple, une application initialement prévue pour un usage spécifique peut progressivement être étendue pour un autre usage sans le consentement de la personne.
– En cas d'abus ou de compromission de la base de données, à cause de son unicité, le modèle biométrique ne peut pas être révoqué et réémis ce qui rend cet identifiant très sensible.

1.2.2 Les risques d'usurpation d'identité

Il a été démontré par différents travaux, qu'à partir d'un modèle biométrique compromis, un agent attaquant (qu'il soit interne ou externe) pouvait s'introduire dans le système biométrique en se faisant passer pour l'utilisateur légitime. En 2007, Ross *et al.* [RSJ07] montrent que trois types d'information sur l'empreinte d'origine pouvaient être déduits des minuties : l'image d'orientation, la classe de l'empreinte et la structure des crêtes. En 2008, Galbally *et al.* [GCL+08] montrent qu'à partir du modèle des minuties, une reconstruction de l'image d'origine était possible. Il a été de plus possible de fabriquer un moulage à partir de cette image, présenté au capteur et accepté par le système.

L'*attaque par mascarade* [CLMM07b, CLMM07a, FJ09] qui consiste à contourner le capteur et à envoyer directement au module d'extraction une image contrefaite est un autre type de menace, généralement, plus pernicieuse que la précédente (car elle ne nécessite pas la présence physique de l'attaquant). Par exemple, Cappelli *et al.* [CLMM07b] utilisent une approche de reconstruction d'images à partir du modèle des minuties compatible ISO/IEC 19794. Ils obtiennent 81% de succès d'attaques contre 9 systèmes d'empreintes digitales.
L'*attaque du FAR* (False Acceptance Rate attack) ou l'attaque à zéro effort est un autre risque d'usurpation d'identité. En faisant une présentation exhaustive (force brute) d'un large nombre d'entrées biométriques, il pourrait y avoir un exemplaire susceptible de passer la borne de vérification. Supposons un comparateur d'empreintes digitales fonctionnant à un FAR de $0,001\%$ alors il faudrait, en moyenne, 100.000 injections pour réussir l'usurpation. Pour éviter cette complexité, une attaque de type hill-climbing qui envoie des modèles aléatoires au système, en les perturbant de façon itérative est proposée dans [UJ04]. A condition d'avoir le score de comparaison de chaque itération, les auteurs ont pu contourner le système qui opérait avec un FAR à 0.1%, à chaque fois en faisant moins de 1000 tentatives.

Discussion

Nous avons souligné les menaces potentielles dans l'utilisation des modèles biométriques sans méthode de protection. La biométrie fait intervenir une information personnelle hautement sensible sans que le système ne puisse assurer son intégrité (biométrie falsifiable), ou sa confidentialité (donnée non secrète).
La sécurité du modèle biométrique est l'une des questions les plus cruciales dans la conception d'un système biométrique sécurisé. C'est une tâche qui exige une attention impérative et rigoureuse. Au niveau international, le déploiement des systèmes biométriques est régi par des juridictions et des règlementations associées. Une question clé traitée par ces organismes est en relation avec le stockage des modèles de référence sur les bases de données centrales. Par exemple, la CNIL (Commission Nationale Informatique et Libertés) en France, proscrit la création de ces bases de données en particulier en ce qui concerne les modalités avec traces comme c'est le cas pour les empreintes digitales. A côté de ces guides de bonne conduite, de nouveaux paradigmes de protection des données biométriques, comme les Pets (Privacy Enhancing Rechniques), apparaissent. Dans la suite, nous résumons ces principales approches de protection du modèle biométrique.

1.3 Sécuriser le modèle biométrique

Dans cette section, nous nous intéressons aux approches existantes pour protéger le modèle biométrique. Nous définissons la notion de protection des données biométriques par la capacité de contrôler l'accès impropre, tout en évitant les risques d'intrusion. Nous catégorisons les mesures de sécurisation du modèle biométrique dans deux différentes classes : *(i)* les architectures à système fermé et *(ii)* les techniques d'amélioration de la vie privée (ou PETs pour Privacy Enhancing Techniques). Il s'agit donc de deux types de solutions, l'une au niveau architectural et système et l'autre au niveau algorithmique ou fonctionnel, et qui peuvent être utilisées séparément ou conjointement.

1.3.1 Les architectures à système fermé

Il s'agit d'assurer le stockage sécurisé du modèle biométrique sur un dispositif dédié (secure element) comme une carte à puce. Différentes solutions peuvent être proposées :

- Store-on-Card (SoC) : il s'agit d'éliminer la base de données centrale en stockant le modèle biométrique sur le dispositif sécurisé.
- Match-on-Card (MoC) : se réfère aux solutions où le module de comparaison est sur l'élément sécurisé. Le capteur et le module d'extraction sont sur une plateforme hôte.

– System-on-Device (SoD) : le capteur, les modules d'extraction et de comparaison sont embarqués sur le même dispositif.

L'intégration de la biométrie sur un élément sécurisé est une piste prometteuse [SR01, GB07] pour sécuriser l'information biométrique. Cependant, en pratique, le plus grand effort accompli concerne uniquement le stockage du modèle à l'intérieur du dispositif. Il est ensuite extrait du dispositif pour accomplir la vérification sur une station hôte (e.g. le passeport électronique) ce qui engendre les mêmes problèmes que les architectures classiques. Les systèmes *MoC* sont en principe plus sécurisés. Malheureusement, à cause des performances réduites des puces par rapport aux processeurs modernes (*24 MHz vs 3.4 GHz*), la complexité des algorithmes implantés peut être considérablement réduite résultant ainsi en une chute des performances du système biométrique. Aujourd'hui, implémenter un système *MoC* fiable constitue un vrai défi. D'autre part, supposer que ces jetons sont inviolables n'est pas toujours vrai.
La sécurité de tels dispositifs est souvent évaluée par un niveau de certification (par exemple EAL4) donnant quelques éléments sur la possibilité pour qu'un pirate puisse atteindre les informations stockées. Suivant son degré de résistance (tamper resistant element), des attaques physiques peuvent être effectuées. Ainsi, il devient possible, même si elle est rare, que le modèle puisse être atteint à partir d'un jeton volé.

1.3.2 Les techniques d'amélioration de la protection de la vie privée

Les techniques d'amélioration de la protection de la vie privée sont les approches qui conceptualisent la notion de vie privée au même temps que la désignation du système biométrique. Il s'agit, grossièrement, d'embarquer la préservation de la vie privée au niveau de la technologie biométrique. Cette approche globale de consolidation de la vie privée, nommée *Privacy by Design*, a été initiée par les travaux d'Ann Cavoukian [Cav08, Cav99], et a acquis une large reconnaissance internationale depuis son adoption en 2010 par les commissaires à la vie privée [Com10]. Nous citons dans ce qui suit les principales approches biométriques de protection de vie privée pour assurer certaines exigences, comme celles de confidentialité :

Chiffrement du modèle biométrique

En se basant sur des mécanismes de cryptographie, l'ANSI (American National Standards Institute) a proposé le standard X9.84 [Sta02] comme moyen de gérer l'information biométrique. Les spécifications ANSI X9.84 ont été désignées pour maintenir la confiden-

tialité et l'intégrité du modèle biométrique. Cependant, même si la cryptographie a prouvé son efficacité pour sécuriser le stockage et la transmission de l'information, elle devient inadéquate lorsqu'il s'agit de biométrie. En effet, à cause de la variabilité intra-classe du signal biométrique, la comparaison devrait se faire dans l'espace en clair ce qui implique qu'un attaquant puisse toujours essayer d'avoir le contrôle sur la donnée biométrique. La plupart des risques de violation de vie privée demeure problématique.

Les bases de données anonymes

L'objectif dans les bases de données anonymes est de vérifier le statut d'appartenance d'un utilisateur à une base de données sans connaitre son identité. Une question clé dans les bases de données anonymes est d'assurer une collaboration sécurisée entre les deux parties que sont le fournisseur du service et son utilisateur. L'étude formelle d'un tel problème est en relation avec le calcul multi-partie sécurisé ou Secure Multiparty Computation (SMC). Pour le domaine biométrique, la réalisation d'un protocole SMC se base généralement sur la cryptographie homomorphique. La propriété fondamentale de la cryptographie homomorphique est de permettre de réaliser des opérations simples sur des données chiffrées telles que l'addition ou la multiplication. Par exemple, la primitive RSA est un homomorphisme par rapport à la multiplication, le chiffré d'un produit est égal au produit des chiffrés :

$$Enc(m,k) \times Enc(\acute{m},k) = Enc(m \times \acute{m},k) \tag{1.3}$$

Le principal inconvénient à l'application des protocoles SMC pour la biométrie est leur grande complexité de calcul. D'après nos recherches, le premier travail de bases de données anonymes dont l'accès se basant sur la biométrie a été présenté dans [YLZC09]. Le but est de permettre au serveur de connaitre l'appartenance ou non du client à la base de données sans d'autres informations supplémentaires que cela soit son identité ou sa biométrie en claire (non chiffrée). Les auteurs utilisent la modalité d'iris combinée au système homomorphique de Paillier. Dans [BBC+10], le même principe de la cryptographie homomorphique à clé publique a été appliqué sur l'empreinte digitale où le calcul de la distance euclidienne entre le modèle en entrée et le modèle de référence se fait dans le domaine chiffré en utilisant le système de Paillier aussi. Le serveur d'authentification reçoit le modèle en entrée sous une forme chiffrée avec la clé publique de l'utilisateur. Il calcule ensuite la distance quadratique entre ce modèle chiffré et celui stocké dans la base de données et retourne le résultat sous forme chiffrée au client. Ainsi, le serveur n'a aucune connaissance sur la biométrie en entrée. L'idée donc dans les bases de données anonymes est d'isoler le maximum d'information entre le client et le serveur tout en permettant le calcul d'une fonction jointe entre ces deux entités. Néanmoins, les modèles stockés

dans la base restent en clair, il n'y a donc pas de protection de l'information mais juste une anonymisation de l'utilisateur concernant ses activités. Dans le point suivant, nous abordons les techniques orientées données qui ont comme objectif la sécurisation de la donnée biométrique à proprement dit.

Les schémas de protection du modèle biométrique (template protection schemes)

Dans ces schémas, la préservation de la vie privée est liée à la protection du modèle biométrique. Idéalement, comme défini dans plusieurs références [ISO11, MMJP09], ces algorithmes sont conçus de façon à garantir les exigences suivantes :

- **Irréversibilité (irreversibility)** : il devrait être impossible d'obtenir la référence biométrique originale à partir du modèle protégé. Cette propriété assure la confidentialité de la donnée biométrique ce qui à un impact directe sur la préservation de la vie privée.
- **Intraçabilité/Diversité (unlinkability/diversity)** : il devrait être possible de produire un très grand nombre de modèles protégés (à utiliser dans des applications différentes) à partir du même modèle non protégé. Cela permettra d'éviter la poursuite et la surveillance des utilisateurs à travers différentes bases de données.
- **Révocabilité et renouvellement (revocability and renewability)** : en cas de compromission du modèle de référence comme son vol, il devrait être possible de le révoquer et de générer une nouvelle référence, différente de la précédente, à partir du même échantillon biométrique.

L'idée dans les schémas de protection du modèle biométrique est de coder le modèle biométrique X vers la pseudo-identité $E(X)$ avant son stockage effectif. Ce terme de pseudo-identité a été initialement introduit par Breebaart *et al.* [BBGK08] dans le contexte du projet européen Turbine. C'est d'ailleurs, par ce projet, que la norme ISO/IEC 24745 [ISO11] pour la protection des données biométriques a pu être initiée. Nous illustrons le fonctionnement d'un schéma de protection, relativement aux phases d'enrôlement et de vérification par la figure 1.17.

Lors de l'enrôlement, les caractéristiques biométriques X sont supprimées lorsque leur version transformée $E(X)$ est générée et stockée sur un support approprié. Durant la vérification, en cas d'une approche de type *mode*1, la vérification est faite sans recodage des caractéristiques Y. Pour le *mode*2, une nouvelle pseudo-identité $E(y)$ est générée à partir du modèle en entrée Y. Les valeurs $E(X)$ et $E(Y)$ sont ensuite envoyées au module de comparaison pour vérifier l'identité déclarée. Dans la suite, nous appellons les pseudo-identités $E(x)$ et $E(y)$: les modèles transformés.

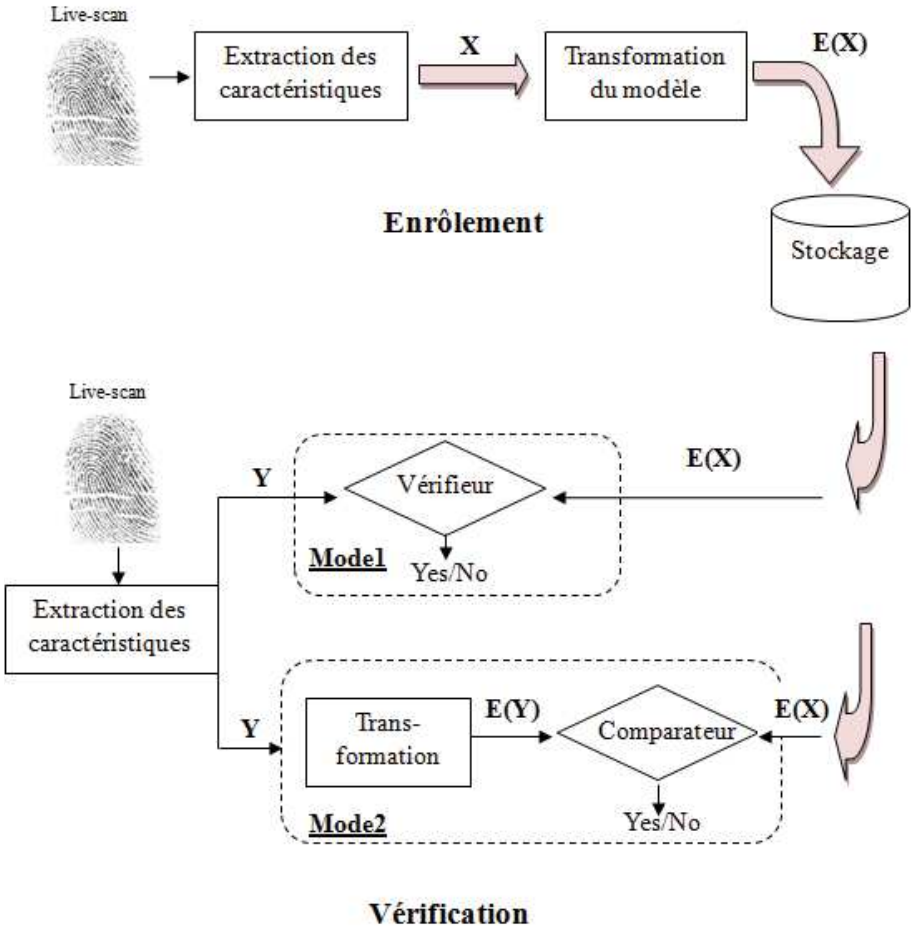


FIGURE 1.17 – Architecture de référence pour les schémas de protection du modèle biométrique. *(i). Mode1* : vérification directe de la pseudo-identité. *(ii). Mode2* : recodage de la pseudo-identité et comparaison.

Discussion

Parmi les techniques d'amélioration de la vie privée mentionnées, les schémas de protection du modèle biométrique semblent être une piste prometteuse réunissant des exigences élevées de sécurité et de protection de la vie privée. Ces techniques offrent, par rapport aux bases de données anonymes, plus de sécurité car elles assurent, par leur principe ; le fait que les données stockées ne suffisent pas à recréer le modèle biométrique d'origine. Nous nous intéressons dans la prochaine section à présenter l'état d'avancement des recherches par rapport aux techniques de protection du modèle biométrique.

1.4 Les schémas de protection du modèle biométrique

Tel que précisé par Jain *et al.* dans [JNN08], il existe principalement deux classes pour les méthodes de protection du modèle biométrique que sont : *(i)* les crypto-systèmes biométriques (plutôt le *mode1* sur la figure 1.17) et *(ii)* les approches par transformation (le *mode2* sur la figure 1.17).

1.4.1 Les crypto-systèmes biométriques

L'idée dans ces approches est de reproduire les mêmes caractéristiques que les fonctions de hachage à sens unique utilisées en cryptographie pour sécuriser le stockage des mots de passe. D'un point de vue théorie de l'information, il s'agit d'extraire une donnée reproductible et identiquement distribuée à partir d'un signal, à la base, non reproductible et distribué de manière non uniforme. La tâche de transformer une donnée bruyante en une donnée stable qui peut, *à fortiori*, être utilisée comme clé de chiffrement n'est pas évidente (notamment pour atteindre une taille raisonnable). Nous classons les primitives existantes dans trois grandes familles :

- Les primitives à base de codes correcteurs d'erreur.
- Les primitives à base d'extraction d'un secret unique.
- Les primitives à base de codes graphiques.

Nous définissons une donnée auxiliaire (ou helper data) comme une donnée garantissant qu'une chaîne unique puisse être dérivée du modèle biométrique bruité et non uniformément distribué. Une donnée auxiliaire est une information stockée dans la base de données. Cette donnée publiée en clair, doit être sécurisée dans le sens où elle ne doit pas être révélatrice du modèle d'origine. Soient les notations suivantes : $H(.)$ une fonction de hachage à sens unique, X le modèle biométrique de référence, Y le modèle en entrée, W la donnée auxiliaire, S un secret sous forme d'un nombre aléatoire attribué à l'utilisateur U. $||$ est le symbole de concaténation entre deux chaines binaires.

Les primitives à base de codes correcteurs d'erreur

Comme généralisation des méthodes existantes, Dodis *et al.* [DRS04] introduisent la primitive du *Secure Sketch* et la formalisent pour un espace de métrique H muni de la distance d. Relativement au contexte d'authentification biométrique, un secure sketch considère le problème d'un point de vue tolérance aux erreurs : un modèle $b \in H$ doit être recouvrable de n'importe quel autre modèle suffisamment proche $\grave{b} \in H$ et d'une

donnée auxiliaire W. Au même temps, la donnée auxiliaire W ne devrait pas révéler trop d'informations sur le modèle d'origine b. La figure 1.18 illustre ce fonctionnement.

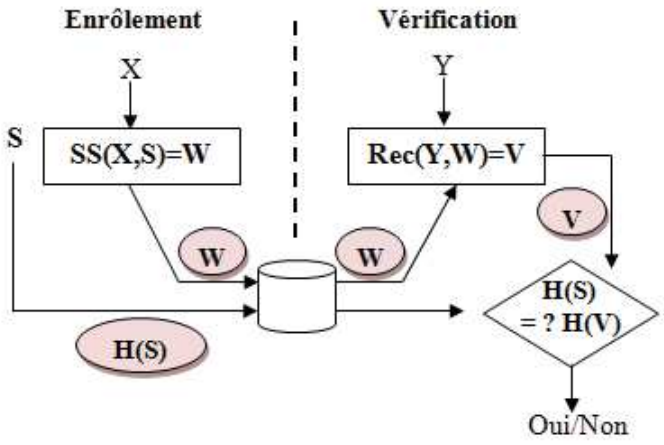


FIGURE 1.18 – Fonctionnement générique de la primitive du Secure Sketch

La définition du secure sketch est donnée comme suit :

Définition : la primitive du secure sketch est la paire des fonctions SS et Rec de telle sorte que :
- La fonction SS génère à partir de $b \in H$, une donnée publique $W \in \{0,1\}^*$. SS est souvent paramétrée par une clé aléatoire S ainsi $W = SS(b, S)$.
- La fonction Rec considère en entrée le valeur $W = SS(b, S)$ et une valeur $\grave{b} \in H$ et génère en sortie une valeur $\grave{\grave{b}}$ tel que $b = \grave{\grave{b}}$ si $distance(b, \grave{b}) \leq t$.

Le premier secure sketch a été proposé par Juels et Wattenberg dans [JW99]. Ce schéma est appelé le *fuzzy commitment*. Le concept original de cette approche fût initialement proposé par Davida *et al.* [DFM99] où la première tentative d'utilisation des **codes correcteurs d'erreurs** a été appliquée sur l'iris. Les travaux de Davida *et al.* impliquent l'extension du modèle biométrique X en un mot de code. La donnée auxiliaire W est prise comme les bits de contrôle ou le syndrome du code algébrique $[n, k, d]$ avec k la longueur de X, considéré alors comme une chaine binaire. Seulement W et le modèle scellé $H(X||W)$ sont stockés sur la base de données. Au moment de la vérification, la biométrie en entrée Y est utilisée avec le code W pour corriger le mot de code $\grave{Y} = Y||W$ et obtenir ainsi le modèle correcte $\grave{\grave{Y}}$. La valeur $\grave{\grave{Y}}||W$ est alors hachée pour être comparée à la référence $H(X||W)$ qui

se trouve sur la base de données. L'authentification est positive lorsque la comparaison réussie.

En raison de la redondance des codes correcteurs d'erreur, les bits de contrôle peuvent conduire à une divulgation d'informations (information leakage) sur les données biométriques de l'utilisateur. Comme solution possible, Juels et Wattenberg [JW99] traitent le modèle lui-même sans aucune modification comme un mot de code corrompu en introduisant une nouvelle primitive cryptographique nommée le *fuzzy commitment*. Un fuzzy commitment peut être vu comme un secure sketch sur l'espace $\{0,1\}^n$ muni de la distance de Hamming d_H utilisant un code algébrique linéaire.

Un code algébrique $[n,k,d]$ est un sous-espace vectoriel de $\{0,1\}^n$ de dimension k et composé des vecteurs x muni du poids de Hamming $w_H(x) > d$ avec $w_H(x)$ le nombre d'éléments différents de 0 dans x. La capacité de correction du code est $t = (d-1)/2$. Le principe du fuzzy commitment est alors décrit comme suit :

Primitive du fuzzy commitment

- **Durant l'enrôlement**, un mot de code $C \in \{0,1\}^n$ est calculé à partir de la clé S. Le choix de ce code dépend de la quantité d'erreur à traiter. On sauvegardera sur la base de données uniquement le couple : $(C \otimes X, H(C))$.
- **Durant la vérification**, la valeur $(C \otimes X \otimes Y)$ est calculée et corrigée pour dériver le secret $\acute{C}$. La comparaison réussie si $H(C) = H(\acute{C})$.

L'un des points faibles du fuzzy commitment est qu'il devient impraticable lorsque le taux d'erreur est assez élevé (un code correcteur $[n,k,d]$ peut corriger un maximum de $(d-1)/2$ erreurs). Il n'est donc possible de retrouver C que si $d_H(X,Y) \leq t$ avec $t = (d-1)/2$.

Parmi les applications de ce protocole sur les empreintes digitales, nous citons celle de Tuyls *et al.* [TAK+05] qui utilisent le code BCH et une méthode de sélection des éléments les plus fiables dans le vecteur X. Une autre application est celle d'Arakala *et al.* [AJH07] toujours en utilisant le code BCH mais en impliquant des descripteurs locaux sur le modèle des minuties. Malheureusement, en pratique, ces applications engendrent des taux d'erreurs assez importants. Par exemple, dans Arakala *et al.* [AJH07] nous trouvons un taux d'erreur $EER = 15\%$ sur la base de données publique FVC2000 [MMJP09].

En 2002, Juels et Sudan [JS02] modifient cette approche afin qu'elle soit utilisée pour des représentations partielles sous l'appellation du *fuzzy vault* où le principe d'interpolation polynomiale a été utilisé comme illustré sur la figure 1.19.

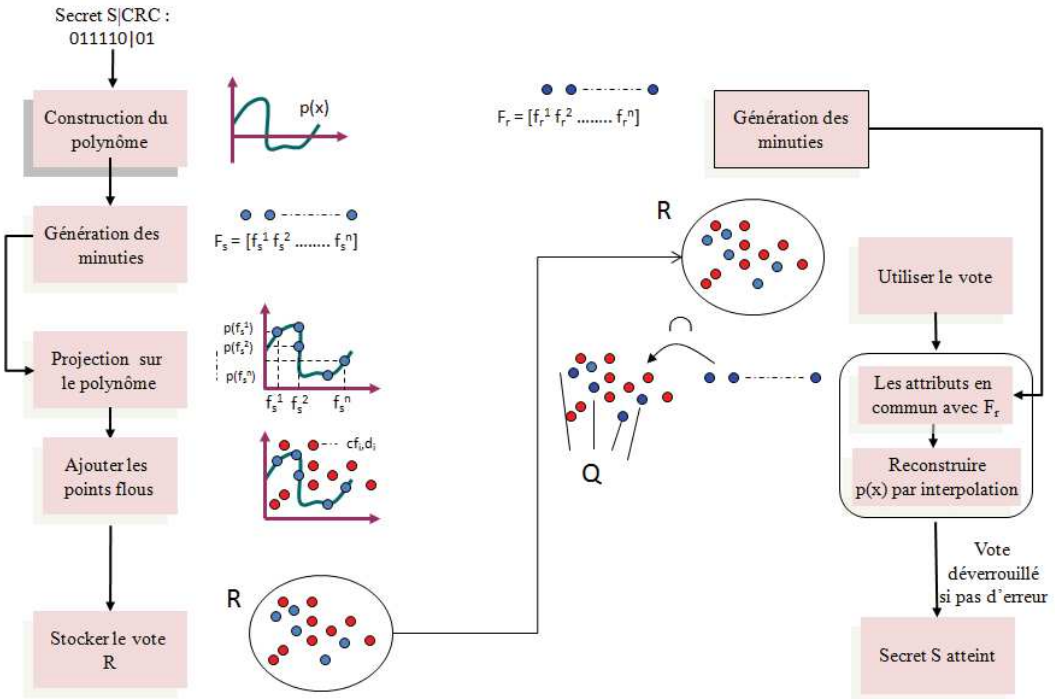


FIGURE 1.19 – Principe du fuzzy vault

Le fuzzy vault est aussi un secure sketch mais opéré sur une métrique différente qui est relative à la différence entre deux ensembles (set difference) au lieu d'une distance de Hamming. Ce schéma est potentiellement applicable aux minuties d'empreintes digitales contrairement au précédent qui ne considère en entrée que des séquences stables en ordre et en taille.

Les applications du fuzzy vault aux minuties ont été initiées dans plusieurs travaux [CKL03, QZX06, UJ06]. Parmi les implémentations les plus performantes, nous citons celle de Nandakumar *et al.* [NJP07]. Dans ce travail, plusieurs données supplémentaires ont été introduites afin d'améliorer le taux de faux rejet du système, comme : les points de courbure maximale pour faciliter l'alignement des minuties ou leur index de qualité pour permettre de sélectionner les plus pertinentes. Ils obtiennent enfin un $FRR = 10\%$ (pour un FAR à 0%) sur une version partielle de la FVC2002 [MMJP09].

Plusieurs travaux remettent en question la sécurité et la préservation de la vie privée impliquées dans les différentes constructions du secure sketch :

– La donnée auxiliaire W du fuzzy commitment occasionne une divulgation d'information sur le modèle d'origine comme il est montré dans [Smi04]. Les travaux initiés par Zhou dans [ZWBK09a, ZKVB11] montrent que la corrélation existante dans les modèles biométriques binaires intraclasse peut aider à estimer à partir de W, la clé secrète S (la corrélation implique que la prédiction d'un élément x_j à partir d'un élément x_i dans le vecteur X est possible).
– Dans [STP09], Simoens *et al.* prouvent formellement que le fuzzy commitment ne peut pas être réutilisé dans différentes bases de données à partir de la même biométrie : le fuzzy commitment est dans ce cas non révocable. En effet, ils estiment une probabilité proche de 1 pour qu'un attaquant puisse lier les sketches entre eux. Le secure sketch n'a donc pas les propriétés de préservation de vie privée puisque sa réutilisation n'est pas possible.
– Dans [SB07], quatre différentes attaques, par rapport au fuzzy vault ont été distinguées. Ces attaques permettent de retrouver le modèle d'origine lorsque celui-ci est protégé par le fuzzy vault sur différentes bases de données. Le fuzzy vault lui aussi n'est pas réutilisable (ou révocable). Une tentative d'amélioration de cet aspect a été proposée dans [NNJ10b].

Afin d'améliorer la sécurité du *fuzzy commitment* qui est en relation avec la protection de la donnée auxiliaire W, Bringer et Chabanne [BC09] proposent d'utiliser la cryptographie hommomorphique. Ils combinent le fuzzy commitment avec la primitive de Goldwasser-Micali. Dans ce schéma de cryptographie asymétrique, une clé publique p_k et sa clé secrète s_k sont générées. La propriété hommomorphique de cette primitive est la suivante :

$$Enc(m, p_k) \times Enc(\acute{m}, p_k) = Enc(m \otimes \acute{m}, p_k) \tag{1.4}$$

ou bien,

$$Dec(Enc(m, p_k) \times Enc(\acute{m}, p_k), s_k) = m \otimes \acute{m} \tag{1.5}$$

Le protocole de vérification biométrique est maintenant décrit comme suit :

Fuzzy commitment et cryptographie hommomorphique

- **Durant l'enrôlement**, l'utilisateur U enregistre sa biométrie X auprès du serveur d'authentification. La serveur génère aléatoirement le mot de code C. En utilisant Goldwasser-Micali, le serveur chiffre $C \otimes X$ avec la clé publique p_k. Il sauvegarde sur la base de données $H(C)$ et $Enc(C \otimes X, p_k)$.
- **Durant la vérification**, l'utilisateur chiffre sa biométrie Y avec sa clé publique p_k et envoie $Enc(Y, p_k)$ au serveur. Celui-ci récupère $Enc(C \otimes X, p_k)$ et $H(C)$ de la base de données et envoie le produit $Enc(C \otimes X, p_k) \times Enc(Y, p_k)$ au gestionnaire de clé (*key manager*). Le key manager utilise la clé privée s_k pour calculer $Dec(Enc(C \otimes X, p_k) \times Enc(Y, p_k), s_k) = C \otimes X \otimes Y$ et envoie le résultat au serveur d'authentification pour le décoder vers le mot de code $\acute{C}$. Il vérifie ensuite si $H(C) = H(\acute{C})$

.

La propriété hommomorphique de Goldwasser-Micali assure que ni le modèle Y, ni la donnée W ne soient révélés en clair au niveau de la base de données. D'autres implémentations de la cryptographie hommomorphique pour assurer des protocoles privés de vérification biométrique existent. Pour les empreintes, nous citons le travail intéressant de Upmanyu *et al.* [UNSJ10]. Néanmoins, il reste à noter que la cryptographie hommomorphique est complexe à implémenter en terme de temps de calcul pour l'authentification biométrique.

Les primitives pour l'extraction d'un secret unique

Initialement proposées par Linnartz et Tuyls [LT03], le but est d'étudier le choix approprié d'une fonction de codage F qui permettrait d'estimer à partir du modèle biométrique X et de la clé S (S étant préalablement générée pour chaque utilisateur), la donnée auxiliaire W de telle façon que :

$$F(X, W) = S.$$

F est appelée *fonction de blindage* (*shielding function*). La figure 1.20 résume le principe de cette approche. Dans [TG04], différentes constructions de la fonction F ont été proposées. Les auteurs supposent que les données biométriques du vecteur X sont identiquement et indépendamment distribuées. Cette statistique idéale n'est pas évidente à réaliser en pratique et un travail plus accentué reste à faire afin de permettre à F d'être pratiquement faisable.

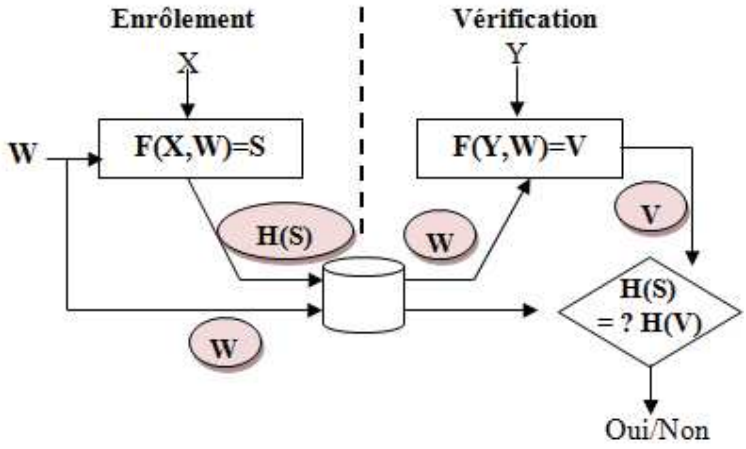


FIGURE 1.20 – Principe de fonctionnement des fonctions de blindage

Les primitives à base de codes graphiques

Comme il a été reporté précédemment, les systèmes à base des codes correcteurs algébriques dépendent fortement de la capacité de correction du code, qui parfois, ne suffit pas à combler la variabilité du signal biométrique en entrée, engendrant ainsi des taux d'erreur importants. Draper *et al.* [DKM+07] proposent d'utiliser comme alternative aux codes algébriques, les codes graphiques comme le code LDPC (Low Density Parity Chek). La figure 1.21 en illustre le fonctionnement générique.

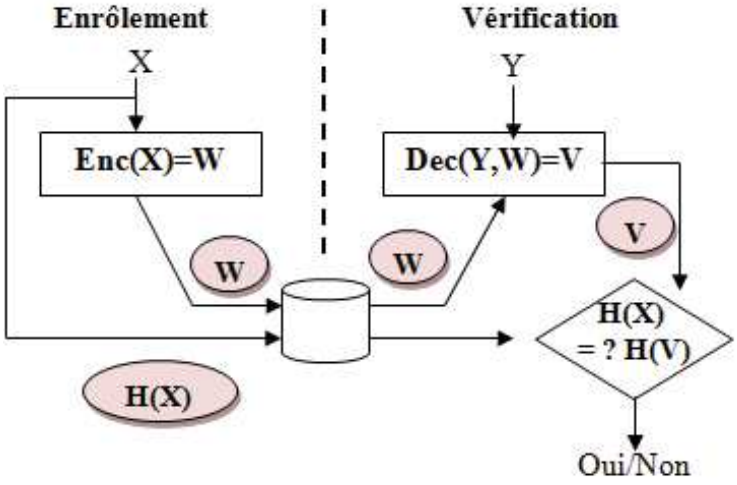


FIGURE 1.21 – Fonctionnement générique des primitives à base de codes graphiques

Le protocole de cette approche est décrit maintenant comme suit :

Primitive à base de codes graphiques

- **Durant l'enrôlement**, l'opération d'encodage *Enc* calcule le syndrome de X qui est alors la donnée auxiliaire W en utilisant le code LDPC. Seulement W et $H(X)$ sont stockés sur la base de données.
- **Durant la vérification**, le modèle en entrée Y est combiné avec W pour être décodé avec la fonction *Dec* dans le but d'estimer le vecteur X. L'authentification réussie dans le seul cas où cette estimation est parfaite : $H(X) = H(V)$.

Cette méthode impose une contrainte majeure sur le modèle X. En effet, celui-ci devrait être sous forme d'un vecteur binaire avant l'opération d'encodage (à l'instar du fuzzy commitement). Pour qu'elle soit applicable au modèle des minuties, les auteurs dans [NRV10] proposent un certain traitement, sur l'ensemble des minuties, pour les convertir en une représentation vectorielle et binarisée. Ils génèrent aléatoirement un ensemble de n régions cubiques. Sur chaque cuboïde, une information d'agrégation concernant les minuties appartenant au cube en question est extraite. Une binarisation est ensuite opérée en sélectionnant le seuil à partir de la médiane (la médiane étant estimée à partir d'une base d'apprentissage). Cette valeur représentera le cuboïde et l'empreinte sera caractérisée par l'ensemble des n cuboïdes. Il est à noter qu'une méthode d'alignement similaire à celle de Nandakumar *et al.* [NJP07] est effectuée à chaque nouvelle authentification. Une fois sécurisé par la primitive *Enc*, le taux de faux rejets devient assez intéressant : $FRR = 5\%$ pour un FAR $= 1.02\%$ sur une version partielle de la FVC2002-DB2. Sur l'aspect analyse de sécurité, la méthode semble résistante avec une certaine complexité au problème de réversibilité. Cette complexité est en fonction de la dimension k, du code LDPC. Il s'agit d'avoir une valeur k assez importante pour empêcher de trouver un vecteur X ayant le même syndrome que celui stocké sur la base de données. Néanmoins, cette analyse de complexité n'a pas été conduite sur les modèles issus de plusieurs scénarios de révocation et donc il n'existe pas encore de preuve complète sur la préservation de la vie privée par rapport à sa réutilisation.

Discussion

Nous venons de voir les crypto-systèmes biométriques comme moyen de sécurisation du modèle biométrique. De manière générale, ces systèmes tendent à corriger le bruit de la donnée biométrique en calculant une information publique appelée donnée auxiliaire W (ou helper data) à partir du modèle biométrique X. La donnée W peut aider soit à cacher une clé utilisateur S à l'intérieur de la biométrie, soit à générer une nouvelle clé S. Dans

tous les cas, uniquement W sera stockée sur la base de données et la révélation de S ne peut se faire qu'en introduisant la bonne biométrie Y. Néanmoins, il existe de nombreuses difficultés dans l'application de ces schémas. Généralement, ces schémas imposent en entrée un modèle X sous forme vectorielle et discrétisée ce qui n'est pas évident à modéliser pour certaines représentations d'empreintes comme c'est le cas pour les minuties. Cela a d'ailleurs ouvert la voie à plusieurs recherches qui s'intéressent à trouver ce type de représentations [XVB+09, BD10]. De plus, ils engendrent une perte d'entropie (la notion d'entropie sera explicitée dans le prochain chapitre) lorsqu'il y a une certaine corrélation entre les éléments du vecteur X comme c'est le cas pour le fuzzy commitment ou les fonctions de blindage (shielding functions). Pour assurer l'irréversibilité du système, il faudrait que X soit identiquement et indépendamment distribué ce qui n'est pas évident à garantir pour les modèles biométriques. Nous avons aussi vu que du coté préservation de la vie privée, ces schémas, reposant généralement sur les codes correcteurs d'erreur étaient difficilement réutilisables et donc non résiliables. Enfin, nous avons aussi constaté que par rapport aux empreintes digitales, les taux d'erreur étaient relativement élevés.

Dans la suite, nous abordons la seconde classe des méthodes de protection que sont les approches par transformation non-inversible ou les transformations révocables.

1.4.2 Les transformations révocables

Les approches de cette famille n'utilisent pas de données auxiliaires pour compenser la variabilité du signal biométrique, ce qui signifie que la comparaison est effectuée dans le domaine de la transformation directement entre les modèles transformés.

Supposons que X sera transformé en données codées T lors de l'enrôlement par l'utilisation d'une fonction F. Pour la vérification, la requête biométrique Y sera transformée en $\grave{T}$ toujours en utilisant la fonction F et l'authentification réussira si T est proche de $\grave{T}$ en utilisant une certaine mesure de similarité. Pour assurer la révocabilité du système, une donnée aléatoire S sous forme d'une clé est attribuée à chaque utilisateur U. La clé S est alors considérée comme un paramètre d'entrée de la fonction de transformation F. La révocation consiste au remplacement direct de cette clé utilisateur. La figure 1.22 résume le fonctionnement des transformations révocables.

La difficulté est alors dans la désignation de la fonction de transformation F. Pour ce faire, bon nombre de méthodes ont été proposées. Ces méthodes se distinguent entre elles de différentes manières. Par exemple, dans l'utilisation du paramètre S. La fonction F peut utiliser S de deux manières différentes : S peut contribuer en amont/aval de la transformation effective, soit en générant une configuration initiale à partir de laquelle

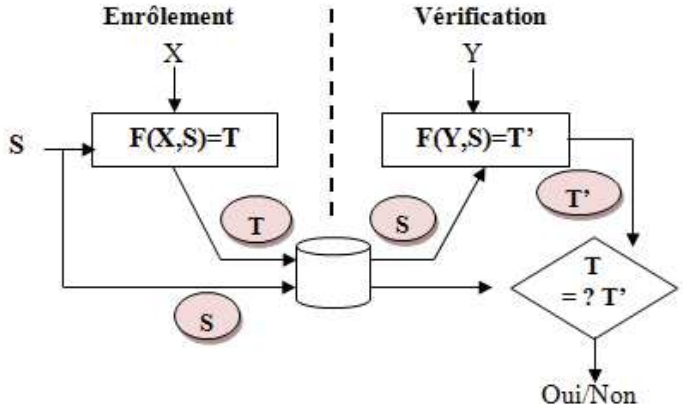


FIGURE 1.22 – Fonctionnement générique des transformations révocables

la transformation pourrait se faire ou en perturbant le modèle transformé par permutation par exemple. Dans l'autre cas, la fonction F s'agira d'un mécanisme de salage où la transformation est une opération qui combine X avec une donnée aléatoire générée à partir de S. Dans tous les cas, S est considéré comme le grain (seed) d'un générateur pseudo-aléatoire. Une autre façon de distinguer les méthodes entre elles a été adoptée par Rathgeb et Uhl [RU11]. Ils les classifient suivant celles qui utilisent une méthode de pré-alignement et celles qui sont basées sur l'alignement implicite (alignement free). Dans [TRCB08], nous trouvons une autre étude qui compare les transformations révocables avec et sans pré-alignement.

Ici, nous distinguons les méthodes entre elles en se basant sur différents critères que nous considérons comme éléments caractéristiques des méthodes de transformation. Il s'agit de :

- La représentation du modèle en entrée.
- Le format de la représentation finale après transformation.
- La fonction de transformation.
- La méthode d'alignement.
- Le scénario de révocabilité.
- L'évaluation de la méthode de transformation. Ce point est très important car il s'agit d'analyser si la méthode de protection proposée réponde bien aux exigences de sécurité et de préservation de vie privée souhaitées. Dans les différents papiers, cette évaluation est faite de façon *AD HOC* car il n'existe pas encore de méthodes formelles ou unifiées pour le faire. Généralement, il est question d'estimer le taux d'erreur à la reconnaissance ou le risque d'inversibilité de la transformation. Il faut aussi considérer le scénario du vol de la clé S. Cette attaque n'existait pas dans les

crypto-systèmes biométriques car la clé utilisateur est en quelque sorte cachée avec la biométrie et seule la donnée auxiliaire W est sauvegardée. Dans les transformations révocables, la clé est un paramètre qui doit être sauvegardé ce qui engendre des possibilités de divulgation.

Dans ce qui suit, nous présentons par ordre chronologique, les approches les plus pertinentes des transformations révocables :

L'approche d'Ang *et al.* [ARM05]

Le modèle des minuties est transformé en une représentation basée sur un ensemble de points. La clé $0 \leq S \leq \pi$ spécifie une ligne à travers le point core. La transformation consiste à déplacer les minuties qui sont sous cette ligne pour qu'elles soient sur la ligne. La méthode utilise un alignement absolu basé sur le point core (algorithme R92). Sur une base de données partielle, les auteurs obtiennent un $EER = 16.8\%$ ce qui reste assez faible. De plus, la méthode est facilement inversible à cause des minuties qui sont sur la ligne et qui restent donc inchangées.

L'approche de Teoh *et al.* [TNG04]

Cette approche, appelée *BioHashing*, illustre le processus de salage utilisé dans les transformations révocables. C'est une transformation, applicable à différentes modalités biométriques, qui génère un vecteur binaire appelé *BioCode*. Le principe du BioHashing consiste en la projection des données biométriques sur une base orthonormée obtenue à partir de la clé S. Les données biométriques doivent être sous forme d'un vecteur X de taille fixe : $X = (x_1, , x_n)$ avec $X \in R^n$. Cette étape de projection consiste à cacher l'information biométrique dans un certain espace comme suit : $AX = W$. Avec A une matrice aléatoire de dimension $m \times n$, générée à partir de la clé S, de telle sorte que $m < n$ pour éviter l'inversion du processus. Etant donné que A soit orthonormée, il va y avoir une préservation des distances après projection ce qui peut engendrer une faille de sécurité dans cette transformation. Une étape de binarisation du vecteur W est donc nécessaire pour assurer l'irréversibilité du processus. Teoh *et al.* [TNG04] appliquent le BioHashing aux empreintes digitales à partir de la transformée de Fourier-Mellin de l'image d'empreinte. Un pré-alignement par rapport au pont core a d'abord été nécessaire. Le BioHashing arrive à réaliser une séparation claire entre les distributions des utilisateurs légitimes et celle des imposteurs. Ainsi, en combinant le déterminisme de la clé S avec la donnée biométrique, la variation inter-classe va augmenter tandis que les distances intra-classe vont être préservées.

Cependant, comme constaté dans les autres transformations révocables, le BioHashing est lui aussi sensible au vol de clé est les performances ont tendance à se dégrader face à cette situation.

L'approche de Ratha *et al.* [RCCB07]

Cette approche a d'abord été initiée dans [BCR02] où l'idée des fonctions de transformation révocable (cancelable biometrics) a été introduite pour la première fois. La transformation consiste à introduire une distorsion intentionnelle sur la biométrie originale. Dans [RCCB07], les minuties sont transformées suivant trois différentes méthodes : cartésienne, polaire ou fonctionnelle. Un alignement des minuties par rapport au point core est d'abord nécessaire (les positions et les orientations des minuties sont exprimées par rapport au point core). A titre d'exemple, l'équation 1.6 illustre le principe de la transformation cartésienne entre un schéma 2×2 et une matrice binaire générée à partir d'une clé S.

$$\begin{bmatrix} 1 & 2 & 3 & 4 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} 3 & 2 & 4 & 3 \end{bmatrix} \tag{1.6}$$

$$\begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \Rightarrow \begin{bmatrix} 3 & 2 \\ 4 & 3 \end{bmatrix}$$

Pour les besoins d'irréversibilité, la matrice de transformation ne doit pas être une matrice de permutation de manière à ce que la transformation inverse soit de type one-to-many. Les auteurs ont démontré, sur une base de données partielle, que la transformation fonctionnelle avait une meilleure performance que les transformations cartésienne ou polaire avec un $EER = 10\%$. Dans un travail récent de Nagar et Jain [NJ09], la diversité et l'irréversibilité de la transformation fonctionnelle étaient sérieusement discutables dans la cas où les paramètres de la fonction de transformation étaient révélés (comme la clé S).

L'approche de Kumar *et al.* [KTG10]

Cette approche, initiée par Tulyakov *et al.* [TFG05] introduit la notion de fonctions de hachage symétriques pour transformer l'ensemble des n minuties en un ensemble de $\acute{n}$ valeurs de hachage.

Au lieu de décrire la minutie par le triplet (x, y, θ), chaque minutie est définie relativement à son k-voisinage local. Des caractéristiques de second degré, à savoir, les différences

d'orientation c_i entre la minutie de référence et chacune de ces k voisines sont extraites. Une minutie sera donc définie par k valeurs c_i. La méthode de hachage proposée consiste à générer pour chaque ensemble $c_1, c_2, \cdots, c_k$, les m valeurs suivantes :

$$\begin{aligned} h_1(c_1, c_2, \cdots, c_k) &= c_1 + c_2 + \cdots + c_k \\ h_2(c_1, c_2, \cdots, c_k) &= c_1^2 + c_2^2 + \cdots + c_k^2 \\ &\cdots \\ h_m(c_1, c_2, \cdots, c_k) &= c_1^m + c_2^m + \cdots + c_k^m \end{aligned} \tag{1.7}$$

Si le nombre m des fonctions de hachage est inférieur au nombre d'inconnues k alors il n'est pas possible de retrouver les valeurs initiales c_i à partir des m valeurs de hachage. Pour assurer la révocabilité, les auteurs, proposent de faire correspondre un index pour chaque k-voisinage. Pour les n minuties, on sélectionnera n indexes. L'index est dérivé à partir de l'angle le plus petit dans le k-voisinage. Ils génèrent ensuite à partir de la clé S une sorte de mappage entre l'ensemble des indexes possibles et les fonctions de hachage à considérer : h_1, h_2, jusqu'à h_{k-1}. Les auteurs obtiennent sur la $FVC2002 - DB2$, un EER convenable de 4.98%. Néanmoins, nous pensons que cette méthode a besoin de plusieurs scénarios de tests pour être solvable d'un point de vue sécurité. Par exemple, Il aurait été intéressant de tester la diversité de la méthode en considérant un scénario où le mappage *index-fonction* serait le même sur toute la base de données afin de simuler le scénario de vol de clé.

L'approche de Lee et Kim [LK10]

Cette approche transforme l'ensemble des minuties en un ensemble de vecteurs binaires. Chaque minutie sera donc représentée par un vecteur binaire. Cette approche met en avant l'exemple des transformations révocables qui se basent sur la génération d'histogrammes. Le principe général des ces méthodes est le suivant :

1. Générer une configuration initiale de taille fixe qui pourrait être assimilée à un nouveau système de coordonnées, qui de plus se base sur une certaine sectorisation.
2. Considérer, récursivement, chaque minutie comme point de référence de cette configuration et transformer les autres minuties par rapport à cette référence pour obtenir de nouvelles coordonnées.
3. Générer un histogramme en comptabilisant le nombre des minuties qui appartiennent aux secteurs de la configuration de base.

L'approche de Lee et Kim [LK10] considère une configuration initiale sous forme d'un tableau de cellules à 3 dimensions comme le montre la figure 1.23. Soit $M_i = [x_i, y_i, \theta_i, t_i]$

la minutie i représentée par sa position, son orientation et son type, respectivement. Soit $M_r = [x_r, y_r, \theta_r, t_r]$ la prochaine minutie de référence. Les autres minuties sont alors géométriquement transformées par rapport à M_r, de façon à ce que M_r soit au centre du tableau et son orientation soit parallèle à l'axe horizontal comme le montre la figure 1.23.

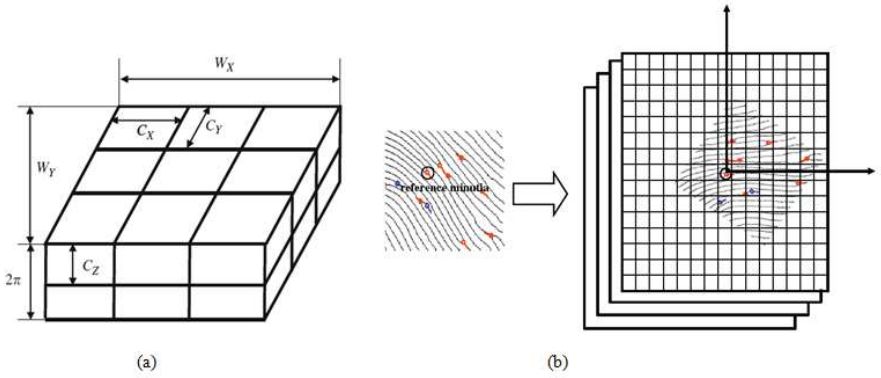


FIGURE 1.23 – Transformation des minuties suivant l'approche de Lee et Kim [LK10]. (a) Configuration initiale, (b) Principe de la transformation

Une minutie transformée $M_i^T = [x_i^T, y_i^T, \theta_i^T]$ est obtenue par l'équation 1.8.

$$\begin{pmatrix} x_i^T \\ y_i^T \\ \theta_i^T \end{pmatrix} = \begin{pmatrix} \cos\theta_r & -\sin\theta_r & 0 \\ \sin\theta_r & \cos\theta_r & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_i - x_r \\ -(y_i - y_r) \\ \theta_i - \theta_r \end{pmatrix} + \begin{pmatrix} W_x/2 \\ W_y/2 \\ 0 \end{pmatrix} \tag{1.8}$$

Après cette transformation, chaque cellule du tableau 3D qui contient au moins une minutie, aura une valeur de 1, 0 sinon (i.e. Il s'agit d'un tableau 3D dans le but de prendre en compte les 3 composantes x, y, et θ). Le tableau est ensuite linéarisé sous forme d'un vecteur binaire et sera considéré comme représentatif de la minutie M_r. L'alignement est implicitement géré par la fonction de transformation. Pour permettre la révocabilité, les éléments du tableau sont permutés suivant une matrice générée à partir de la clé utilisateur S. Il y aura autant de vecteurs binaires que de minuties. En considérant le scénario de vol de clé, les auteurs obtiennent un $EER = 9.5\%$ sur la FVC2004-DB2.

L'approche d'Ahmad *et al.* [AHW11]

Dans cette approche, le processus de transformation est d'abord précédé par une étape de sélection des minuties. Parmi le nombre total des minuties, il ne sera retenu que celles qui sont suffisamment distantes l'une de l'autre. La transformation est ensuite basée sur le même principe que la transformation polaire proposée par Ratha *et al.* dans [RCCB07]. Cependant, pour éviter le pré-alignement par rapport au point core, les auteurs proposent d'exprimer dans le système polaire, le rapport entre une minutie de référence est les autres

minuties de l'ensemble. Comme le montre la figure 1.24, cette information polaire entre deux minuties m_i et m_j est exprimée par le vecteur $V_{ij} = (L, \alpha_i, \beta_j)$ avec L la distance entre m_i et m_j, α_i est l'angle entre la ligne $m_i m_j$ et l'orientation de la minutie m_i, β_j est l'orientation du vecteur $\overrightarrow{m_i m_j}$.

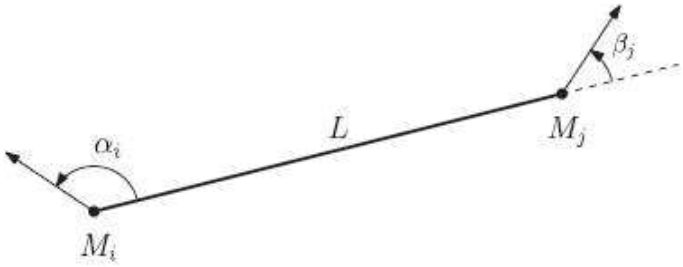


FIGURE 1.24 – Représentation par vecteur polaire d'Ahmad *et al.* [AHW11]

Pour la fonction de transformation, le système des coordonnées polaires est divisé en secteurs autour de la minutie de référence. Chaque secteur est ensuite mappé vers un autre secteur en utilisant la formule : $new\_sect = (old\_sect + v_w[old\_sect]) \mod total\_sect$. v_w est un vecteur aléatoire généré à partir de la clé S qui assure la révocabilité de cette transformation. Le vecteur $V_{ij} = (L, \alpha_i, \beta_j)$ qui appartenait au secteur old_sect est transformé vers le vecteur $\Gamma(V_{ij}) = (\grave{L}, \grave{\alpha}_i, \grave{\beta}_j)$ qui appartient maintenant au secteur new_sect.

Etant donnée que la transformation utilisée est de type one-to-many s'agissant du vecteur v_w, l'inversibilité de la méthode devient assez complexe. Sur une version partielle de la $FVC2002 - DB2$, les auteurs réalisent un $EER = 6\%$ en stipulant que la clé S est était volée ce qui est conforme aux avancées actuelles. Malheureusement, nous notons dans les tests l'utilisation restrictive de la $FVC2002 - DB2$ ce qui n'est pas représentatif de toute la base avec ces différentes qualités d'images. Nous pensons que cela influe sur l'appréciation générale de la méthode.

L'approche de Jin *et al.* [JTOT12]

Cette approche transforme l'ensemble des minuties en un ensemble de vecteurs binaires. Pour chaque minutie de référence, les minuties restantes sont transformées en utilisant les coordonnées polaires par rapport à cette référence. Ainsi, chaque minutie i sera exprimée par le couple (ρ_i, α_i) que sont la distance radiale et l'angle radial, respectivement. Les coordonnées polaires éliminent le besoin d'un pré-alignement. Une grille polaire de l secteurs est ensuite définie autour de la minutie de référence comme le montre la figure 1.25.

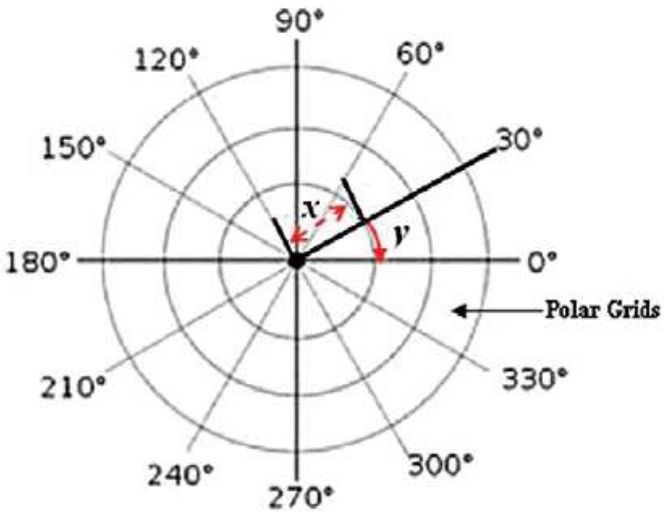


FIGURE 1.25 – La grille polaire avec x la distance radiale et y l'angle radial [JTOT12]

Les auteurs définissent maintenant un histogramme H de taille l qui comptabilise le nombre des minuties appartenant à chaque secteur de la grille polaire. Le vecteur caractéristique de la minutie est défini par la binarisation suivante :

$$\forall i \in [0, l[, b_i = \begin{cases} 1 & \text{si } h_i \geq 1 \\ 0 & \textit{sinon.} \end{cases} \tag{1.9}$$

L'empreinte digitale sera représentée par autant de vecteurs binaires que de minuties. Afin d'assurer la révocabilité de la méthode, une clé S est utilisée pour permuter aléatoirement les bits dans chacun des n histogrammes. Sur le $FVC2002 - DB2$, les auteurs obtiennent une $EER = 5.65\%$. Pour le scénario de vol de clé, cet EER devient égal à 6.94%. Néanmoins, même si cela n'est pas précisé par les auteurs, nous pensons qu'assurer la révocabilité par une simple permutation n'est pas tout à fait satisfaisant sur le point sécurité de la transformation. En effet, la matrice de permutation est inversible dans le cas de révélation de la clé S. Il devient donc facile de retrouver les positions des minuties dans la grille polaire ce qui permettra d'estimer un pseudo-inverse du modèle d'origine capable d'être correctement apparié.

L'approche de Wang *et al.* [WH12]

En se basant sur le même modèle d'origine qu'Ahmed *et al.* [AHW11], à savoir les vecteurs polaires V_{ij}, les auteurs opèrent une toute autre transformation. D'abord, chaque vecteur $V_{ij} = (L, \alpha_i, \beta_j)$ est quantifié sur $n = n_L + n_{\alpha_i} + n_{\beta_j}$ *bits* en supposant que $n_L, n_{\alpha_i}, n_{\beta_j}$ soient respectivement les bits nécessaires pour représenter les éléments L, α_i, β_j.

Un histogramme H de taille 2^n est ensuite créé de telle façon que chacun de ces éléments est à 1 si une seule occurrence de V_{ij} existe, il est à 0 sinon (dans les tests, $n = 15$). Il est maintenant question d'assurer la protection du vecteur binaire H. H est d'abord transformé en un vecteur de réels en utilisant la transformée de Fourrier discrète. Une transformation basée sur un mappage infinite-to-one et non seulement comme précédemment many-to-one est opérée de la manière suivante : H de dimension $q = 2^n$ est algébriquement transformé en un vecteur T de taille p à partir de la matrice aléatoire A de dimension $p \times q$ en utilisant l'équation : $AH = T$. En posant $p < q$, ce système linéaire contiendra plus d'inconnues que d'équations ce qui lui assure un nombre infinie de solutions et garantit, d'après les auteurs l'irréversibilité du modèle transformé. Sur la $FVC2002 - DB2$, les auteurs obtiennent un $EER = 5\%$ pour le scénario de vol de clé, ce qui les met en très bonne position par rapport aux récents travaux. Néanmoins, un travail de Li et Hu [LH13] a remis en question la sécurité de cette méthode en considérant qu'après plusieurs scénarios de révocation, la concaténation de plusieurs matrices de transformation A pourrait générer une matrice de rang q de telle sorte qu'il devient trivial de retrouver H.

Récemment, une approche assez originale a été proposée par Wong *et al.* dans [WTWY13]. Cette approche s'inspire du descripteur *MCC* (Minutia Cylinder-Code) utilisé par Ferrara *et al.* [FMC12]. Nous retenons de cette solution une taille de modèle assez importante de 14KO mais un EER très satisfaisant pour le scénario de vol de clé sur la FVC2002-DB2 : $EER = 2.54\%$.

Discussion

Nous venons de présenter différentes approches de transformation révocable pour les empreintes digitales. Ces méthodes, même si elles se distinguent de manière claire dans leurs principes, se rejoignent sur les différents points suivants :

- D'abord, leur objectif commun est d'éviter de stocker le modèle biométrique d'origine et dans l'idéal, pouvoir effectuer une reconnaissance fiable dans le domaine de la transformée.
- Elles permettent de révoquer et de générer un nouveau modèle de référence. C'est principalement l'introduction d'une clé utilisateur qui permet cette diversification du modèle à partir du même trait biométrique.
- Elles doivent prendre en charge le problème d'alignement des empreintes. Soit, elles procèdent par pré-alignement (difficile car la détection des points singuliers n'est pas évidente). Soit, elles suscitent des méthodes d'alignement implicite (registration-free).

– Lorsqu'il s'agit de minuties, ces méthodes doivent être robustes à la variabilité qu'engendre ce type de caractéristiques (non chevauchement, oubli ou rajout, etc.).
– Enfin, le point culminant reste de pouvoir faire une analyse de sécurité rigoureuse de la méthode proposée. Aujourd'hui, il n'existe pas encore de méthode formelle ou bien standardisée pour le faire. Cela constitue une véritable problématique.
– Majoritairement, ces méthodes sont sensibles à des attaques communes auxquelles il faudrait que la recherche pallie. Il s'agit entre autres du vol de la clé utilisateur (Le FAR est dans ce cas assez élevé pour une application sécurisée de la méthode). De plus, d'après les méthodes présentées, la transformation est souvent inversible (partiellement ou totalement) lorsque tous les paramètres sont connus comme la clé S et le modèle issu de la transformation.

1.5 Conclusion

Après avoir souligné l'importance de protéger le modèle biométrique, nous avons pu voir deux grandes familles de solutions. Principalement, des solutions basées sur l'architecture du système et des solutions logicielles orientées sur le traitement du modèle biométrique.

D'une part, nous avons vu qu'utiliser un élément sécurisé (ES) pour stocker le modèle biométrique était une solution pratique. Néanmoins, il n'est pas évident de pouvoir mettre un tel un système avec des microprocesseurs à capacité assez réduite. De plus, même s'il est techniquement possible de mettre à jour le contenu d'un ES, ces solutions n'apportent pas de garantie sur la propriété de révocabilité du modèle biométrique.

D'autre part, qu'il s'agisse de crypto-systèmes biométriques ou de transformations révocables, les approches ne répondent pas encore efficacement à tous les impératifs de sécurité souhaités. La difficulté est que la transformée est dans la plupart des cas, entièrement ou partiellement inversible. Le critère de révocabilité tant souhaité n'est pas évident à réaliser sans engendrer d'autres risques comme l'attaque par vol de clé.

Dans les chapitres qui suivent nous présentons nos principales solutions. Nous commençons dans le chapitre 2 par aborder le problème d'évaluation des méthodes de protection des modèles biométriques. Ce maillon manque considérablement dans les chaines de traitement alors qu'il est essentiel pour analyser la sécurité ou le degré de préservation de la vie privée des méthodes proposées. Nous nous intéressons principalement aux transformations révocables en proposant un processus d'évaluation sous forme de critères et de métriques.

Chapitre 2

Approche pour l'évaluation des schémas de biométrie révocable

Ce chapitre présente la méthode d'évaluation des systèmes de protection de gabarits biométriques que nous avons développée. Cette proposition s'articule autour de trois points : la paramétrisation du processus d'évaluation par la fonction de transformation en entrée, la définition d'un ensemble de critères pour évaluer les systèmes par transformation révocable et la confrontation des critères proposés face à une base commune d'attaques.

Sommaire

2.1 Introduction

Les modèles biométriques, qui servent à reconnaître les individus, sont des données à caractère personnel et sensible. Afin d'assurer leur sécurité, les schémas de protection des modèles biométriques ont été présentés comme solution possible. Leur objectif commun est de transformer le modèle d'origine pour générer une nouvelle référence, capable de vérifier l'identité des personnes, tout en étant non inversible, révocable et intraçable. Bon nombre de méthodes de protection existent dans la littérature, cependant, il reste

difficile de les évaluer en terme de sécurité ou de protection de la vie privée. L'absence d'un processus d'évaluation généralisé et opérationnel empêche de les comparer, de les classifier ou de garantir leur fiabilité. Ainsi, il devient important d'établir un cadre général, pour évaluer systématiquement de telles technologies.
La définition de métriques reste l'élément principal d'un processus d'évaluation. Dans ce contexte, contrairement à son aspect mathématique, une métrique sert à produire des valeurs qui permettent de comparer les différents algorithmes par rapport aux critères à évaluer. Nonobstant, nous remarquons que les travaux sur l'évaluation de la littérature sont plus axés sur l'identification des points de vulnérabilités et des classes d'attaques que sur la définition de métriques. Dans ce chapitre, nous proposons une méthodologie d'évaluation générique paramétrable. Pour consolider toutes les méthodes de protection, cette méthodologie s'adapte en fonction du schéma de protection à évaluer. Nous générons ensuite, à partir de cette méthodologie, une méthode d'évaluation pour les transformations révocables. Cette méthode est fondée sur un ensemble de métriques qui quantifient les exigences de sécurité et de préservation de la vie privée souhaités. Cette méthodologie d'évaluation servira de ligne directrice pour analyser les schémas de protection présentés dans le cadre de cette thèse.

Le chapitre est structuré comme suit : dans la section 2.2, nous présentons une rétrospective des travaux existants. Nous proposons notre méthodologie d'évaluation dans la section 2.3 et nous proposons plusieurs critères d'évaluation pour analyser les systèmes par transformations révocables dans la section 2.4. Nous concluons, dans la section 2.5, par un résumé sur les propositions faites dans ce chapitre.

2.2 Travaux existants

Dans le chapitre 1, nous avons présenté un nombre important de méthodes de protection de gabarits biométriques (e.g. [ARM05, TNG04, RCCB07, KTG10, LK10, AHW11, WH12, UJ06, JW99]). Une question essentielle est liée à leur évaluation. En s'appuyant sur les quelques références de la littérature, nous remarquons que les principaux travaux sur l'évaluation se focalisent sur un algorithme particulier dans le but de détecter des failles ou des points d'attaques. Les références A. Smith [Smi04], T. Ignatenko [Ign09], Zhou *et al.* [ZKVB11], Simoens *et al.* [STP09] analysent le risque de divulgation d'informations dans le fuzzy commitment [JW99]. Ces travaux démontrent l'importance d'étudier la distribution statistique du modèle biométrique X en entrée du fuzzy commitment. En effet, si une corrélation existe dans X, la complexité de son estimation diminue. De plus, si le type de code correcteur d'erreur utilisé est connu, il n'est pas nécessaire d'estimer tout le

vecteur X pour pouvoir déduire la clé S.

L'article de Scheirer et Boult [SB07] propose différentes attaques contre le fuzzy vault [JS02]. Les travaux de Quan *et al.* [QFAF08] utilisent trois différentes attaques pour inverser la transformation fonctionnelle de Ratha *et al.* [RCCB07]. Les résultats démontrent que la plupart des minuties transformées peuvent être exactement inversées vers leur version d'origine. La référence de Nagar *et al.* [NNJ10a] analyse les risques d'inversibilité et de traçabilité dans le BioHashing [TNG04] et la transformation de Ratha *et al.* [RCCB07]. Plus récemment, les auteurs de [LH13] utilisent l'attaque par références multiples pour inverser les différentes transformations révocables présentées dans [LCT+07, AHW11, WH12]. Cette attaque se réfère à l'utilisation de plusieurs modèles révoqués d'un même utilisateur pour estimer le modèle biométrique d'origine.

Récemment, de nouvelles orientations apparaissent, dont le but est d'établir un référentiel commun pour l'évaluation. En 2012, le projet européen BEAT [1] (Biometrics Evaluation and Testing) voit le jour. Son objectif est de standardiser l'évaluation des méthodes de protection, qui est loin d'être un exercice simple. La même année, deux importants travaux sont rendus publics, respectivement, la thèse de Xuebing Zhou [Zho12] et le papier de Simoens *et al.* [SYZ+12]. Dans sa thèse, X. Zhou présente son cadre général d'évaluation qui se base, séquentiellement, sur les étapes suivantes :

1. **La détermination des objectifs de protection** : la première étape consiste à clarifier les objectifs de l'algorithme de protection comme par exemple : l'intraçabilité, la capacité de protection de la vie privée ou la sécurité.
2. **La détermination du modèle d'attaque** : cette étape consiste à définir les capacités d'un adversaire comme les informations ou les ressources disponibles.
3. **La détermination des métriques d'évaluation** : cette étape est la base de l'évaluation. Elle consiste à choisir les bonnes métriques pour quantifier les vulnérabilités de l'algorithme de protection.
4. **L'évaluation et l'analyse** : il s'agit ici d'évaluer concrètement la méthode de protection en documentant toutes les étapes du processus.

A partir de ce modèle, X. Zhou présente une évaluation de la méthode du fuzzy commitment appliquée aux modalités du visage et de l'iris, en définissant des métriques formelles, appropriées pour le fuzzy commitment. La généralité de ce travail se situe au niveau de l'harmonisation des étapes de l'évaluation. Par rapport aux métriques, il y a une dépendance, plus au moins forte, avec l'algorithme de protection considéré. Parallèlement, la

1. http ://www.beat-eu.org

proposition de Simoens *et al.* [SYZ+12] se base sur l'unification des objectifs de protection en définissant un ensemble de critères communs applicables à toutes les méthodes de protection. Ce travail ouvre la voie à une possible standardisation même s'il ne définit pas de métriques pour les critères souhaités.

En effet, ce sont les métriques qui donnent de réelles indications sur les performances ou la fiabilité de la méthode. Une étude présentée par Buhan *et al.* [BKS10] rassemble un ensemble de métriques, issues de la théorie de l'information, qui quantifient la sécurité et la préservation de vie privée dans les crytpo-systèmes biométriques. Nous présentons, pour notre part, dans le tableau 2.1, les principales métriques opérationnelles recensées dans la littérature.

Dans ce tableau, nous remarquons qu'il existe essentiellement deux types de métriques : des métriques théoriques et des métriques pratiques. Les métriques théoriques se basent principalement sur le calcul d'entropie pour valider la sécurité des protocoles. L'entropie de Shannon mesure la quantité d'information réellement présente dans une donnée aléatoire. Un autre concept de Shannon est la notion d'information mutuelle. L'information mutuelle mesure de combien l'entropie de la première variable aléatoire va diminuer lorsque la seconde variable aléatoire est connue. Le calcul des métriques à base d'entropie repose sur l'étude des propriétés statistiques des différentes variables aléatoires, en l'occurrence, la distribution des modèles biométriques. L'entropie d'une source de données est maximale lorsque tous ses symboles sont équiprobables. Pour mesurer l'irréversibilité du fuzzy commitment, Zhou dans [ZKVB11], propose de calculer l'entropie conditionnelle $H(X|W)$, qui mesure le nombre d'essais moyen pour estimer la donnée biométrique X à partir du modèle transformé W. Cela est une utilisation possible de l'entropie.

Dans les métriques pratiques, il s'agit de mesurer la complexité des attaques. Pour ne pas surestimer les méthodes, le processus d'évaluation devrait être exhaustif en termes de risques possibles. Il serait donc intéressant d'avoir une base commune d'attaques. Pour l'heure, cela n'existe pas encore. Nous faisons l'effort, dans ce qui suit, de présenter une liste des attaques les plus pertinentes :

- **Attaque à zéro effort [JNN08]** : il s'agit d'essayer de s'introduire dans le système en exploitant le FAR (False Acceptance Rate) du système. Si ce taux n'est pas suffisamment faible, des risques d'intrusion existent.
- **Attaque par vol de clé [KCZ+05]** : il s'agit de la même attaque que précédemment mais en ayant connaissance de la clé utilisateur. Cette attaque peut être sérieusement dangereuse dans les systèmes à base de transformation révocable.

- **Attaque du FAR off-line [TH11]** : il s'agit de l'attaque précédente mais effectuée de manière *off-line*. Il faut pour cela avoir aussi le modèle de référence pour simuler la borne biométrique.
- **Attaque par traçabilité (linkage attack) [SB07]** : il s'agit de chercher une correspondance entre les différents modèles d'un même utilisateur à travers différentes applications.
- **Estimation de l'inverse** : il s'agit d'estimer le modèle biométrique d'origine de façon totale ou partielle.
- **Estimation du pseudo-inverse [NJ09]** : il s'agit d'estimer un modèle biométrique qui ne correpond pas au modèle d'origine mais qui pourrait correspondre, après transformation, au modèle transformé de référence.
- **Attaque par références multiples [LH13]** : il s'agit de faire l'attaque par estimation inverse mais à partir de plusieurs modèles révoqués appartenant au même utilisateur.
- **Attaque par force brute** : il s'agit de l'estimation aléatoire du modèle d'origine.
- **Attaque par écoute** : il s'agit d'utiliser les modèles révoqués d'un même utilisateur pour estimer le prochain modèle accepté.
- **Attaque par vol de biométrie [Hil01]** : il s'agit d'utiliser une donnée biométrique volée, comme la trace d'une empreinte, pour être accepté par le système.

Les métriques présentées dans le tableau 2.1 prennent en compte partiellement ces attaques. Principalement, les métriques de Nagar *et al.* [NJ09] qui quantifient les attaques à zéro effort, par vol de clés, estimations de l'inverse/pseudo-inverse et l'attaque par traçabilité.

Travaux	**Sécurité**	**Irréversibilité**	**Divulgation partielle de la biométrie d'origine**	**Intraçabilité et Diversité**
Les travaux de Zhou *et al.* [ZKVB11] pour évaluer le fuzzy commitment $W = S \otimes X$	$H(S\|W)$: entropie conditionnelle	$H(X\|W)$: entropie conditionnelle	$I(X;W)$: information mutuelle	–
Les travaux de Nagar *et al.* [NJ09] sur les transformations révocables	ROC_{diff} : courbe ROC de l'attaque à zéro effort et ROC_{same} : courbe ROC de l'attaque par vol de clé	*IRIS* : Risque d'intrusion dû à l'inversion de la totalité de la donnée biométrique	*IRIS* : Risque d'intrusion dûe à l'inversion partielle de la donnée biométrique	Cross-match rate ou taux de compatibilité croisé
D'autres travaux sur les transformations révocables	EER du système sans vol de clé et EER du système par vol de clé	Complexité en terme de nombre d'essais par force brute nécessaires pour trouver la biométrie X	–	Moyenne et variance de la distribution pseudo-imposteur

TABLE 2.1: Principales métriques d'analyse des méthodes de protection biométrique

2.3 Méthodologie proposée

Notre objectif est d'offrir un formalisme commun et complet pour évaluer les algorithmes de protection de données biométriques. Nous entendons par formalisme, l'ensemble des critères à évaluer ainsi que les métriques s'y relatant. Cependant, même si les objectifs et les critères peuvent être communs pour tous les algorithmes de protection, il est difficile d'unifier l'ensemble des métriques, la plupart du temps liées à la fonction de transformation. Ainsi, nous pensons que la généricité des fonctions impliquées dans les schémas de protection est une question fondamentale pour l'unification des métriques d'évaluation. Se basant sur notre étude bibliographique (du chapitre 1), nous proposons dans le tableau 2.2, une taxonomie en quatre classes différentes, représentant les constructions génériques possibles pour représenter l'ensemble des algorithmes existants de protection. Ces constructions représentent : les fonctions de transformation utilisées lors de l'enrôlement et de la vérification, la fonction de test de vérification et l'ensemble de l'information qui peut être publique en accord avec le principe de Kerchkoff stipulant que l'ennemi connait le système. Sur le tableau 2.3, nous donnons quelques exemples sur ces constructions.

N°	Enrôlement	Vérification	Test	Information publique		
				Fonctions	Paramètres	Données stockées
1	$F(X,S) = W$	$G(Y,W) = V$	$H(S)? = H(V)$	F, G	X, Y, S	W, H(S)
2	$F(X,W) = S$	$F(Y,W) = V$	$H(S)? = H(V)$	F	X, Y, S	W, H(S)
3	$F(X) = W$	$G(Y,W) = V$	$H(X)? = H(V)$	F, G	X, Y	W, H(X)
4	$F(X,S) = T$	$F(Y,S) = \grave{T}$	$Distance(T, \grave{T})$	F et Distance	X, Y, S	S, T

TABLE 2.2: Taxonomie des constructions génériques des algorithmes de protection des modèles biométriques

Construction générique	Exemples
1	Secure Sketch [DRS04]
2	Fonction de blindage [LT03]
3	Draper *et al.* [DKM+07]
4	Transformations révocables

TABLE 2.3: Constructions génériques et exemples

Sur la figure 2.1, nous présentons la méthodologie d'évaluation proposée. Nous reprenons

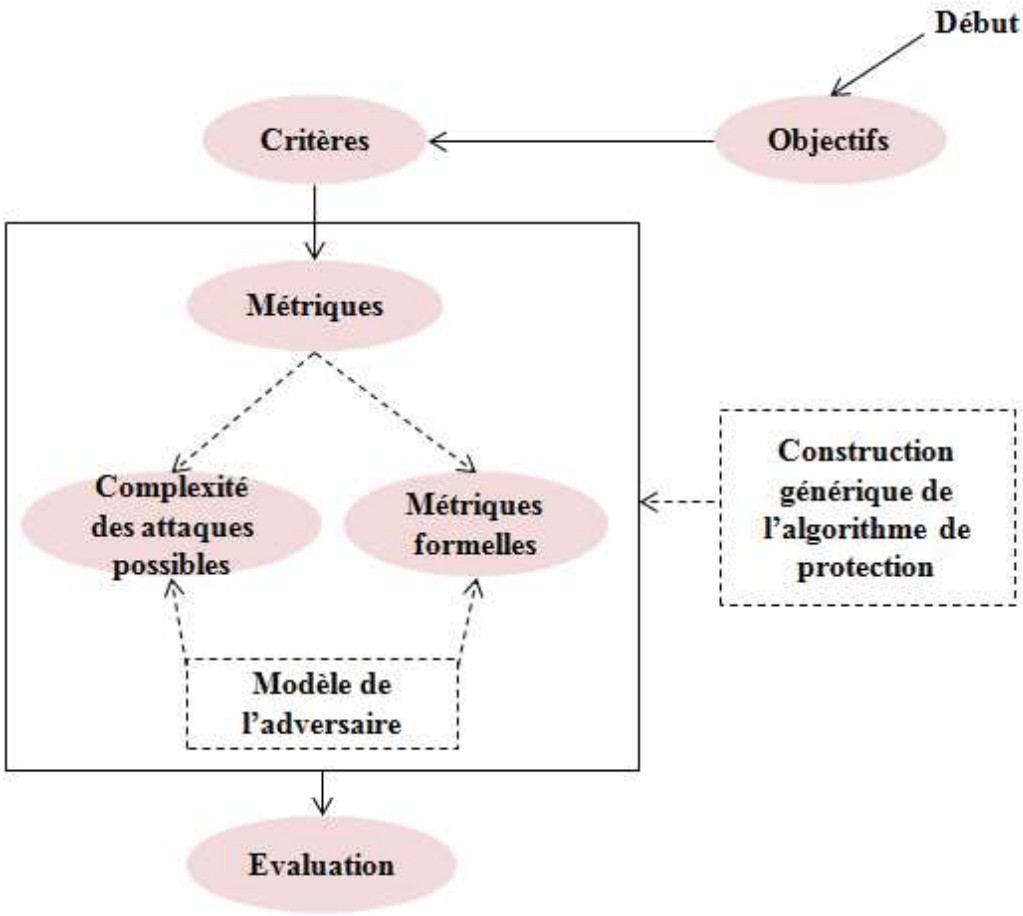


FIGURE 2.1 – Processus général d'évaluation des schémas de protection des modèles biométriques

les principales étapes présentées par X. Zhou [Zho12] en y ajoutons une paramétrisation des métriques par rapport à la construction générique de l'algorithme de protection. Nous impliquons aussi le modèle de l'adversaire comme information supplémentaire pour la formulation des métriques. Nous explicitons, maintenant, les principales étapes d'évaluation présentes sur la figure 2.1 comme suit :

1. **Détermination des objectifs** : Avant que les différents critères à évaluer ne soient précisément formulés, il faut définir correctement les objectifs de l'algorithme de protection.
2. **Détermination des critères** : il s'agit de formaliser, sous forme de critères communs, l'ensemble des objectifs cités dans l'étape précédente.

3. **Détermination des métriques** : cette étape consiste à préciser les mesures pour les différents critères souhaités. Elle est paramétrée par le choix de la construction générique de l'algorithme de transformation et du modèle de l'adversaire. Une taxonomie constituée de quatre classes est proposée dans le tableau 2.2. Les métriques sont ensuite formulées suivant la construction générique sélectionnée. Nous proposons d'utiliser deux classes de métriques. D'abord, nous proposons des métriques mesurant la complexité des attaques possibles, comme en cryptanalyse. Il s'agit de répertorier l'ensemble des attaques qu'un imposteur puisse accomplir et d'estimer leur complexité. L'analyse est ensuite complétée par la seconde classe de métriques, plus formelles, issues de la théorie de l'information. Ces métriques mesurent la difficulté de casser le processus de protection biométrique dans le cas extrême où l'adversaire possèderait des ressources illimitées. Chaque métrique dépend du modèle de l'adversaire qui définit les capacités de l'attaquant. Dans ce processus d'évaluation, nous utilisons deux modèles possibles, à savoir :
   - **L'adversaire malicieux** : un adversaire malicieux est un adversaire actif qui connait le système. Il a en effet, connaissance de toutes les fonctions, les paramètres et les données générées durant l'enrôlement et l'authentification. Il peut de plus changer les informations utilisées durant l'authentification.
   - **L'adversaire honnête mais curieux** : ce type d'adversaire ne peut pas changer d'information mais a lui aussi connaissance de toutes les fonctions, les paramètres et les données générées durant l'enrôlement et l'authentification.
4. **Evaluation** : il s'agit du calcul effectif des métriques. La simulation des attaques se fait à partir d'une base de données, réelle ou synthétique, et en suivant un protocole de test précis.

Dans la section suivante, nous instancions ce modèle pour les systèmes de protection par transformation révocable, les transformations révocables étant le sujet principal de cette thèse.

2.4 Critères d'évaluation des systèmes de biométrie révocable

En se référant au processus général d'évaluation présenté dans la section précédente, l'analyse des systèmes par transformation révocable peut être faite relativement aux étapes suivantes :

2.4.1 Objectifs de l'algorithme de protection biométrique

Les algorithmes de protection de la biométrie sont principalement conçus pour satisfaire des objectifs de sécurité et de préservation de la vie privée. Comme dans [BKS10], nous définissons la sécurité comme étant « la difficulté pour un adversaire d'avoir une fausse acceptation ». Cependant, comme le système peut être attaqué sur différents niveaux, la sécurité peut aussi concerner le contrôle d'intégrité de la biométrie en entrée (fausse donnée), la capacité de révocabilité des références biométriques dans le cas de leur compromission ou la confidentialité de la référence biométrique. Il est à noter que le contrôle d'intégrité de la biométrie n'est pas un objectif des algorithmes de protection. De même, la préservation de la vie privée pourrait être vue comme étant le niveau de protection contre un adversaire qui cherche à obtenir une quelconque information autre que la décision de vérification ou l'identité acclamée.

2.4.2 Détermination des critères à évaluer

Relativement aux objectifs que nous venons de citer, les critères potentiels à évaluer peuvent être reformulés comme suit :

- **Critères concernant la sécurité**
  1. Les risques d'intrusion.
  2. La révocabilité et le renouvellement.
  3. Le contrôle de confidentialité : ce critère peut être assimilé aux deux critères d'irréversibilité et de divulgation partielle de l'information biométrique.
- **Critères concernant la préservation de la vie privée**
  1. Irréversibilité.
  2. Divulgation partielle de l'information biométrique (privacy leakage).
  3. Intraçabilité.
  4. Diversité.

Ces critères sont maintenant définis de manière plus formelle dans le cadre des transformations révocables. Nous utilisons les notations suivantes comme dans l'article de Nagar *et al.* [NNJ10a] avec :

- b_z et $\acute{b}_z$, représentant respectivement, les modèles de référence et en entrée de l'utilisateur z. Nous utilisons la même notation b_z pour tous les modèles de référence quelque soit la variation intra-classe qui puisse exister entre eux.
- f la fonction de transformation. Nous notons n la dimension du résultat de $f(b_z)$ pour l'utilisateur z.
- K_z, l'ensemble des paramètres de la transformation correspondant à l'utilisateur z.

– D_O et D_T, les fonctions de distance entre les modèles biométriques dans le domaine original et le domaine de la transformation, respectivement.

Le module de vérification du système révocable retourne R_z comme valeur de décision de telle sorte que :

$$R_z = 1_{\{D_T(f(b_z,K_z),f(\acute{b}_z,K_z))\leq\epsilon\}} \tag{2.1}$$

La valeur ϵ est appelée seuil de décision du système révocable. Nous définissons les critères à évaluer de la manière suivante :

1. **Risque d'intrusion** : la fonction de transformation ne doit pas affecter les performances du système biométrique d'origine. Pour évaluer la fiabilité d'un système biométrique, nous considérons généralement les deux mesures d'erreur suivantes :

$$FRR_O(\epsilon) = P(D_O(b_z, \acute{b}_z) > \epsilon) \tag{2.2}$$

$$FAR_O(\epsilon) = P(D_O(b_z, \acute{b}_y) \leq \epsilon) \tag{2.3}$$

Avec FRR_O et FAR_O, respectivement, les taux de faux rejets et de fausses acceptations du système sans aucune transformation. Pour le système avec protection, nous considérons les deux mesures d'erreur suivantes telles que définies par Nagar *et al.* [NNJ10a] :

$$FRR_T(\epsilon) = P(D_T(f(b_z, K_z), f(\acute{b}_z, K_z)) > \epsilon) \tag{2.4}$$

$$FAR_T(\epsilon) = P(D_T(f(b_z, K_z), f(\acute{b}_y, K_y)) \leq \epsilon) \tag{2.5}$$

Avec FRR_T et FAR_T, respectivement, les taux de faux rejets et de fausses acceptations du système avec transformation.

2. **Révocabilité/Renouvellement** : Il devrait être possible de révoquer le modèle transformé et de générer un nouveau modèle à partir des mêmes données d'origine. Il est possible de calculer le modèle $f(b_z, K_z^1)$ en utilisant les paramètres K_z^1 et de le révoquer en le remplaçant $f(b_z, K_z^2)$ par l'introduction d'un autre ensemble de paramètres K_z^2.

3. **Irréversibilité et divulgation partielle de l'information biométrique** : il ne devrait pas être possible d'obtenir, à partir du modèle transformé, suffisamment d'information sur la biométrie d'origine (modèle ou échantillon biométrique). Cela assure la confidentialité de la donnée biométrique et prévient des attaques consistant

à falsifier la biométrie à partir d'un modèle compromis ou volé. Quelque soit le processus d'inversion, l'imposteur aura une information A_z qui peut l'authentifier comme l'utilisateur légitime. Le succès de l'attaque est donné par :

$$FAR_A(\epsilon) = P(D_T(f(b_z, K_z), A_z) \leq \epsilon) \tag{2.6}$$

4. **Intraçabilité/Diversité :** Il devrait être possible de générer différents modèles transformés pour de multiples applications pour le même utilisateur, sans qu'aucune information ne soit déduite de la comparaison ou de la corrélation des différentes réalisations. Soit $B_z = \{f(b_z, K_z^1), .., f(b_z, K_z^Q)\}$, l'ensemble des Q modèles générés pour l'utilisateur z et K_z^i l'ensemble des paramètres concernant l'utilisateur z par rapport à la $i$ème révocation. B_z doit constituer un sous-ensemble aléatoire dans $\{0,1\}^{Q^*}$, ce qui veut dire, que la corrélation des différentes réalisations doit être la plus faible possible.

Ainsi, la définition claire des critères à évaluer est une étape importante pour aider au choix cohérent des métriques. Ces définitions peuvent être communes à tous les algorithmes de protection. Néanmoins, le formalise utilisé, est généralement dépendant de la construction générique en entrée. Ces métriques doivent être paramétrées par le modèle de l'adversaire, malicieux ou semi-honnête (cf. page 67). Relativement à la définition de nos objectifs de sécurité, nous associons le modèle de *l'adversaire malicieux* aux métriques qui décrivent la sécurité du système. Le modèle de l'adversaire *honnête mais curieux* est associé aux métriques qui concernent le respect de la vie privée. Dans certaines situations, nous pouvons considérer la situation d'un adversaire curieux qui peut devenir actif. L'information en possession de l'adversaire est résumée dans le tableau 2.2.

2.4.3 Détermination des métriques

En supposant avoir sélectionné la construction générique numéro 4 du tableau 2.2 relative aux transformations révocables dans le processus général d'évaluation de la figure 2.1, nous proposons un ensemble de métriques $A_1 \ldots A_{17}$.

1. **Critères sur les risques d'intrusion**
   - *A_1 : Taux d'erreur égal*
     Pour l'usage/le risque d'intrusion dans le système révocable nous considérons, respectivement, les mesures suivantes :

$$FRR_T(\epsilon) = P(D_T(f(b_z, K_z), f(\acute{b}_z, K_z)) > \epsilon) \tag{2.7}$$

$$FAR_T(\epsilon) = P(D_T(f(b_z, K_z), f(\acute{b}_z, K_z)) \leq \epsilon) \tag{2.8}$$

Pour l'évaluation, il est commun de mettre le seuil de décision ϵ à la valeur ϵ_D où $FRR_T(\epsilon_D) = FAR_T(\epsilon_D)$. Dans ce cas, la métrique A_1, est considérée comme la valeur EER lorsque $FRR_T(\epsilon_D) = FAR_T(\epsilon_D)$

- A_2 *: Dégradation des performances*
  Pour vérifier si l'efficacité n'est pas diminuée par l'utilisation du système de protection par rapport au système biométrique d'origine, nous proposons de calculer la mesure suivante :
  $$A_2 = 1 - \frac{AUC(FAR_T, FRR_T)}{AUC(FAR_O, FRR_O)} \tag{2.9}$$
  Avec AUC (Area Under the Curve), l'aire sous la courbe ROC des deux systèmes avec et sans protection. De nombreux cas sont intéressants à considérer. Tout d'abord, il peut arriver que A_2 soit égale à 0 reflétant un système révocable parfait (sans erreur i.e. $EER_T = 0\%$). D'autre part, si la valeur A_2 est négative, cela signifie que l'efficacité du système biométrique est détériorée par le système de protection. Sinon, au contraire, la protection améliore les performances.

  Nous analysons maintenant le critère d'intrusion compte tenu de la complexité des scénarios d'attaques possibles :

- A_3 *: Attaque à zéro effort (adversaire malicieux)*
  Dans ce scénario, l'imposteur tente d'usurper la véritable identité de l'utilisateur z en représentant ses propres données biométriques $\acute{b}_y$ avec des paramètres inconnus K_y. On aura alors : $A_z = f(\acute{b}_y, K_y)$.

- A_4 *: Attaque par force brute (adversaire malicieux)*
  Dans ce scénario, l'imposteur décide de suborner le module de protection en envoyant un modèle prêt à être comparé par le module de comparaison. Il estime aléatoirement différentes valeurs A de telle sorte que $A_z = A$.

- A_5 *: Attaque par vol de clé (adversaire malicieux)*
  L'imposteur a obtenu les paramètres K_z de l'utilisateur z et essaie différentes valeurs $\acute{b}$ pour générer : $Az = f(\acute{b}, K_z)$.

- A_6 *: Attaque par vol de biométrie (adversaire malicieux)*
  L'imposteur obtient $\acute{b}_z$ (directement ou après avoir fait le calcul à partir d'un échantillon biométrique compromis comme la trace d'une empreinte). Il essaie différentes valeurs de K pour générer : $A_z = f(\acute{b}_z, K)$. Cette attaque modélise aussi

les attaques connues par mascarade ou par biométrie truquée.

En utilisant la formule 2.6 où ϵ est remplacé par ϵ_D le seuil de décision du système révocable, les métriques $A_i, i = 3 \ldots 6$ sont calculées à partir de l'équation 2.10.

$$A_i = FAR_A(\epsilon_D), i = 3 \ldots 6 \tag{2.10}$$

2. **Critère de révocabilité et de renouvellement**
   Une simple métrique A_7 de type booléen peut être utilisée pour confirmer ou non la possibilité de révocation.

3. **Critère d'irréversibilité/divulgation partielle**
   Un processus d'inversion vise à récupérer la totalité ou une partie de l'information biométrique, susceptible de réussir la phase d'authentification.

   - *A_8 : Complexité de l'estimation par force brute*
     Il s'agit d'estimer la complexité en terme de nombre d'essais par force brute, nécessaires pour trouver la biométrie d'origine qui correspond, après transformation, au modèle transformé. Cette métrique estime cette complexité en nombre de bits.

   - *A_9 : Estimation de l'inverse*
     Il s'agit de mesurer la possibilité de déterminer, à partir du modèle transformé, le modèle biométrique d'origine. En supposant connus les paramètres K_z de l'utilisateur z et son modèle protégé $f(b_z, K_z)$, le but est d'estimer b_z en cherchant la fonction inverse f^{-1} de f. Cette estimation peut être exacte ou partielle de façon à ce que le modèle estimé puisse, quand même, réussir la phase de vérification dans l'espace d'origine (sans protection). Généralement, la preuve sur l'irréversibilité d'une fonction se fait de façon théorique. Dans le cadre d'un adversaire honnête mais curieux, nous pouvons considérer une métrique A_9 à trois valeurs : ITP (Inversion Totale Possible), IPP (Inversion Partielle Possible) et NI (Non Inversible). L'élaboration de la preuve d'irréversibilité est étroitement liée à la construction de la fonction de transformation et le but est alors d'essayer d'agir comme un cryptanalyste , c'est-à-dire, casser cette preuve. Dans le cadre d'un adversaire curieux qui décide d'être actif, la métrique A_9 peut être estimée par l'équation 2.11.

$$A_9 = P(D_O(b_z, f^{-1}(f(b_z, K_z), K_z)) \leq \epsilon_D) \tag{2.11}$$

   - *A_{10} : Estimation de l'inverse à partir de références multiples*
     Il s'agit de mesurer la possibilité de déterminer le modèle biométrique d'origine ou sa version partielle lorsque l'ensemble $B_z = \{f(b_z, K_z^1), .., f(b_z, K_z^Q)\}$ des

références multiples est connu en même temps que l'ensemble des clés correspondantes $\{K_z^1, ..., K_z^Q\}$.

Dans le cadre d'un adversaire honnête mais curieux, nous pouvons considérer une métrique A_{10} à trois valeurs : ITP (Inversion Totale Possible), IPP (Inversion Partielle Possible) et NI (Non Inversible). Dans le cadre d'un adversaire curieux qui décide d'être actif, la métrique A_{10} peut être estimée par l'équation 2.12.

$$A_{10} = P(D_O(b_z, f^{-1}(f(b_z, K_z), K_z)) \leq \epsilon_D) \tag{2.12}$$

- *A_{11} : Estimation d'une pseudo-inverse*
  Il s'agit de mesurer la possibilité d'estimer une approximation $\tilde{b_z}$ du modèle b_z telle que $\tilde{b_z}$ ne correspond pas à b_z (dans le domaine d'origine) mais la transformée $f(\tilde{b_z}, K_z)$ correpond à la transformée $f(b_z, K_z)$ (il s'agit pour l'intrus de réussir le test de vérification dans l'espace de transformation sans nécessairement correspondre au modèle biométrique d'origine). Le but est d'optimiser le problème d'inversion lorsque l'estimation de f^{-1} n'est pas évidente.
  Dans le cadre d'un adversaire honnête mais curieux, la métrique relative à ce critère est A_{11} à deux valeurs : OF (Optimisation Faisable) et ONF (Optimisation Non Faisable). Dans le cadre d'un adversaire curieux qui décide d'être actif, la métrique A_{11} peut être estimée par l'équation 2.13.

$$A_{11} = P(D_T(f(b_z, K_z), f(\tilde{b_z}, K_z)) \leq \epsilon_D) \tag{2.13}$$

- *A_{12} : Estimation de la pseudo-inverse à partir de références multiples*
  L'estimation de la pseudo-inverse peut être plus facile lorsque l'ensemble $B_z = \{f(b_z, K_z^1), .., f(b_z, K_z^Q)\}$ des références multiples est connu de l'adversaire.
  Dans le cadre d'un adversaire honnête mais curieux, la métrique relative à ce critère est A_{12} à deux valeurs : OF (Optimisation Faisable) et ONF (Optimisation Non Faisable). Dans le cadre d'un adversaire curieux qui décide d'être actif, la métrique A_{12} peut être estimée par l'équation 2.14.

$$A_{12} = P(D_T(f(b_z, K_z), f(\tilde{b_z}, K_z)) \leq \epsilon_D) \tag{2.14}$$

4. **Critère d'intraçabilité/diversité**
   Une caractéristique importante d'un système biométrique révocable est sa capacité à produire différentes références pour la même personne et pour différentes applications. Dans ce contexte, la préservation de la vie privée repose sur le fait que ces références soient suffisamment aléatoires pour prévenir d'un possible lien entre elles. Nous commençons par proposer des métriques qui mesurent la complexité des

attaques possibles.

- A_{13} *: Taux de compatibilité croisée*
  Une façon courante d'évaluer l'intraçabilité est de comparer, entre elles, différentes références obtenues à partir du même échantillon biométrique après avoir assigné à chaque utilisateur t différentes clés K_z. Nous appelons la métrique A_{13}, le taux de compatibilité croisée qui représente le pourcentage des correspondances réussies. A_{13} est ensuite calculée à partir de l'équation 2.10.

- A_{14}, A_{15} *: Attaque par écoute*
  Un imposteur ne doit pas être en mesure d'extraire des informations à partir des différents modèles délivrés pour le même utilisateur. Un imposteur peut intercepter N modèles différents dans la perspective de prédire une valeur de modèle admissible (en fixant, par exemple, chaque bit compte tenu de la plus haute valeur de probabilité). A partir de l'équation 2.10, nous calculons A_{14} pour $N = 3$ et A_{15} pour $N = 11$. Un nombre plus élevé ne semble pas être réaliste.

Nous évaluons maintenant la propriété de diversité d'un point de vue théorie de l'information.

- A_{16} *: Information mutuelle*
  Afin de mesurer la propriété de diversité, nous proposons de calculer l'information mutuelle entre les modèles révoqués. Selon l'équation 2.15, l'information mutuelle est une valeur qui mesure la quantité d'information commune entre deux variables aléatoires. Elle est nulle si les deux variables sont complètement indépendantes.

$$I(X,Y) = \sum_{x}\sum_{y} P(x,y) log(\frac{P(x,y)}{P(x)P(y)}) \tag{2.15}$$

  Avec X et Y deux variables aléatoires et P la distribution de probabilité jointe du couple (X, Y).
  Afin de mesurer la propriété de la diversité, nous quantifions la valeur la plus élevée de l'information mutuelle entre les différents modèles transformés pour chaque individu. La métrique A_{16} est ensuite calculée en utilisant la valeur moyenne pour tous les utilisateurs de la plus haute valeur de l'information mutuelle, selon l'équation 2.16.

$$A_{16} = \frac{1}{N}\sum_{z}\sum_{j=1}^{M} max(I(f(b_z, k_z), f(b_z, K_z^j))) \tag{2.16}$$

Avec :
b_z : le modèle transformé de l'utilisateur z de la base de données.
K_z^j : indique le paramètre j de l'utilisateur z.
N : le nombre d'individus dans la base de données.
M : le nombre de modèles générés pour chaque individu. Une valeur réaliste de M est 10.
P : l'estimation de la loi de probabilité.

- *A_{17} : L'entropie de diversité*
  En terme général, l'entropie est une mesure usuelle de la sécurité. Dans la cryptanalyse, l'entropie est considérée comme une mesure standard de l'imprévisibilité de la clé de chiffrement. L'entropie ou l'incertitude est maximale lorsque tous les éléments de la clé sont identiquement et indépendemment distribués. Une technique commune pour calculer l'entropie d'une variable aléatoire X est d'appliquer la formule 2.17.

$$H(X) = -\sum_i p_i log_2(p_i) \tag{2.17}$$

  où p_i représente la probabilité d'occurrence de la i^e valeur possible de la variable X. Des variantes de cette formule ont été utilisées pour mesurer la quantité d'information contenue dans une représentation ou un modèle biométrique, avant ou après sa transformation. Nous citons par exemple, l'entropie minimale présentée dans [DRS04] pour quantifier l'incertitude d'une représentation binaire en fonction de la plus grande probabilité d'apparition dans la représentation. Des estimations empiriques de l'entropie ont aussi été explorées dans le domaine de la biométrie comme la mesure d'effort présentée dans [NJ09]. Cette mesure est relative aux empreintes digitales et consiste à estimer le nombre d'essais nécessaires pour trouver le premier ensemble des minuties qui réussit le test de vérification.

  Pour mesurer la diversité en terme d'entropie, nous supposons qu'un attaquant va essayer de prédire un modèle accepté après l'écoute des M modèles du véritable utilisateur. L'entropie va nous permettre de savoir si la prédiction du $(M+1)^e$ modèle est possible. Soit $B_z = \{f(b_z, K_z^j)\}$, $j = 1 \ldots M$, l'ensemble des M modèles générés pour l'utilisateur z. Nous supposons que chacun des éléments dans B_z est un vecteur binaire. Nous commençons par étudier la distribution des distances de Hamming entre les modèles de l'ensemble B_z. Nous répétons ce calcul pour les N utilisateurs de la base de données (il s'agit d'obtenir la distribution pseudo-imposteur).

Soit x la variable aléatoire qui représente cette distribution (la distribution des nombres de bits différents entre deux vecteurs modèles). Si la distribution de x peut être approximée par une distribution binomiale alors le calcul de l'entropie de diversité peut se faire. L'épreuve de Bernoulli est l'évènement que deux bits de même position soient égaux ou non et le nombre d'essais est égal à la longueur n des vecteurs de B_z. Idéalement, l'entropie de chaque vecteur de l'ensemble B_z est égale à la longueur n. Cependant, si une corrélation existe entre les différents éléments des vecteurs $B_z = f(b_z, K_z^j)$, $j = 1 \ldots M$, cette entropie diminue. D'après Daugman [Dau03], lorsque différentes réalisations de Bernoulli sont corrélées, la distribution constitue toujours une loi binomiale mais avec une diminution de la valeur n qui sera dans ce cas égale au degré de liberté de la loi binomiale. En s'inspirant des travaux de Daugman [Dau03], nous proposons d'estimer l'entropie de diversité comme étant le degré de liberté de la distribution statistique de la variable x (qui représente la distribution pseudo-imposteur). La métrique A_{17} d'entropie de diversité peut être estimée en utilisant l'équation 2.18.

$$A_{17} = \frac{P(1-P)}{\sigma^2} \tag{2.18}$$

Où P est la valeur de probabilité des réalisations de Bernoulli et σ^2 est la valeur de variance de la distribution de x.

x étant une loi binomiale, sa moyenne est égale à $M \times P$. Réciproquement, P peut être directement déduite à partir des valeurs moyennes de x. Ce calcul d'entropie pour mesurer la diversité du modèle transformé a l'avantage de se faire directement par rapport à la distribution pseudo-imposteur. Il diffère donc considérablement des mesures classiques d'entropie utilisées dans les différents travaux sur la question [BKS10, ZWBK09b, ZKVB11] où de plus, l'étude d'entropie n'est pas faite sur les modèles diversifiés mais uniquement par rapport à une seule occurrence du modèle. Un autre avantage est que ce calcul est applicable pour toutes les transformations possibles car il est indépendant de la fonction de transformation et de la représentation des modèles transformés. Cette entropie a donc la propriété d'être mesurable, contrairement à beaucoup d'autres qui le sont moins facilement.

2.5 Conclusion

Dans ce chapitre, nous avons spécifié un procédé général pour évaluer les schémas de protection des modèles biométriques. En raison de l'absence d'une construction générique

pour tous les algorithmes existants, nous avons proposé de paramétrer ce procédé par l'une des quatre constructions génériques que nous avons pu déduire de notre recherche bibliographique (voir tableau 2.2). Cette paramétrisation nous permet ensuite de construire les métriques spécifiques à l'algorithme en question. Les métriques proposées quantifient le système de protection en termes de sécurité et de préservation de la vie privée sur deux niveaux différents. D'abord, par identification des scénarios d'attaque, les métriques mesurent la complexité de calcul en terme d'effort de l'adversaire. Ces métriques sont complétées par des mesures spécifiques, propres à la théorie de l'information. Leur but est d'étudier la faisabilité de la construction par rapport à certains critères. La figure 2.2 résume ce processus d'évaluation pour les algorithmes par transformation révocable.

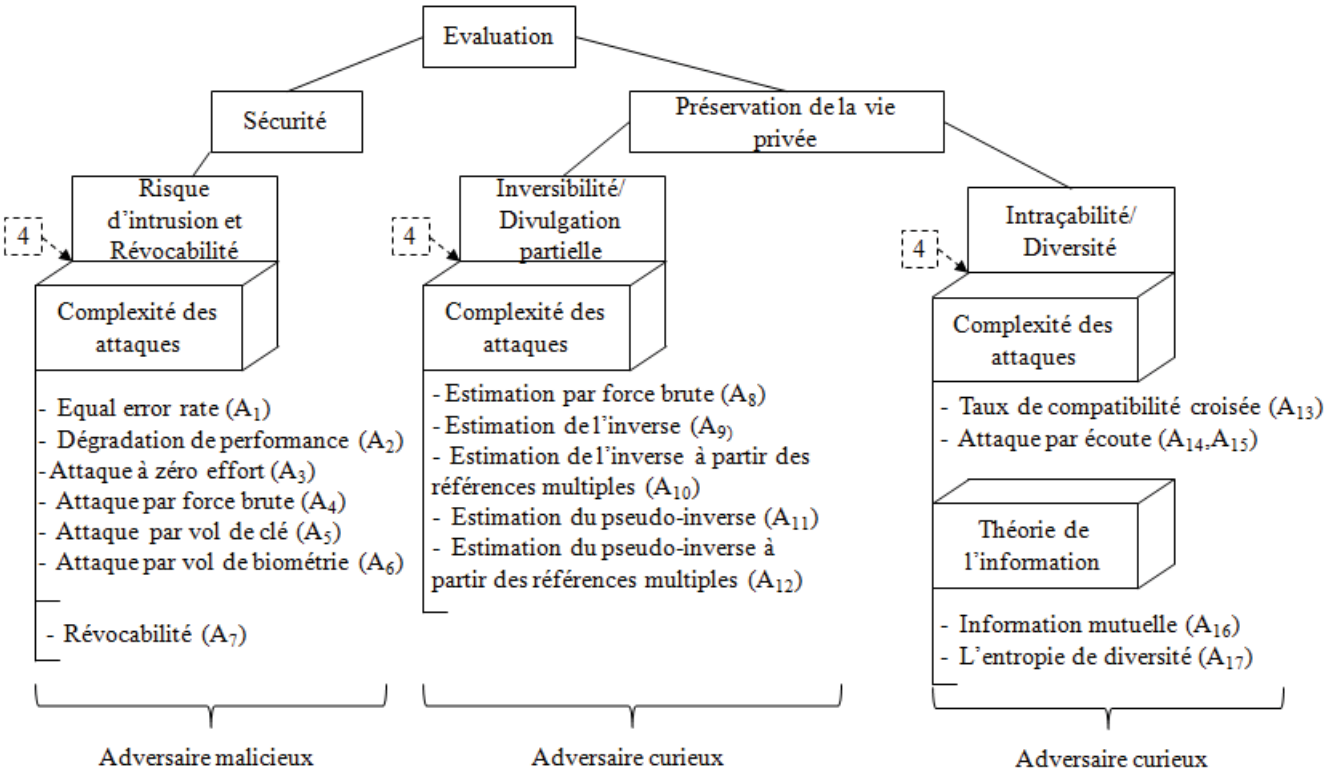


FIGURE 2.2 – Cadre opérationnel pour l'évaluation des transformations révocables où le paramètre de sélection de la construction générique est à 4

Nous pouvons évaluer notre proposition en la confrontant à la base d'attaques présentées dans le section 2.2. Le tableau 2.4 identifie les liens entre les métriques proposées et l'attaque en question. Nous remarquons que ces métriques prennent bien en charge les attaques mentionnées. Néanmoins, nous souhaitons que notre modèle soit facilement extensible et non figé. Le rajout d'autres métriques est donc possible. Ces métriques seront utilisées dans les chapitres suivants où il est question de présenter des schémas révocables d'empreintes digitales.

	Attaque à zéro effort [JNN08]	Attaque par vol de clé [KCZ+05]	Attaque par traçabilité [SB07]	Estimation inverse	Estimation pseudo-inverse [NJ09]	Attaque par références multiples [LH13]	Attaque par force brute	Attaque par écoute	Attaque par vol de biométrie [Hil01]
A1	★	★							
A2	★								
A3	★								
A4							★		
A5		★							
A6									★
A8							★		
A9				★					
A10						★			
A11					★				
A12						★			
A13			★						
A14,A15								★	
A16					★				
A17					★				

TABLE 2.4: Evaluation des métriques proposées

Chapitre 3

Schémas révocables d'empreintes digitales

Dans ce chapitre, nous proposons un nouveau schéma révocable d'empreintes digitales. L'élaboration de cette solution s'appuie sur un ensemble de tests et d'études comparatives qui se dessinent au fur et à mesure du chapitre. Cette démarche nous permet d'argumenter et de valider chacun de nos choix.

Sommaire

3.1 Introduction

L'exploration de nouveaux schémas révocables pour les empreintes digitales est un sujet en pleine expansion. Ce chapitre propose des schémas de protection en considérant différents modèles d'empreintes digitales. Un modèle ou un descripteur est une représentation discriminante qui peut être utilisée pour la vérification. Globalement, il existe deux catégories de descripteurs pour les empreintes digitales : *les descripteurs globaux* et *les descripteurs locaux*. La plupart des algorithmes de reconnaissance mettent l'accent sur l'extraction et la comparaison des minuties. Cependant, comme indiqué par Maltoni

et al. dans [MMJP09], il est parfois utile de chercher d'autres traits discriminants sur l'empreinte, au-delà des minuties. Les principales raisons sont résumées comme suit :

- Il est très difficile d'extraire de manière précise les minuties sur une empreinte de mauvaise qualité. Dans ce cas, les minuties n'offrent pas le meilleur compromis entre précision et robustesse.
- Les processus d'extraction et de comparaison des minuties sont coûteux en terme de temps de calcul. Cela n'est pas visible sur nos puissants calculateurs d'aujourd'hui (bien que dans les systèmes d'identification à grande échelle, cela peut constituer une véritable contrainte) mais cette préoccupation demeure, en raison de l'intérêt croissant d'intégrer des algorithmes de reconnaissance biométrique sur des dispositifs embarqués et autonomes.
- Étant donné l'ensemble des minuties, il est possible de reconstruire l'image originale et de créer une fausse empreinte qui peut tromper le système, comme détaillé dans [CLMM07b]. Ce n'est pas encore le cas pour d'autres descripteurs comme les caractéristiques de texture.

Dans certains cas, les descripteurs globaux se présentent comme une alternative aux minuties. Ils utilisent d'autres caractéristiques de l'empreinte comme l'image d'orientation, le motif des crêtes ou l'information de texture qui existe sur l'empreinte. Nous citons les travaux de [JPHP00, JLS04, NL09], à titre illustratif des méthodes d'appariement d'empreintes en utilisant des descripteurs globaux.

Dans la section 3.2, nous présentons un schéma révocable pour des descripteurs globaux d'empreintes. Pour améliorer les résultats, nous étendons dans la section 3.3 ce schéma pour les minuties. Une conclusion fera le point sur ce chapitre.

3.2 Schéma révocable basé descripteurs globaux d'empreintes digitales

Nous commençons notre étude par une analyse de texture de l'empreinte digitale. Nos motivations par rapport à l'utilisation de l'information de texture comme descripteur d'empreintes sont multiples :

1. Dans certains schémas de protection, il est assez intéressant d'avoir une représentation vectorielle de l'empreinte qui a l'avantage d'être stable en ordre et en taille. L'utilisation des descripteurs globaux, comme la texture, aboutit le plus souvent à une telle représentation qui de plus peut être invariante aux distorsions géométriques.
2. Un descripteur vectoriel est plus facile à intégrer sur un dispositif embarqué car le

module de comparaison peut se réduire à un simple calcul de distance entre vecteurs. Dans le cas où le vecteur a pu être binarisé, il s'agira d'une simple distance de Hamming.

3. Les attributs de texture sont plus sécurisés par rapport aux minuties en terme de falsification de l'empreinte car ils ne permettent pas la reconstruction de l'image originale.
4. L'extension vers les minuties est possible, puisque nous pouvons chercher des comparaisons locales entre minuties en dérivant l'information de texture autour de chacune d'elles.

3.2.1 Etude comparative sur les descripteurs de texture d'empreintes digitales

La texture donne une information sur l'arrangement spatial des couleurs et des nuances sur une image. L'étude quantitative de la texture mesure cet arrangement sur toute ou une partie de l'image. Pour savoir quel indice de texture est adapté, il n'y a pas vraiment de règles. L'idée générale est d'opérer un traitement spécifique pour capturer la texture et de la quantifier ensuite en utilisant, le plus souvent, des mesures statistiques comme le calcul d'histogramme. La référence [TJ98] présente un résumé sur les méthodes d'analyse de texture.
Nous souhaitons maintenant étudier différents opérateurs de texture pour choisir le plus approprié. Nous listons plusieurs techniques récentes largement utilisées dans les problèmes de vision par ordinateur. Principalement, les filtres de Gabor et l'opérateur LBP (Local Binary Pattern) avec un ensemble de ses variantes.

Les fonctions de Gabor sont un outil d'analyse de texture très prisé. Elles ont des propriétés sélectives en orientation et en fréquence. Elles ont donc une résolution conjointe spatiale/fréquentielle optimale (c'est-à-dire qu'elles sont très performantes pour sélectionner à la fois une fréquence et une orientation). D'un point de vue mathématique, un filtre de Gabor est le produit d'une sinusoïde complexe et d'une enveloppe gaussienne. Un filtre 2D complexe dans le domaine spatial (x, y) de l'image est défini par l'equation 3.1.

$$G(x,y) = \exp(-\frac{1}{2}[\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}]) \times \exp(i(2\pi F x_\theta + \phi)) \tag{3.1}$$

Avec :

- $\begin{pmatrix} x_\theta \\ y_\theta \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$
- F la fréquence spatiale de la sinusoïde et ϕ la phase de la sinusoïde
- σ_x et σ_y qui représentent, respectivement, l'écart-type de l'enveloppe gaussienne le long des axes x et y

La composante réelle du filtre, souvent utilisée en pratique, est donnée par l'équation 3.2 et sa réponse impulsionnelle est illustrée sur la figure 3.1.

$$G(x,y) = \exp(-\frac{1}{2}[\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}]) \times \cos(2\pi F x_\theta + \phi) \tag{3.2}$$

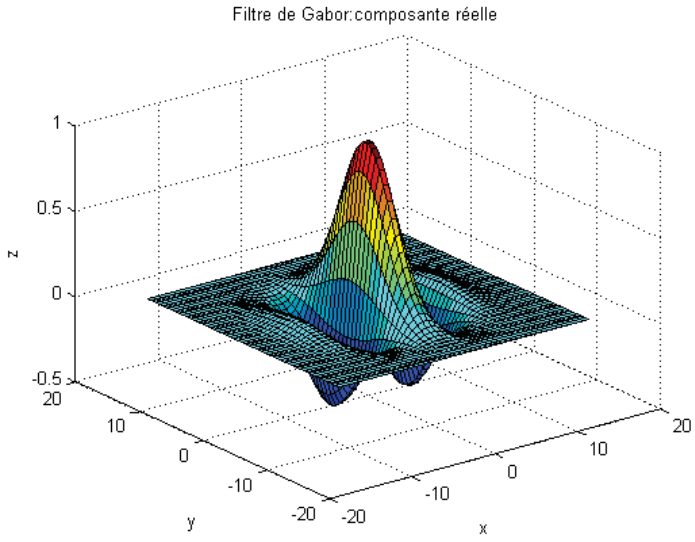
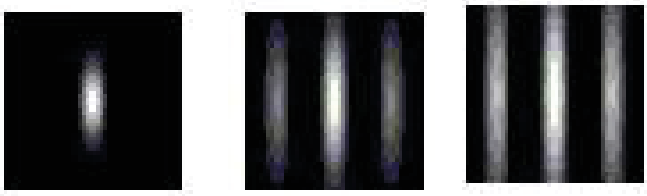


FIGURE 3.1 – Réponse impulsionnelle du filtre de Gabor symétrique

L'application de ce filtre sur une image I se fait par produit de convolution (voir équation 3.3) entre chaque pixel $I(i,j)$ de l'image et le masque M du filtre de Gabor.

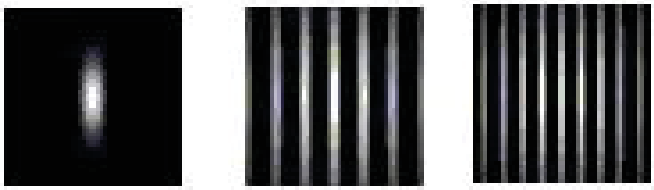
$$I(i,j) = \sum_{n=\frac{-(d-1)}{2}}^{\frac{d-1}{2}} \sum_{m=\frac{-(d-1)}{2}}^{\frac{d-1}{2}} I(i+n, j+m)M(n,m) \tag{3.3}$$

Le masque M est une matrice carrée de dimension impaire d calculée à partir de l'équation 3.2. Nous illustrons sur les figures 3.2, 3.3, 3.4, en niveaux de gris, plusieurs masques de Gabor différemment paramétrés. Sur toutes les figures, nous avons $\sigma = \sigma_x = \sigma_y$.



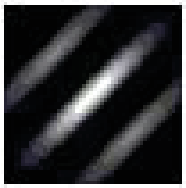
(a) (b) (c)

FIGURE 3.2 – Masque de Gabor sur différentes échelles. (a). $\sigma = 4$, (b). $\sigma = 8$, (c). $\sigma = 12$



(a) (b) (c)

FIGURE 3.3 – Masque de Gabor sur différentes fréquences. (a). $F = 0.1$, (b). $F = 0.2$, (c). $F = 0.3$



(a)

FIGURE 3.4 – Masque de Gabor orienté à $\pi/4$

Nous remarquons, par exemple, que la fréquence agit sur la largeur des contours à mettre en évidence. Pour les empreintes, il est commode de fixer F à la distance inter-crêtes.

L'impact de la variance gaussienne apparait plus clairement sur la figure 3.5, où une empreinte a été filtrée par différents masques de Gabor. Sur l'image *(b)*, nous remarquons que les crêtes orientées à $3\pi/4$ apparaissent plus clairement que les autres. Une meilleure sélection de σ sur l'image *(c)* a pu rajouter un effet de lissage qui a estompé la partie inutile de l'image. Un mauvais réglage de la fréquence sur l'image *(d)* montre la divergence du filtrage. Pour détecter toute la texture de l'image, on lui applique généralement un ensemble de filtres de Gabor paramétrés à différentes fréquences et orientations, appelé *banc de filtres*.

Dans cette étude comparative, nous proposons d'appliquer trois bancs de filtres de Gabor sur toute l'image d'empreinte pour obtenir trois différents descripteurs.

(a) (b) (c) (d)

FIGURE 3.5 – Image d'empreinte filtrée avec un masque de Gabor orienté à $3\pi/4$. (a). Image d'origine, (b). Masque de Gabor avec $F = 0.1$ et $\sigma = 2$ (c). Masque de Gabor avec $F = 0.1$ et $\sigma = 4$, (d). Masque de Gabor avec $F = 0.2$ et $\sigma = 4$

Nous paramétrons, respectivement, le couple *(Nombre de fréquences, Nombre d'orientations)* aux valeurs suivantes : $(4, 8)$, $(8, 8)$, $(16, 8)$. L'algorithme *1* illustre le calcul des descripteurs de Gabor. Le produit de convolution est effectué dans le domaine fréquentiel (il se réduit à une simple multiplication). La réponse du filtrage est représentée par sa moyenne et sa variance. Un descripteur consiste en l'ensemble des moyennes et des variances de toutes les réponses du banc de filtres. Nous appelons, respectivement, *Gabor*64, *Gabor*128, *Gabor*256 les trois descripteurs générés.

Algorithme 1 : Génération du descripteur de l'empreinte par un banc de filtres de Gabor

Input : I l'image en entrée
$Freq$ le nombre de fréquences du banc
$orientations$ le nombre d'orientations du banc

$A \leftarrow fft2(I)$ % avec $fft2(I)$ la transformée de Fourrier rapide de l'image I
for $s \leftarrow 1$ **to** $Freq$ **do**
  for $n \leftarrow 1$ **to** $orientations$ **do**
    $M \leftarrow$ Masque de Gabor calculé en utilisant l'équation 3.2
    $GW \leftarrow fft2(M)$
    $D \leftarrow abs(ifft2(A \times GW))$
    $Feature(1, (s-1) \times orientations + n) \leftarrow mean(D)$
    $Feature(2, (s-1) \times orientations + n) \leftarrow variance(D)$

L'opérateur LBP a récemment prouvé son efficacité dans diverses applications de la vision par ordinateur comme la segmentation. Cet opérateur, tel que proposé par Ojala *et al.*[TO96], consiste en trois étapes (voir la figure 3.6) :

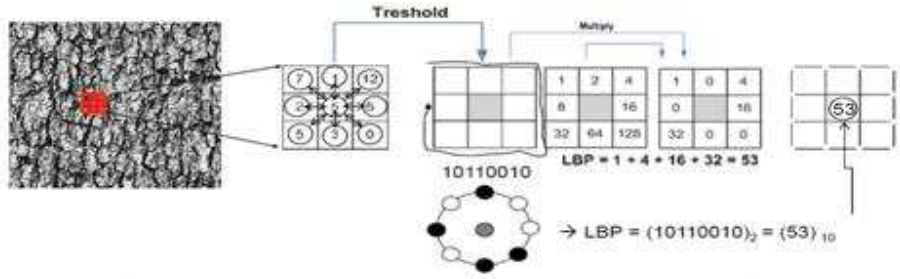


FIGURE 3.6 – Opérateur LBP de base

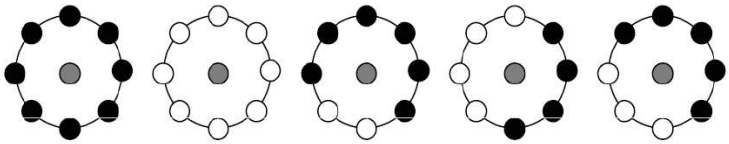
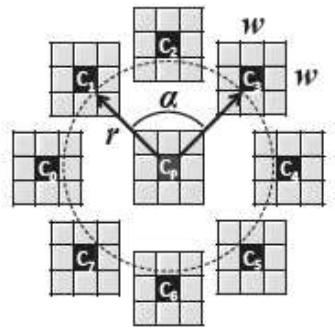


FIGURE 3.7 – Exemple de micro-motifs LBP

1. Seuillage de chaque voisinage 3×3 en considérant la valeur d'intensité du pixel central comme la valeur du seuil. Une valeur de 1 est assignée à chaque pixel dans l'intensité est supérieure ou égale au seuil, 0 autrement. Une chaine binaire de 8 bits est alors déduite.
2. La chaine binaire est convertie, par pondération, en un nombre décimal qui est assigné au pixel central. Il existe 256 valeurs possibles pour représenter les différents micro-motifs possibles. Un exemple de ces micro-motifs est représenté par la figure 3.7.
3. Un histogramme de 256 valeurs comptabilise le nombre d'occurrences de chaque micro-motif pour tous les pixels de l'image. L'empreinte digitale est représentée par cet histogramme.

L'approche LBP codifie et collecte, en un histogramme, les occurrences des différents micro-motifs représentant la texture du voisinage local.

Comme précédemment, cette méthode assigne aussi à chaque pixel de l'image une valeur binaire sur 8 bits. Cependant, tel que proposé dans [WHT08], ce calcul se fait sur une structure plus complexe dans le but d'avoir des informations additionnelles sur la texture locale. Des régions sous forme de blocs 3×3 appelées *patch* sont sélectionnées dans le voisinage circulaire du pixel central (voir figure 3.8).



(a)

$$
\begin{aligned}
&= \\
&f(d(C_0,C_p)-d(C_2,C_p))2^0+ \\
&f(d(C_1,C_p)-d(C_3,C_p))2^1+ \\
&f(d(C_2,C_p)-d(C_4,C_p))2^2+ \\
&f(d(C_3,C_p)-d(C_5,C_p))2^3+ \\
&f(d(C_4,C_p)-d(C_6,C_p))2^4+ \\
&f(d(C_5,C_p)-d(C_7,C_p))2^5+ \\
&f(d(C_6,C_p)-d(C_0,C_p))2^6+ \\
&f(d(C_7,C_p)-d(C_1,C_p))2^7
\end{aligned}
$$

(b)

FIGURE 3.8 – Le code PBLBP. (a). Structure du voisinage avec $\alpha = 2$ et $S = 8$. (b). Calcul du code. [WHT08]

Sur cette structure, il existe S patchs $C_1, ..., C_S$ uniformément distribués sur un cercle de rayon r. Le patch central est noté C_p. Un code de S bits est assigné au pixel central suivant la formule de l'équation 3.4 (voir aussi un exemple de calcul sur la figure 3.8).

$$code = \sum_{i=1}^{S} f(d(C_i, C_p) - d(C_{(i+\alpha) mod S}, C_p))2^i \tag{3.4}$$

Dans cette équation :

- C_p est le patch du pixel central
- α est un paramètre de décalage entre 2 patches.
- $d(.,.)$ fonction de distance entre deux patches (e.g. La norme des différences de leurs niveaux de gris)
-

$$f(x) = \begin{cases} 1 & x \geq 0 \\ 0 & sinon \end{cases}$$

La texture est ensuite représentée par l'histogramme de toutes les chaines binaires possibles.

Comme expliqué dans [HZ08], LRS est un opérateur qui utilise une représentation symbolique du voisinage local. Pour chaque pixel, une chaine est assignée représentant les valeurs de comparaison avec ses pixels voisins. Si nous considérons une structure à 4-connexions, et les opérateurs de comparaison entre deux pixels parmi {>, =, <}, alors le nombre maximal de chaines possibles pour représenter un pixel est $3^4 = 81$ chaines. L'opérateur LRS codifie la fréquence d'apparition des chaines de symboles en un histogramme.

Résultats expérimentaux

L'objectif de cette partie étant d'étudier le meilleur descripteur de texture parmi ceux que nous avons considérés, il suffit alors de suivre une *approche holistique* pour leur génération. Cela consiste à considérer toute l'image I comme entrée des méthodes d'analyse citées ultérieurement sans traitement spécifique comme : la sectorisation par blocs ou la recherche d'invariant géométrique. Néanmoins, nous identifions 4 scénarios possibles pour la génération des descripteurs suivant le pré-traitement subi par l'image I comme le montre la figure 3.9 :

(a) (b) (c) (d)

FIGURE 3.9 – Pré-traitement de l'image d'empreinte en entrée. (a). Image d'origine, (b). Région d'intérêt (ROI), (c). Image binarisée, (d). Région d'intérêt binarisée

- I est l'image originale contenant l'empreinte digitale.
- I est l'image après binarisation entre crêtes et vallées.
- I est la de région d'intérêt (ROI) sans l'information de fond.
- I est la région d'intérêt après binarisation.

Nous effectuons les tests sur la base de données $FVC2002 - DB2$ [1] fournie publiquement dans le cadre de la compétition internationale pour la vérification d'empreintes digitales. Les images sont acquises avec un capteur optique d'une résolution de 569 *dpi*, générant des images de 560×296 pixels. Elle est composée de 800 images pour 100 individus et 8 échantillons par individu. Nous réalisons les comparaisons à la base des deux tests suivants :

- **Le test de performance** : nous mesurons la précision des différents descripteurs de texture en terme du taux d'erreur. Le calcul du score de comparaison est basé sur la distance de Minkowski. Durant les expérimentations, nous utilisons le protocole de l'enrôlement unique. Le premier échantillon de chaque individu est utilisé comme référence et les autres sont utilisés pour les tests. Nous aurons $7 \times 100 = 700$ scores de comparaison pour la distribution intra-classe et $(99 \times 100)/2 = 4950$ scores de comparaison pour la distribution inter-classe. Nous calculons les deux mesures FAR et FRR qui varient suivant les seuils de décision. Le courbe DET est utilisée pour illustrer cette variation. La performance sera ensuite caractérisée par le taux d'erreur égal EER. Nous illustrons sur l'algorithme 2 la manière de calculer cet EER. En effet, nous avons remarqué que dans la communauté large qui traite de la biométrie, des erreurs non négligeables sur le calcul de cet EER s'introduisent.

Algorithme 2 : Estimation de l'EER

Input : $FAR(t)$ les différentes valeurs du FAR en fonction des seuils de décision
$FRR(t)$ les différentes valeurs du FRR en fonction des seuils de décision

$D \leftarrow Inf$
for $pas \leftarrow 1$ **to** max **do**
  if $abs(FAR(pas) - FRR(pas)) < D$ **then**
    $D \leftarrow abs(FAR(pas) - FRR(pas))$
    $EER \leftarrow (abs(FAR(pas) - FRR(pas)))/2$

- **Le test de redondance** : ce test est important pour mesurer le taux d'information réellement présente dans le descripteur. Nous estimons ce taux de redondance par l'analyse en composantes principales (ACP). En ne gardant que 99.9% de l'information, la nouvelle dimension de l'espace calculée à partir des 800 descripteurs de la base de données, permet d'estimer la redondance. Par exemple, après l'application de l'ACP, la taille du descripteur LBP (initialement à 256) diminue à 38, la redondance est estimée à $(256 - 38)/256 = 85\%$.

1. http ://bias.csr.unibo.it/fvc2002/

Nous commençons par comparer les trois descripteurs de Gabor entre eux : *Gabor*64, *Gabor*128 et *Gabor*256. A partir de la courbe *DET* de la figure 3.10, nous obtenons, respectivement, les *EER* suivants : 34.42%, 29.04% et 28.11%. Cela informe que le descripteur *Gabor*256 est le plus adapté pour cette base de données.

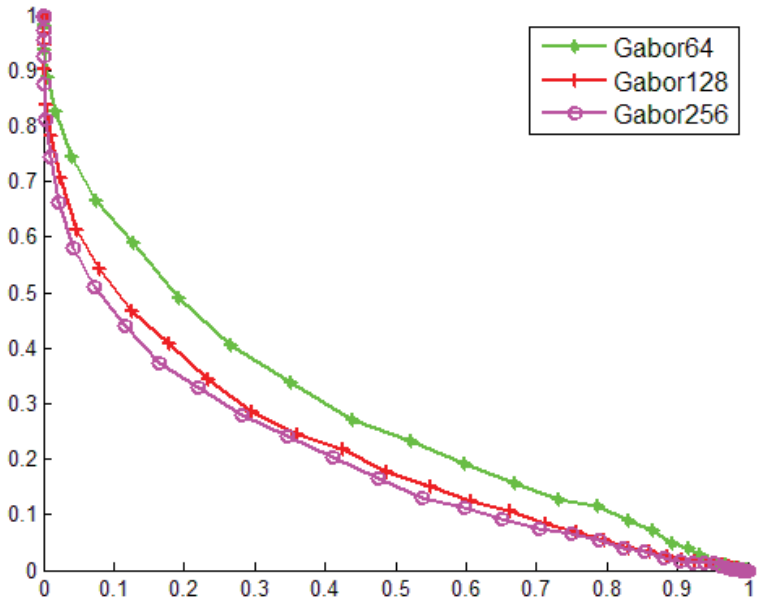


FIGURE 3.10 – Courbe DET pour les différents descripteurs de Gabor

Soient : *LBP*, *PBLBP*, *Gabor*256 et *LRS* les quatre descripteurs à comparer. Le tableau 3.1 résume les EERs obtenus en considérant les différents scénarios de tests.

	LBP	PBLBP	Gabor256	LRS
Image originale	28.26	39.97	28.11	33.3
Image binarisée	29.93	37.20	20.26	31.20
ROI	38.09	42.08	27.67	37.75
ROI binarisée	31.73	39.56	**19.34**	32.48

TABLE 3.1: Résultats de l'étude comparative sur l'analyse de texture en terme d'EER(%) par rapport aux différents scénarios de tests

Pour la plupart des descripteurs, les meilleurs résultats sont obtenus sur les images binarisées. Les filtres de Gabor permettent d'obtenir les meilleurs résultats. Les différentes courbes DET de la figure 3.11 montrent également que la courbe de Gabor est à chaque fois sous les autres courbes. L'utilisation de l'image binarisée par rapport à l'image originale permet un gain énorme, de 9% environ en considérant les filtres de Gabor.

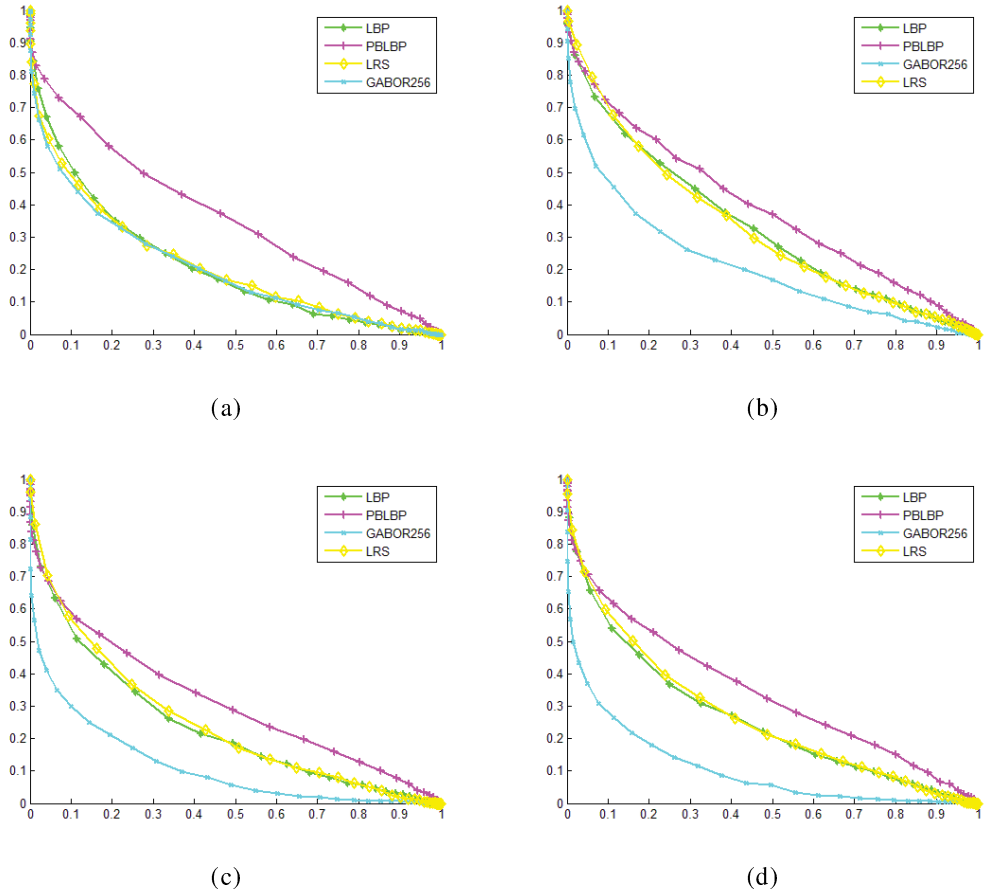


FIGURE 3.11 – Les différentes courbes DET de l'étude comparative. (a). Image originale, (b). Image binarisée, (c). ROI, (d). ROI binarisée

Le tableau 3.2 résume les résultats du calcul de redondance de chaque descripteur. Comme nous pouvons le constater, presque tous les descripteurs ont une forte redondance. Les paramètres de Gabor présentent la plus faible redondance. Ils ont de plus l'avantage d'être ajustables en terme de taille.

	Dimension	Redondance
LBP	256	85%
PBLBP	7280	89%
Gabor64	64	**9%**
Gabor128	128	23%
Gabor256	256	38%
LRS	81	79%

TABLE 3.2: Les descripteurs de texture : dimension et redondance

Conclusion de l'étude comparative

Si l'on considère l'état de l'art sur les systèmes de vérification d'empreintes, les résultats obtenus, en raison entre autres de l'approche holistique, apparaissent pauvres. Néanmoins, cela est suffisant pour distinguer les filtres de Gabor comme l'approche souhaitable, parmi celles citées, pour analyser les textures d'empreintes. Dans la suite de cette section, nous présentons un schéma révocable à partir des descripteurs de Gabor de l'empreinte digitale.

3.2.2 Création du descripteur global d'empreintes digitales

La création du descripteur global d'empreintes digitales se fait suivant les principales étapes explicitées dans ce qui suit et résumées par la figure 3.12.

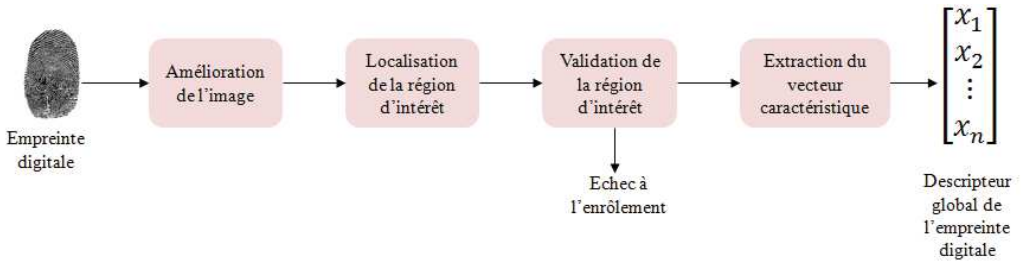


FIGURE 3.12 – Processus d'extraction du descripteur global d'empreintes digitales

1. **Amélioration de l'image**
   Comme précisé dans le chapitre 1, l'étape d'extraction peut être significativement altérée si l'image en entrée est de mauvaise qualité. Un processus d'amélioration de l'image s'impose dans ce cas. Nous utilisons la technique proposée par Chikkerur *et al.* [CCG07]. Cette technique se base sur le filtrage de l'image à partir de filtres directement conçus dans le domaine fréquentiel. Ces filtres ont été, initialement, proposés par Sherlok *et al.* [SMM94]. Il s'agit de filtres passe-bande directionnels, exprimés en utilisant les coordonnées polaires (ρ, ϕ) par la fonction : $H(\rho, \phi) = H(\rho)H(\phi)$. $H(\rho)$ est la réponse radiale du filtre qui dépend de la fréquence ρ et $H(\phi)$ est sa réponse angulaire qui dépend de l'orientation du filtre. Le filtrage proposé dans [CCG07], opère par blocs d'images. Chaque bloc B est représenté par le couple (ρ, ϕ), où ρ est la fréquence spatiale des crêtes dans le bloc B et ϕ et leur orientation dominante. L'estimation du couple (ρ, ϕ) se fait à partir de l'analyse du spectre de la transformée de Fourier discrète (TFD) du bloc B . En effet, Chikkerur *et al.* [CCG07] exploitent le fait que le spectre d'une TFD se compose de deux impulsions dont

sa distance par rapport à l'origine indique la fréquence angulaire et sa localisation indique l'orientation de l'onde. L'algorithme 3 résume le fonctionnement du filtrage. La figure 3.13 en illustre le résultat.

Algorithme 3 : Amélioration de l'image dans le domaine fréquentiel

Input : B Le bloc de l'image à traiter
$F_1, ..., F_N$ L'ensemble des filtres directionnels espacés de 15°, réglés à la même fréquence, et calculés par la méthode de Sherlok *et al.*[SMM94]

$\grave{B} \leftarrow fft2(B)$ Obtenir le TFD du bloc
L'analyse TFD permet d'obtenir l'orientation locale ϕ
$AF \leftarrow$ le filtre le plus proche de ϕ dans l'ensemble $F_1, ..., F_N$
$\grave{B} \leftarrow \grave{B}.AF$
$B \leftarrow ifft2(\grave{B})$ reconstruction du bloc aprés filtrage

(a) (b)

FIGURE 3.13 – Résultat de l'étape d'amélioration. (a). Image originale, (b). Image améliorée

2. **Localisation de la région d'intérêt (ROI)**
   La localisation de la région d'intérêt est une étape essentielle pour améliorer les résultats précédents du descripteur de Gabor ($EER = 19.34\%$). Elle a comme objectif de :
   - Sélectionner un point de référence pour prendre en charge les problèmes de translation et/ou de rotation.
   - Calculer le descripteur à partir d'une sectorisation de l'image (contrairement à l'approche holistique) ce qui permet de mieux gérer l'information contextuelle de l'empreinte.

Nous choisissons de commencer par l'estimation du point core qui sera considéré comme le point central de l'image. Jain *et al.* [JPHP00] proposent de l'estimer à partir du champ d'orientation de l'image comme l'illustre la figure 3.14.

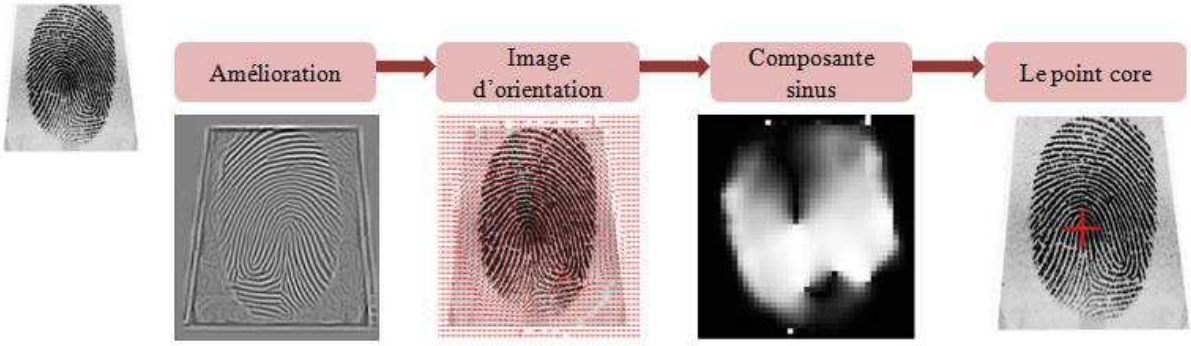


FIGURE 3.14 – Processus d'extraction du point core

La structure des crêtes définit, sur chaque bloc de l'image, une orientation dominante le caractérisant. L'orientation locale du pixel (x, y) est l'angle $\theta_x \in [0..\pi[$ que les crêtes de l'empreinte, sur un voisinage local centré au point (x, y), forment avec l'axe horizontal.
L'approche la plus naturelle pour l'extraction de cette orientation repose sur le calcul de l'angle du gradient. Le gradient exprime la direction de la variation maximale des niveaux de gris. Soient ∇x et ∇y, respectivement, les composantes horizontale et verticale du gradient au point (x, y). L'orientation peut alors être estimée à partir de la valeur $\frac{\pi}{2} + \arctan(\frac{\nabla y}{\nabla x})$. Sur un bloc de l'image, Rao [Rao90] propose de calculer l'orientation à partir de la moyenne locale des gradients. Pour un calcul plus robuste aux erreurs, un ajustement par les moindres carrés est utilisé. Dans l'algorithme 4, nous illustrons les détails de cette méthode. Un lissage, par filtre moyenneur par exemple, est ensuite appliqué pour enlever les artefacts de calcul. Cependant, à cause de leur caractère circulaire, le calcul de la moyenne des orientations pose problème (i.e. Quelle est la moyenne entre 0 et 90° ou entre 5 et 175° ?). Une solution simple et efficace est de doubler les orientations. Chaque élément est codé par la valeur $d_{ij} = (\cos 2\theta_{ij}, \sin 2\theta_{ij})$. La moyenne des orientations d'une fenêtre $n \times n$ est effectuée en faisant la moyenne séparément des deux composantes x et y : $\bar{d} = (\frac{1}{n^2}\sum_{i,j}\cos 2\theta_{ij}, \frac{1}{n^2}\sum_{i,j}\sin 2\theta_{ij})$. Dans l'algorithme 4, l'adoucissement de l'image d'orientation est opéré par un filtre gaussien 5×5.

En utilisant la composante sinus de l'image d'orientation, l'estimation du point core est détaillée dans l'algorithme 5.

Algorithme 4 : Estimation du champ d'orientation de l'image d'empreinte

Input : I Image d'empreinte
Output : O Champ d'orientation dans l'intervalle $[0, \pi[$

1 Diviser I en blocs de taille $W \times W$
2 **for** *tout bloc* B **do**
3 Utiliser les masques de Sobel 3×3 pour estimer le gradient $\begin{pmatrix} G_x \\ G_y \end{pmatrix}$ de chaque pixel (i, j) du bloc B
4 Estimer l'orientation dominante $O(B)$ à partir des formules des moindres carrées suivantes :

$$V_x(i,j) = \sum_{u=i-\frac{W}{2}}^{i+\frac{W}{2}} \sum_{v=j-\frac{W}{2}}^{j+\frac{W}{2}} G_x(u,v) G_y(u,v)$$

$$V_y(i,j) = \sum_{u=i-\frac{W}{2}}^{i+\frac{W}{2}} \sum_{v=j-\frac{W}{2}}^{j+\frac{W}{2}} (G_x(u,v)^2 - G_y(u,v)^2)$$

$$O(B) = \frac{1}{2} \tan^{-1}\left(\frac{V_x(i,j)}{V_y(i,j)}\right)$$

5 Adoucissement du champ d'orientation O en utilisant un filtre gaussien $G(x, y)$ de taille 5×5 où chaque bloc est remplacé en moyennant son voisinage comme suit :

$$O(B) = \frac{1}{2} \tan^{-1}\left(\frac{G(x,y) \sin 2O(x,y)}{G(x,y \cos 2O(x,y)}\right)$$

Algorithme 5 : Estimation du point core

Input : O Image d'orientation de taille $M \times N$
A matrice de taille $M \times N$ initialisée à 0
Output : $(core_x, core_y)$ coordonnées cartésiennes du point core

1 Calcul de ξ la composante sinus de l'image d'orientation O : $\xi \leftarrow \sin(O)$
2 **for** *chaque pixel* (i, j) *dans* ξ **do**
3 Intégrer les valeurs d'intensités dans les régions R_I et R_{II}, définies dans le figure 3.15, par la formule : $A(i, j) \leftarrow \sum_{R_I} \xi - \sum_{R_{II}} \xi$
4 $(core_x, core_y) \leftarrow$ le pixel qui a la valeur maximale dans A

Commentaire : Les régions R_I et R_{II} ont été déterminées empiriquement dans l'étude faite par Jain *et al.* [JPHP00]. Dans nos tests, nous avons pris *surface*(R_I)=2 × *surface*(R_{II}). Cette géométrie des régions R_I et R_{II} permet de détecter le point core dans les crêtes concaves.

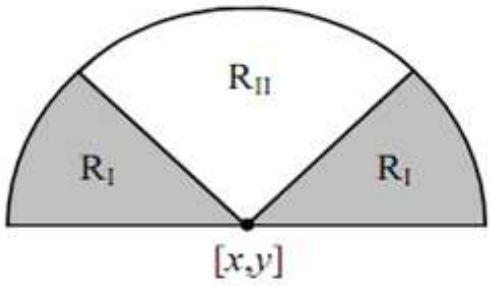
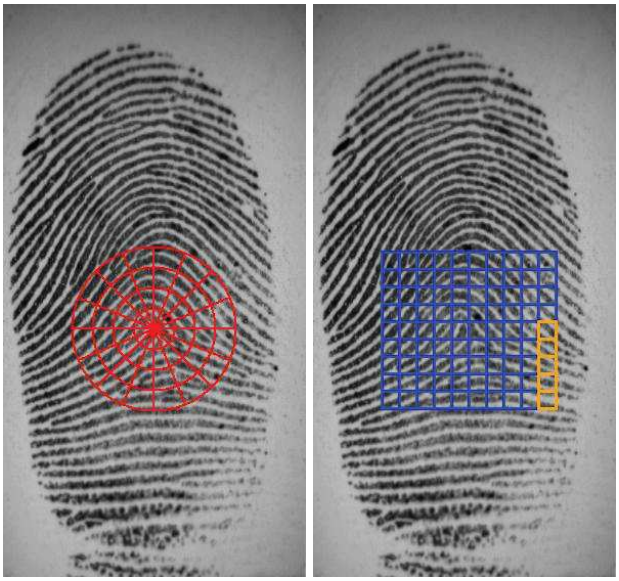


FIGURE 3.15 – Régions d'intégration R_I et R_{II}

Autour du point core, nous définissons deux différentes sectorisations : circulaire ou carrée comme indiqué sur la figure 3.16.



(a) (b)

FIGURE 3.16 – Configurations de la région d'intérêt. (a) circulaire, (b). carrée

La mosaïque circulaire est constituée de B bandes concentriques de même largeur b. Chaque bande est divisée en 16 secteurs d'une largeur de $22,5°$ chacun. Pour la configuration carrée, nous proposons de diviser l'image en $nbW \times nbL$ blocs de taille égale W, centrés autour du point core.

3. **Validation de la région d'intérêt**
   Une phase de validation de la région d'intérêt, suivant le protocole défini dans l'al-

gorithme 6, est ensuite opérée. Si la région d'intérêt n'est pas validée, l'algorithme retourne une erreur de type Fail To Enroll (FTE).

Algorithme 6 : Validation de la région d'intérêt

Input : *ROI* région d'intérêt
Output : valide : booléen

Vérifier si chaque Secteur ou Bloc S représente une alternance stricte entre crêtes et vallées. L'alternance est déduite de l'énergie E du spectre de Fourier de S :
if $sum(E) > T_o$ **then**
    Alternance vérifiée (T_o est un seuil global calculé en utilisant la méthode d'Otsu [Ots79])
else
    pas d'alternance
if *ROI est dans les limites de l'image et alternance vérifiée pour tous les secteurs ou blocs* **then**
    $Valide \leftarrow true$
else
    $Valide \leftarrow false$

4. **Extraction du vecteur caractéristique**

   L'extraction du descripteur de texture est inspirée du *FingerCode* proposé par Jain *et al.* dans [JPHP00]. Un banc de filtres de Gabor est appliqué sur la région d'intérêt binarisée. Pour la binarisation, nous utilisons le code matlab de *P. Kovesi* disponible sur l'URL : [http ://www.csse.uwa.edu.au/ pk]. Pour diminuer la redondance, nous paramétrons le couple *(Nombre de fréquences, Nombre d'orientations)* à $(1, 8)$. En effet, il suffit de mettre la fréquence du filtre de Gabor à la valeur moyenne de la distance inter-crêtes dans l'image d'empreintes.

   Le vecteur caractéristique $X = (x_1, ..., x_n)$ est basé sur la statistique de l'écart-type absolu moyen. Elle est calculée pour chacun des secteurs/blocs de la région d'intérêt circulaire/carrée, respectivement, comme suit :

   – Soit ROI_θ, la ROI filtrée avec la direction θ du banc de Gabor. $X = (x_1, ..., x_n)$ est alors calculé à partir de ROI_θ de telle sorte que :

$$\forall x_{i\theta} \in X, x_{i\theta} = \frac{1}{n_i} \sum_{n_i} |ROI_{i\theta}(x, y) - \rho_{i\theta}|$$

   Avec, n_i le nombre de pixels dans le secteur/bloc numéro i et $\rho_{i\theta}$ leur valeur moyenne après filtrage dans la direction θ.

$i = 1..B \times 16$:nombre de secteurs dans la région circulaire,
$i = 1..nbW \times nbL$: nombre de blocs de la région carrée et
$n = B \times 16 \times 8$ ou $n = nbW \times nbL \times 8$: taille du vecteur caractéristique de la région circulaire ou carrée.

5. **Gestion des distorsions géométriques**
   Alors que le problème de translation est implicitement pris en charge par l'utilisation d'un point de référence, le problème de rotation demeure. Nous le gérons différemment suivant la configuration de la région d'intérêt en entrée.
   – **Région circulaire**
     a) A partir du vecteur X, générer 5 décalages cycliques pour gérer les rotations à 22.5°.
     b) Extraire le vecteur X à partir d'une rotation de 11.5° de l'image initiale et générer ensuite 5 décalages cycliques.
     c) Le score de correspondance entre deux empreintes digitales est égal à la distance euclidienne minimale entre les 10 codes de références et la principale instance du code en entrée.

   – **Région carrée**
     Nous gérons le problème de rotation par l'utilisation d'une base d'apprentissage sélectionnée aléatoirement à partir du benchmark de test comme le montre la figure 3.17.

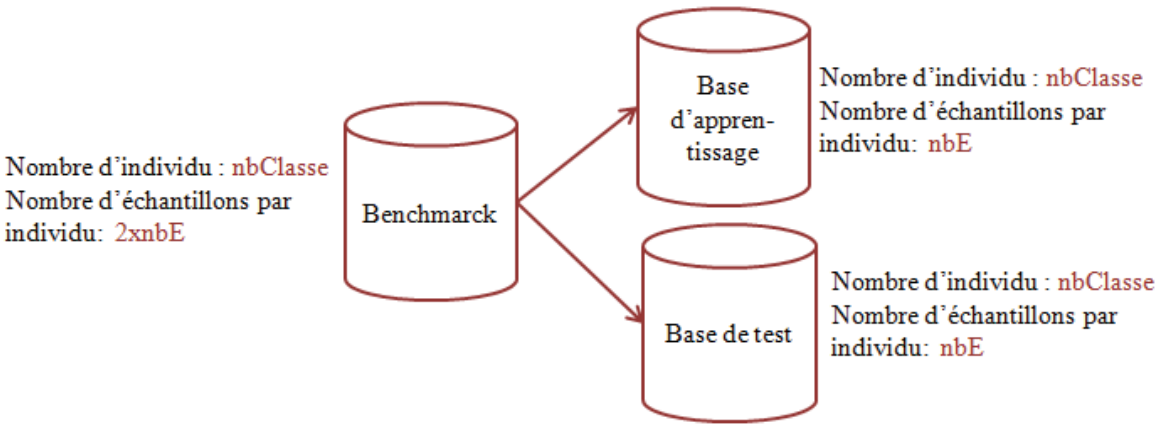


FIGURE 3.17 – Sélection des bases d'apprentissage et de test

En utilisant cet ensemble , nous représentons la distribution de chaque individu par son *vecteur moyenne* $C = (C_1, C_2, ..., C_n)$. La comparaison se fait par le calcul

de la distance euclidienne entre le vecteur moyenne et celui en entrée.

A partir de la base d'apprentissage, nous pouvons encore mieux représenter les différents individus, en utilisant des éléments de la théorie bayesienne, plus précisément le principe de vraisemblance. Un individu est réduit à une classe représentée par un modèle constitué d'une moyenne et d'une covariance. Nous n'allons pas nous encombrer ici avec tous les aspects théoriques. L'annexe A en fait un bref rappel. Pour de plus amples détails, le lecteur intéressé peut se référer au livre de Duda *et al.* [DHS00]. Nous décrivons donc le côté pratique de notre approche en la synthétisant par l'algorithme 7. L'idée est d'utiliser le facteur de vraisemblance pour calculer la probabilité qu'un individu appartienne ou non à la classe considérée lors de la vérification.

6. **Expérimentations de la vérification par descripteurs globaux**
   Comme pour l'étude comparative, nous accomplissons nos tests sur la $FVC2002 - DB2$ déja présentée. Les premiers tests concernent la précision de l'algorithme de détection du point core.
   – **Test de précision du point core**
   Dans plusieurs travaux qui reposent sur le point core comme [LN07, TNG04], les auteurs éliminent manuellement les images suspectes, soit parce qu'elles ne contiennent pas de point core (comme les images de type arc) ou lorsque la détection présente un déplacement exagéré. Grâce à notre méthode de validation de la région d'intérêt, l'algorithme 6 peut être complètement autonome en éliminant, de lui-même, les fausses alarmes. Ci-dessous les résultats obtenus :

   - Le point core n'existe pas sur l'image mais le module de validation le signale : **22** images - **2.75%** du total
   - La détection du point core est fausse mais le module de validation le signale : **50** images - **6.25%** du total
   - La détection du point core est fausse sans que le module de validation ne s'en aperçoive : 5 images.

   Ainsi, le taux d'échec à l'enrôlement est égal à : $FTE = 2.75 + 6.25 = 9\%$.

Algorithme 7 : Vérification par facteur de vraisemblance

Input : nb : nombre total d'échantillons dans la base d'apprentissage ou la base de test
$nbClasse$: nombre d'utilisateur dans la base d'apprentissage ou la base de test
nbE : nombre d'échantillons par individu dans la base d'apprentissage ou la base de test
n : taille du vecteur caractéristique
A : Matrice de taille $nb \times n$ comprenant tous les vecteurs caractéristiques de la base d'apprentissage
T : Matrice de taille $nb \times n$ comprenant tous les vecteurs caractéristiques de la base de test
Output : score de décision

1 Accomplir une ACP sur les deux matrices A et T pour réduire la taille des vecteurs caractéristiques de n à k avec $(K < n)$
2 Calculer pour chaque individu j son vecteur caractéristique moyenne C_j à partir de la matrice A
3 $WCovariance \leftarrow$

$$\frac{1}{nbClasse} \sum_{i=1}^{nbClasse} Covariance(A(nbE(i-1)+1 : nbE \times i, 1 : k))$$

Il s'agit de la matrice covariance intra-classe
4 $TCovariance \leftarrow covariance(A)$ Il s'agit de la matrice covariance totale
5 Sauvegarder les deux matrices $WCovariance$ et $TCovariance$ comme références pour tous les individus
6 Sauvegarder pour chaque individu j son vecteur moyenne C_j
7 Soit Y le vecteur caractéristique en entrée qui représente un des vecteurs ligne de la matrice T. Le but est de calculer la correpondance entre Y et C_j
8 Nous utilisons pour cela le log-likelihood ratio (LLR) comme mesure de similarité, Ainsi :
9 $diff \leftarrow Y - C_j$
10 $WCov\grave{a}riance \leftarrow inverse(WCovariance)$ Calcul de la matrice inverse
11 $TCov\grave{a}riance \leftarrow inverse(TCovariance)$ Calcul de la matrice inverse
12 $PosteriorProbaDensity \leftarrow diff \times WCov\grave{a}riance \times diff$
13 $Tlikelihood \leftarrow Y \times TCov\grave{a}riance \times Y$
14 $LLR \leftarrow 0 - \frac{PosteriorProbaDensity}{2} + Tlikelihood$
15 **return** LLR

– **Test de précision de la vérification biométrique**
Les paramètres utilisés sont les suivants :
- **Les filtres de Gabor,** taille du masque : 33×33, fréquence : 0.1, $\delta_x = \delta_y = 4$ et 8 directions.

- **Région d'intérêt circulaire,** nombre de bandes : $B = 5$, nombre de secteurs : $16 \times 5 = 80$, largeur de chaque bande : $b = 20$ pixels, taille du vecteur caractéristique : $n = 80 \times 8 = 640$ entiers normalisés dans l'intervalle $[0, 255]$.

- **Région d'intérêt carrée,** taille du bloc : 17×17, nombre de blocs : $nbW \times nbL = 10 \times 9 = 90$, taille du vecteur caractéristique : $90 \times 8 = 720$ entiers normalisés dans l'intervalle $[0, 255]$. Lorsque l'ACP est opérée, les vecteurs sont de taille $k = 400$ éléments.

La normalisation du vecteur caractéristique dans l'intervalle $[0, 255]$ nous permet d'afficher le vecteur en niveaux de gris comme le montre la figure 3.18.

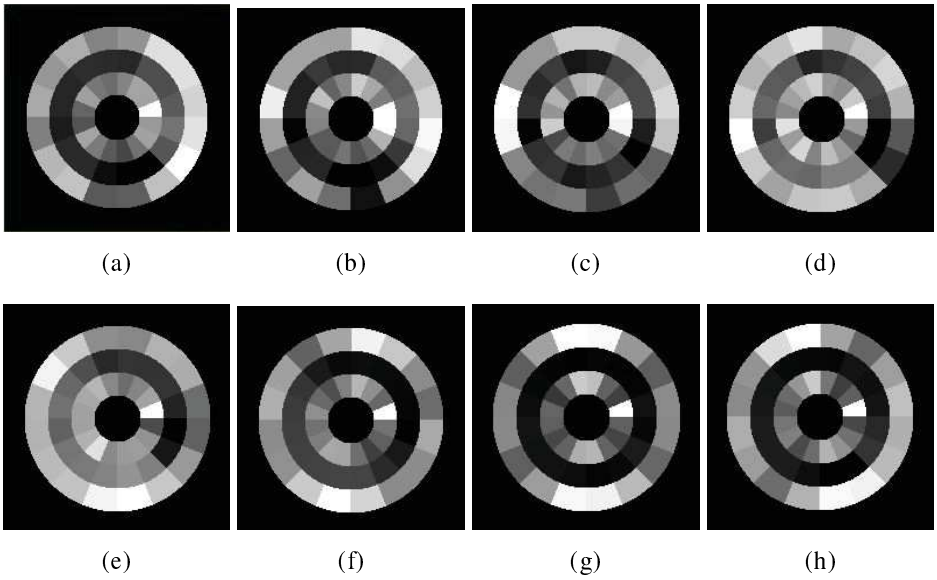


FIGURE 3.18 – Vecteur caractéristique de la région circulaire pour un banc de Gabor à 8 directions

Nous suivons le même protocole de test des performances de vérification biométrique que la FVC (Fingerprint Verification Competition), à savoir :

– Pour calculer le FRR, chaque échantillon est comparé avec les échantillons restants du même utilisateur. Ainsi pour 100 utilisateurs et 8 images par utilisateur, nous aurons $((8 \times 7)/2) \times 100 = 2800$ tests (Nous divisons par 2 pour éliminer la redondance). Cependant, à cause du FTE, nous n'aurons que 352 tests.
– Pour calculer le FAR, chaque premier échantillon est comparé avec les premiers échantillons des autres utilisateurs. Nous aurons $((99 \times 100)/2) \times 100 = 4950$.

Cependant, à cause du FTE, nous n'aurons que 3828 tests.

La courbe DET des différents tests est représentée par la figure 3.19. Les résultats en terme d'EER sont résumés dans le tableau 3.3.

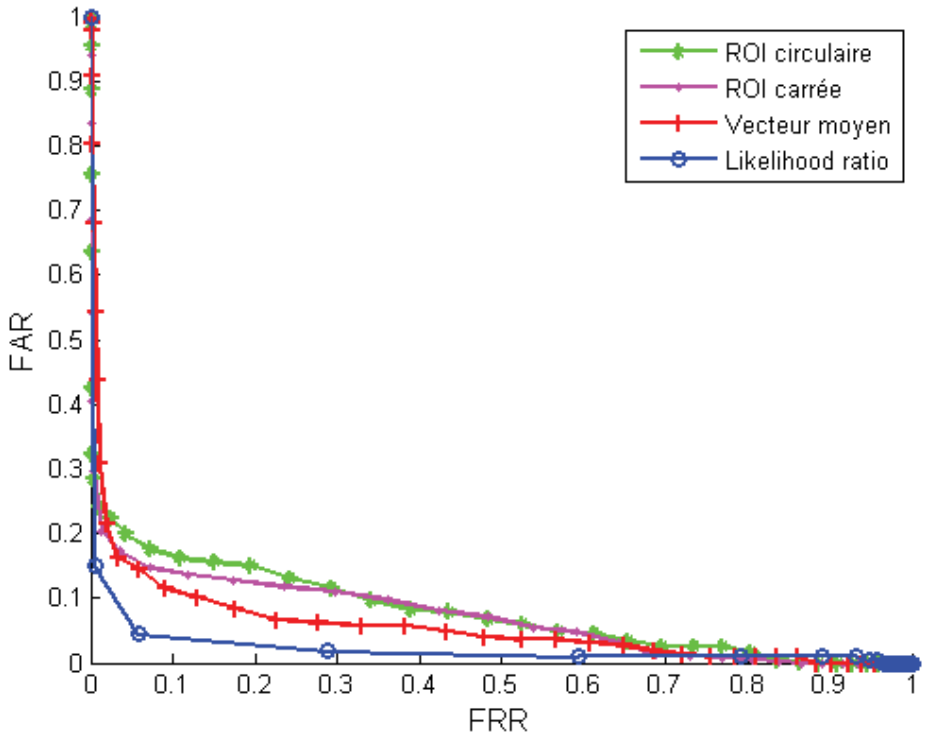


FIGURE 3.19 – Courbe DET de la vérification en utilisant différents descripteurs globaux

Méthode	EER
ROI circulaire	14.17%
ROI carrée	12.77%
Vecteur moyenne (ROI carrée)	10.25%
Facteur de vraisemblance (ROI carrée)	5.14%

TABLE 3.3: Performances de la vérification d'empreintes par descripteurs globaux

Nous remarquons que la région carrée apporte de meilleurs résultats que la région circulaire. L'utilisation d'une base d'apprentissage améliore encore les résultats. La vérification par l'utilisation du facteur de vraisemblance donne des résultats optimaux avec une erreur moyenne de 5.14%.

3.2.3 Protection du descripteur biométrique

A partir du vecteur caractéristique obtenu, nous souhaitons garantir les critères de révocabilité et de protection de vie privée, explicités dans le chapitre 2. Dans cette thèse, nous nous intéressons aux transformations révocables qui sont plus axées sur les exigences de révocabilité. En considérant le descripteur de la région carrée, le modèle biométrique consiste en un vecteur moyenne $C = (c_1, ..., c_n)$. Pour la vérification par facteur de vraisemblance, il s'agit du vecteur moyenne C, de la matrice de covariance intra-classe $\sum_W$ et de la matrice de covariance totale $\sum_T$.
Dès le début de notre recherche, en 2010 donc, nous nous sommes intéressés à la transformation par *BioHashing* [TNG04], bien adaptée aux représentations vectorielles. Nous allons donc considérer un module de protection par BioHashing pour le vecteur $C = (c_1, ..., c_n)$. Pour la vérification par facteur de vraisemblance, il n'est pas évident d'envisager le BioHashing, nous pensons à d'autres techniques comme la *cryptographie homomorphique* pour le calcul du facteur de vraisemblance dans un espace chiffré. Pour lors, cela se dessine plutôt comme une perspective. Nous nous concentrons, maintenant, sur la transformation par BioHashing.

1. **Principe du BioHashing**
   Le principe du BioHashing est de générer un code unique, appelé *biocode*, à partir de deux données : le modèle biométrique et un nombre aléatoire qui, pour plus de protection, doit être stocké sous forme de jeton ou de clé USB. Le même schéma de transformation est appliqué à la fois :
   – Pendant l'étape d'enrôlement, où le biocode est stocké à la place du modèle biométrique.
   – Pendant l'étape de vérification, où un nouveau biocode est généré à partir du nombre aléatoire attribué à l'utilisateur pendant l'enrôlement.

   La vérification repose ensuite sur le calcul de la distance de Hamming entre le biocode de référence et celui nouvellement émis. Ce principe garantit la révocabilité et la diversité du biocode en utilisant des nombres aléatoires différents pour différentes applications. Le procédé du BioHashing est illustré par la figure 3.20.
   Nous pouvons constater qu'il s'agit d'un système à deux facteurs d'authentification, dans le sens où la fonction de transformation combine un nombre aléatoire, sous forme de grain, stocké dans un jeton, avec le modèle biométrique exprimé en tant que vecteur de longueur fixe $X = (x_1, ..., x_n)$, $X \in \mathbb{R}^n$.

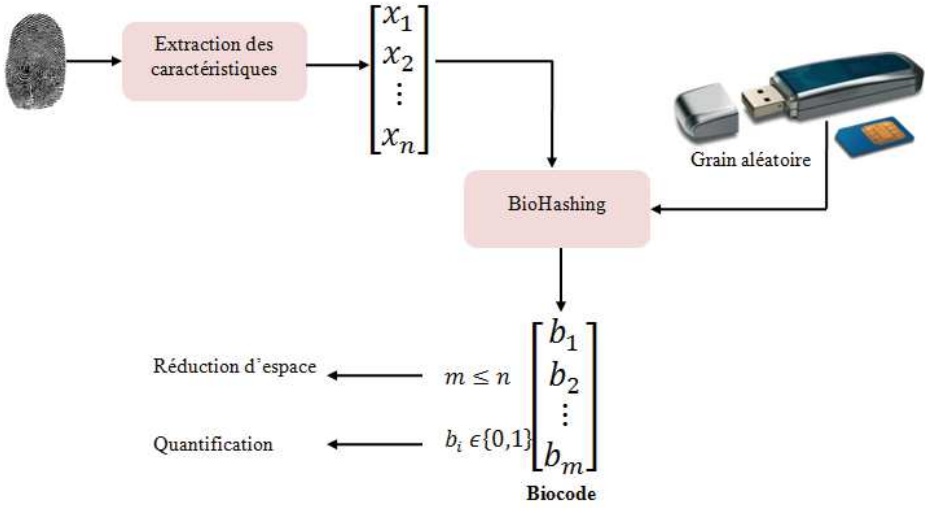


FIGURE 3.20 – Transformation par BioHashing

Le principe du BioHashing, détaillé dans [TNG04], consiste en une projection aléatoire suivie d'une étape de quantification. Nous donnons maintenant plus de détails sur ces étapes.

– **Projection aléatoire**

Une projection aléatoire consiste à multiplier un vecteur de données $X \in \mathbb{R}^n$ par une matrice aléatoire R pour générer un vecteur $W \in \mathbb{R}^m$ de dimension réduite $m < n$, à partir du produit : $W = RX$. En biométrie, l'utilité d'une telle projection dépend du fait que les distances entre les différents vecteurs caractéristiques d'un même utilisateur seront préservées ou non. On note $R = (R_{ij})_{i,j\in[1,n]\times[1,m]}$, la matrice de projection. Il a été montré par S. Kaski [Kas98] que si la matrice R est orthonormée alors la similarité entre les vecteurs est préservée (R devient une base).

Nous pouvons valider cela comme suit :
Soient deux vecteurs unitaires V_1 et V_2. Il est bien connu, que le cosinus de l'angle entre 2 vecteurs est une mesure de leur similarité. Lorsqu'il s'agit de vecteurs unitaires, le cosinus n'est autre que leur produit scalaire $V_1^t V_2$. Soient $\grave{V}_1 = RV_1$ et $\grave{V}_2 = RV_2$. Il s'agit de vérifier si $\grave{V}_1^t \grave{V}_2 = V_1^t V_2$.

$$\grave{V}_1^t \grave{V}_2 = V_1^t R^t R V_2$$

Si R est orthonormée alors l'inverse de R est sa transposée : $R^t R = I$ et ainsi la distance est préservée. Néanmoins, en pratique, il n'est jamais évident de réaliser une orthonormalisation parfaite à partir de vecteurs générés aléatoirement.

Dans ce cas $R^tR = I + \delta$.
Le lemme de Johnson-Lindenstrauss étudié dans [DG99] assure que sous certaines conditions sur le nombre de colonnes m de la matrice R, δ peut converger vers ϵ, $0 < \epsilon < 1$.

Lemme 1 . Soit $\epsilon \in]0, 1[$ et P un ensemble de k points dans $\mathbb{R}^n$. Soit m un entier positif vérifiant $m \geq \frac{4 \log k}{\epsilon^2/2 - \epsilon^3/3}$. Alors, il y a une application $f : \mathbb{R}^n \to \mathbb{R}^m$ telle que :
$\forall x, y \in P, (1 - \epsilon)\|x - y\|^2 \leq \|f(x) - f(y)\|^2 \leq (1 + \epsilon)\|x - y\|^2$

Ce lemme indique qu'il est possible d'aplatir tout espace de k points dans un espace de dimension réduite m où les distances seront bien conservées à condition qu'une borne inférieure soit atteinte. Comme conclusion de ces différentes remarques, nous pouvons déduire que les distances de la projection aléatoire sont préservées sous les hypothèses que : la matrice R soit orthonormée et que le nombre de ses colonnes m soit suffisemment proche d'une certaine limite inférieure. Le résultat de la projection est noté $W = (w_1, ..., w_m)$, avec $W = RX \in \mathbb{R}^m$. Cette projection consiste, en quelque sorte, à masquer les données biométriques dans un certain espace.
En partique,

- La matrice R est générée à partir du nombre aléatoire stocké sur le jeton, appelé **clé**, et considéré alors comme le grain d'un générateur pseudo-aléatoire.
- Les tests (i.e. que nous avons accomplis) montrent qu'il suffit d'utiliser un générateur de loi uniforme.
- Une orthonormalisation est nécessaire pour transformer la matrice aléatoire R en une base de projection. Pour ce faire, nous utilisons le processus de Gram-Schmidt [Hof89]. Il est aussi nécessaire de vérifier que les vecteurs colonnes de R soient linéairement indépendants avant d'appliquer le processus de Gram-Schmidt.

– **Quantification**
Cette étape est consacrée à la transformation vers un vecteur à valeurs binaires, du vecteur à valeurs réelles W, issu de l'étape de projection. Cette binarisation, de type one-to-many est ajoutée pour renforcer la non inversibilité de la transformation (qui s'appuie déjà sur le processus de projection). Elle nécessite la définition d'un seuil τ_b pour calculer le biocode final $B = (b_1, ..., b_m)$, à partir de la formule

suivante :

$$b_i = \begin{cases} 0 & Si \ w_i \leq \tau_b \\ 1 & Si \ w_i > \tau_b \end{cases} \tag{3.5}$$

En pratique, le seuil τ_b est choisi égal à 0 car les résultats de la projection ont la même probabilité d'être négatifs ou positifs. Ainsi, chaque bit b_i du biocode B aura la même probabilité d'apparition ce qui a comme effet d'augmenter le contenu de l'information réellement présente dans B et ainsi sa robustesse.

2. **Application du BioHashing et tests**

A présent, nous explicitons dans l'algorithme 8 le processus de protection du vecteur moyenne $C = (c_1, ..., c_n)$ par BioHashing.

Algorithme 8 : Processus de protection par BioHashing

Input : C : Le vecteur moyenne de l'utilisateur U de taille n
S : La clé sous forme d'un nombre aléatoire attribuée à l'utilisateur U
Output : B : le biocode de taille m avec $m < n$

1 Générer à partir de S une matrice aléatoire uniforme $R_{n\times m}$ (n lignes et m colonnes)
2 Contrôler que les vecteurs de R sont linéairement indépendants sinon aller à 1
3 Appliquer le processus de Gram-Schmidt pour transformer R en une matrice orthonormée
4 Projeter le vecteur C sur la nouvelle matrice R :

$$[C_1, C_2, ..., C_n] \begin{pmatrix} r_{11} & \dots & r_{1m} \\ \vdots & \ddots & \vdots \\ r_{n1} & \dots & r_{nm} \end{pmatrix} = [w_1, w_2, ..., w_m]$$

5 Binarisation de W, à partir du seuil $\tau_b = 0$, pour obtenir le biocode $B = \{b_1, b_2, ..., b_m\}$ tel que :

$$b_i = \begin{cases} 0 & Si w_i \leq \tau_b \\ 1 & Si w_i > \tau_b \end{cases}$$

6 Effacer le vecteur C et sauvegarder le vecteur B comme référence de l'utilisateur U

Nous testons maintenant ce procédé, en utilisant la $FVC2002 - DB2$ comme base de test, et en le modulant par rapport à différentes configurations. Dans chaque configuration, nous considérons le taux d'erreur égal (EER) pour les deux scénarios : *Best case*, lorsqu'aucune attaque n'est menée et *Worst case* lorsque la clé utilisateur

est connue, à chaque fois, de l'imposteur (stolen token attack).
Soient :

– *Configuration1* : le procédé classique où l'algorithme 8 est opéré.
– *Configuration2* : le même procédé est utilisé mais en normalisant le vecteur C par son module.
– *Configuration3* : Le vecteur C est normalisé dans l'intervalle $[-1, 1]$, la même plage que le résultat de Gram-Schmidt.

Le tableau 3.4 illustre les résultats obtenus lorsque la taille du biocode est à la valeur maximale possible $m = n = 720$. La longueur du biocode a été choisie en faisant varier le paramètre m. Sur la figure 3.21, nous remarquons que l'EER diminue en augmentant la longueur m.

	Configuration1	Configuration2	Configuration3
Best case	0%	0%	0%
Worst case	16.2%	16.2 %	14.45%

TABLE 3.4: Résultats du BioHashing en terme d'EER sur différentes configurations et pour $m = 720$

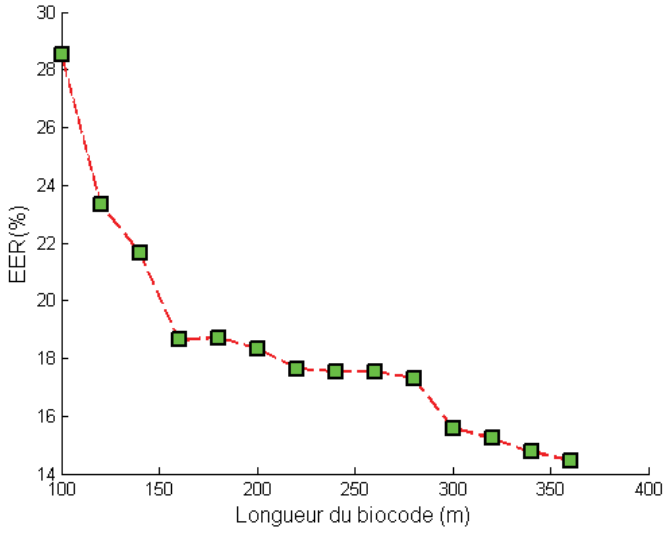


FIGURE 3.21 – EER obtenu en variant m dans la cas du vol de la clé

En outre, la binarisation utilisée dans le BioHashing repose sur un processus de seuillage. La valeur du seuil τ_b est fixée à zéro, en raison de la probabilité théorique que chaque élément, après projection aléatoire, a les mêmes chances d'être positif que négatif. Ceci a comme but d'augmenter la difficulté pour un attaquant qui veut estimer un biocode accepté.

Ainsi, au lieu d'utiliser la valeur 0 pour τ_b , nous proposons de fixer le seuil à la valeur de la médiane. Les seuils deviennent personnels pour chaque individu. Cela définit la *configuration4*. Pour éviter l'utilisation d'une base d'apprentissage dans l'estimation de la médiane, nous nous basons sur l'observation que cette médiane dépend fortement de la matrice de projection. Ainsi, il suffit de l'estimer à partir d'un ensemble fixe d'images d'empreintes digitales indépendantes.

Par exemple, dans la base de données FVC2002, il existe toujours deux ensembles : un ensemble A pour les tests (contenant 800 images) et un ensemble B pour la fixation des paramètres, contenant 80 images, que nous allons utiliser. Le protocole d'estimation de la médiane pour chaque utilisateur U est explicité par l'algorithme 9.

Algorithme 9 : Estimation de la médiane

Input : S : La clé attribuée à l'utilisateur U
set B : L'ensemble B des images indépendantes
Output : $M = (m_1, m_2, ..., m_m)$: le vecteur médiane attribué à l'utilisateur U

foreach *image de l'ensemble set B* **do**
    Calculer $X = (x_1, x_2, ..., x_n)$ le vecteur caractéristique (ROI carrée sans utiliser le vecteur moyenne)
    Normaliser les valeurs de X dans l'intervalle $[-1, 1]$ comme dans la *configuration3*
    Calculer $W = (w_1, w_2, ..., w_n)$ avec $w_i \leftarrow XR_i$ (R_i, la colonne i de la matrice R)
$g \leftarrow$ le nombre de vecteurs W générés
for $i \leftarrow 1$ **to** m **do**
    Calculer m_i la médiane des éléments (Wij), $j = 1..g$
Le vecteur $M = (m_1, m_2, ..., m_m)$ est considéré comme le vecteur des seuils de binarisation pour l'utilisateur U

Dans la tableau 3.5, nous comparons les EERs entre le système biométrique de base (sans module de protection, cf. le tableau 3.3) et le système avec protection pour les deux configurations : *configuration3* et *configuration4*.

Sans protection	Configuration3		Configuration4	
	Best	Worst	Best	Worst
10.25%	0%	14.45 %	0 %	11.40%

TABLE 3.5: Comparaison du système de vérification sans et avec protection en terme d'EER

Nous remarquons que la *configuration4* est meilleure que les autres configurations avec un EER de 11.40% lorsque la clé est connue de l'imposteur. Dans le cas *Best*, quand la clé n'est pas révélée ou volée, le BioHashing réussie à faire une séparation nette entre les clients et les imposteurs avec un EER à 0%. La courbe DET de la figure 3.22 montre une comparaison du BioHashing entre les cas *Best* et *Worst*. Toutefois, si nous accordons une attention particulière à la distribution des scores

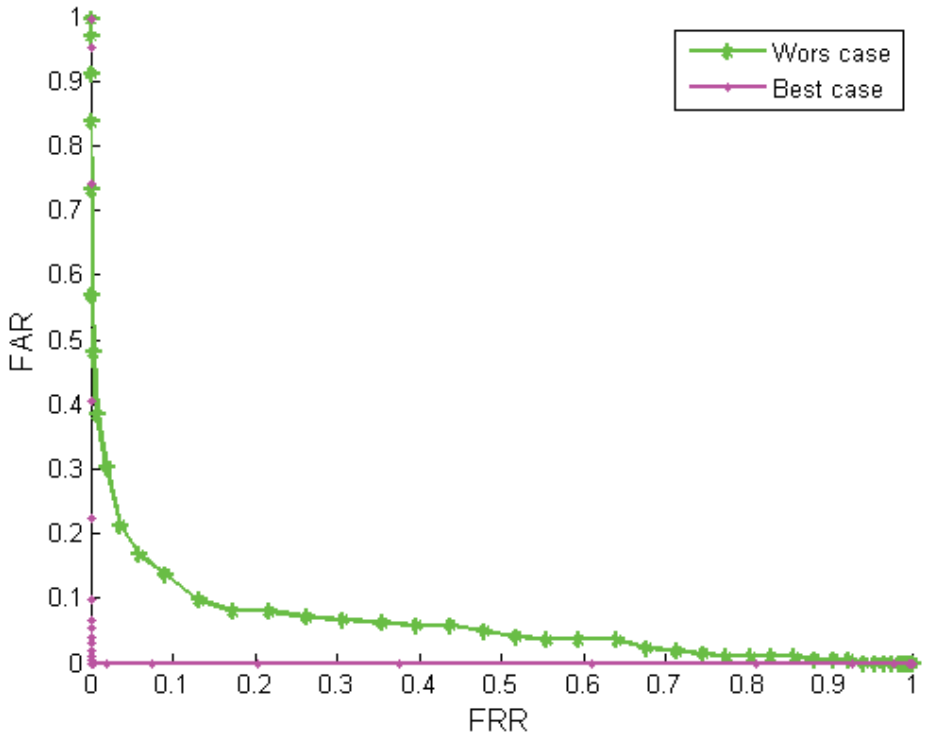


FIGURE 3.22 – Courbes DET du BioHashing pour les cas Best et Worst

représentée par la figure 3.23, nous constatons que cette affirmation n'est pas réaliste en pratique. Ces taux d'erreurs sont possibles, mais en utilisant différents seuils de décision. Dans le cas *Best*, le 0% est obtenu en utilisant un seuil de décision à 17%. Dans le cas *Worst*, EER=11.40% lorsque le seuil de décision est à 6%. Il est clair que le système révocable sera réglé en considérant le seuil lorsque le pire scénario est envisagé (6% dans ce cas).

Une fois que notre module de BioHashing est bien paramétré, nous procédons dans la section suivante à l'évaluation du système révocable obtenu. Pour ce faire, nous utilisons notre cadre d'évaluation présenté dans le chapitre précédent.

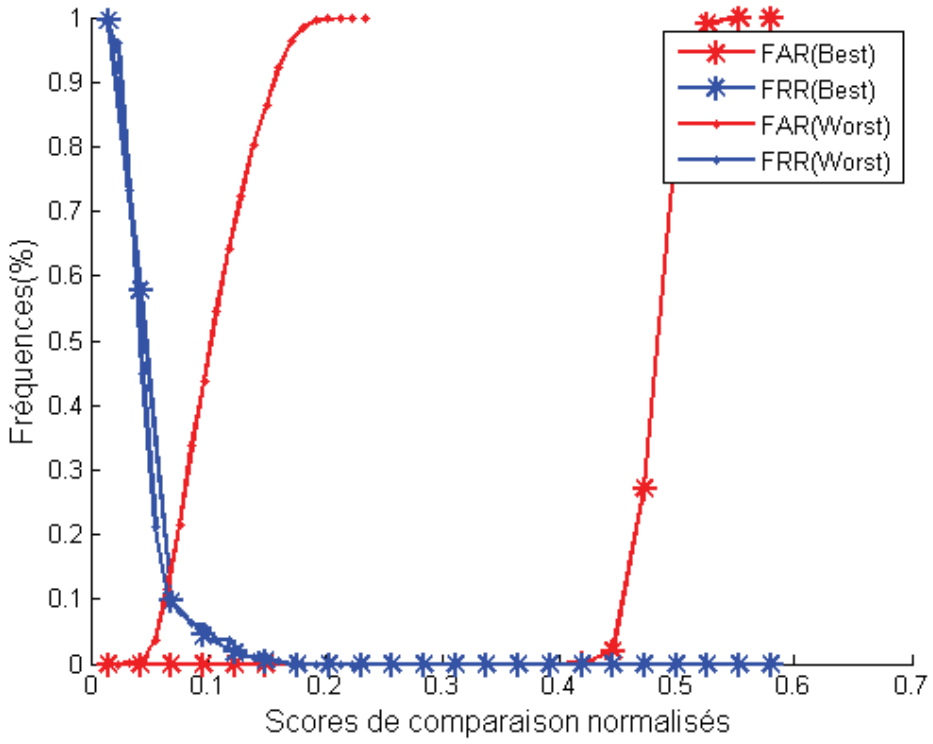


FIGURE 3.23 – FAR vs. FRR pour les cas Best et Worst

3.2.4 Evaluation du système révocable par descripteurs globaux

Nous commençons par paramétrer la valeur du seuil de décision du système révocable à la valeur normalisée $\varepsilon_D(\%) = 6\%$. Le tableau 3.6 résume les valeurs des différentes métriques $A_1, ..., A_{17}$.

A1	A2	A3	A4	A5	A6	A7	A8	A9
11.40%	1	0%	0%	13.11%	0%	true	5760 bits	NI
A10	A11	A12	A13	A14	A15	A16	A17	
IPP	OF	ONF	0%	0%	0%	0	720 bits	

TABLE 3.6: Evaluation du système révocable par descripteurs globaux d'empreintes digitales

La figure 3.24 illustre l'évolution du FAR en fonction des différents seuils de décision pour les métriques : A_3 (attaque à zéro effort), A_4 (attaque par force brute), A_5 (attaque par vol de clé), A_6 (attaque par vol de biométrie), A_{14} (attaque par écoute de 3 biocodes différents d'un même utilisateur), A_{15} (attaque par écoute de 11 biocodes différents d'un même utilisateur).

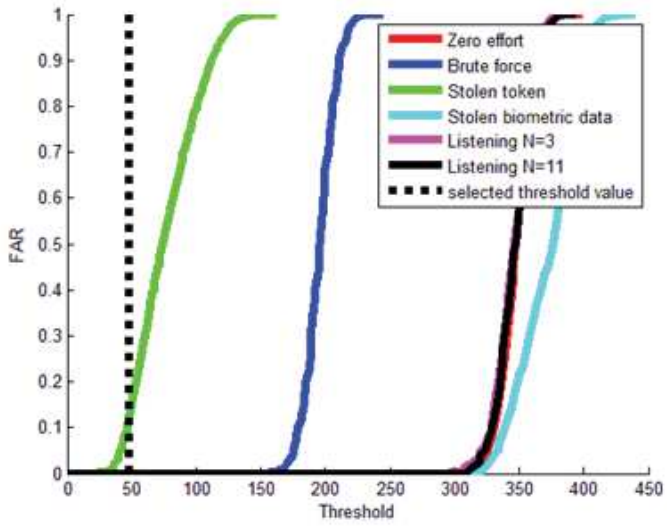


FIGURE 3.24 – Evolution du FAR pour différents scénarios d'attaque

Nous remarquons d'après cette figure que toutes les attaques peuvent être effectives. Cependant, en réglant le seuil de décision du système révocable ε_D à la valeur du cas *Worst* (cas du vol de clé), toutes les valeurs du FAR s'annulent mise à part celle de l'attaque par vol de clé qui reste effective à 13.11%.

Analyse de diversité : Nous constatons aussi que l'un des points forts de la méthode de protection par BioHashing est lié à son aspect de diversité. En effet, la métrique A_{13} du taux de compatibilité croisé est à 0% ce qui a comme effet d'empêcher de lier directement les biocodes des différentes applications entre eux. Les valeurs des métriques théoriques A_{16} (Information mutuelle) et A_{17} (Entropie de diversité) sont aussi satisfaisantes et vont dans le même sens que les attaques pratiques. Le paramètre A_{17} a été calculé à partir de l'histogramme des distances de Hamming normalisées entre les différents biocodes diversifiés de toutes la base de données (nous avons utlisé 10 biocodes diversifiés par individu). Cette distribution pseudo-imposteur est représentée sur la figure 3.25. La valeur moyenne de l'histogramme est de $P = 48.51\%$. Cela indique que la probabilité de prédire une valeur correcte par bit dans le biocode est de $1 - P = 51.49\%$ ce qui répond à une ambiguïté presque totale. La valeur de l'entropie le confirme.

Analyse de la non-inversibilité : Nous nous intéressons maintenant à la possibilité d'inversibilité de la méthode. Nous commençons par supposer que l'attaquant est en possession du biocode $B = (b_1, ..., b_m)$ de l'utilisateur U, de sa clé S ou de sa matrice de projection R_{nm} et éventuellement des paramètres de l'algorithme qui peuvent être

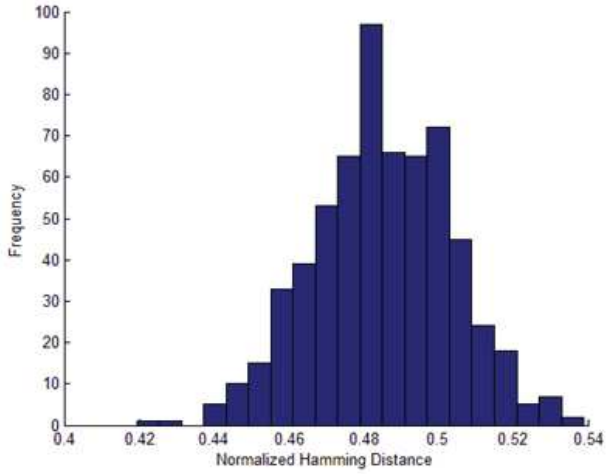


FIGURE 3.25 – Histogramme de la distribution pseudo-imposteur

necessaires. L'objectif est de mesurer la possibilité de déterminer exactement le modèle biométrique d'origine X ou $\tilde{X}$ un modèle approximatif qui pourrait lui correspondre dans le domaine original (sans transformation). L'attaquant pourrait commencer par estimer la valeur du vecteur W issu de la projection de XR de telle sorte que :

$$\begin{cases} Si\ (b_i = 1) & w_i > seuil \\ Si\ (b_i = 0) & w_i \leq seuil \end{cases} \tag{3.6}$$

L'équation 3.6 nous informe qu'à partir du biocode B et en connaissant le vecteur médiane des seuils de binarisation, nous pouvons déduire un vecteur vraisemblable à W, qu'on appellera $\tilde{W}$. Maintenant, il est question d'estimer $\tilde{X}$ tel que : $\tilde{W} = \tilde{X}R$. Il s'agit de résoudre le système d'équations linéaires représenté par l'équation 3.7.

$$\begin{cases} \tilde{x}_1 r_{11} + \tilde{x}_2 r_{21} + ... + \tilde{x}_n r_{n1} = \tilde{w}_1 \\ \qquad\qquad\qquad \dots \\ \tilde{x}_1 r_{1m} + \tilde{x}_2 r_{2m} + ... + \tilde{x}_n r_{nm} = \tilde{w}_m \end{cases} \tag{3.7}$$

Dans ce système, nous avons plus d'inconnues (n) que d'équations (m). C'est un système de type sous-déterminé qui a une infinité de solutions. Ainsi l'estimation de $\tilde{X}$ à partir de l'équation 3.7 n'est pas possible. Cependant, en attribuant des 0 par exemple à certaines inconnues, nous pouvons déduire les autres qui restent et qui peuvent donc générer le vecteur $\tilde{W}$ et ainsi un biocode acceptable. Dans ce cas, un pseudo-inverse peut exister ce qui explique la valeur de la métrique A_{11} à OF (Optimisation Faisable) dans le tableau 3.6 en considérant le modèle de l'adversaire curieux qui n'est pas encore actif. Dans [NJ09], Nagar et Jain ont également proposé une estimation du pseudo-inverse d'une manière plus

complexe mais qui prouve encore une fois cette possibilité.

S'agissant de l'analyse d'inversibilité, nous avons aussi besoin de considérer le scénario des *références multiples*, conséquence directe de l'exigence de révocabilité. Dans ce cas, l'attaquant est en possession de t ($t > 1$) biocodes $\{B_1, ..., B_t\}$ de l'utilisateur U ainsi que de leurs matrices de projection respectives $\{R_1, ..., R_t\}$, issus des révocations possibles. Si la concaténation des matrices $[R_1 R_2 ... R_t]$ aboutit vers une matrice A de rang n alors il y aura autant d'inconnues que d'équations dans le système 3.7. L'estimation de $\tilde{X}$ peut être déduite à partir de : $A^{-1}\tilde{W}$. La valeur de la métrique A_{10} est dans ce cas IPP (Inversion Partielle Possible) alors que $A9$ est à NI (Non Inversible). Cette attaque nous informe aussi que durant l'implémentation, il est conseillé de considérer la taille du biocode m à la plus petite valeur qui converge vers l'EER obtenu lorsque $m = n$. Les tests montrent qu'en fixant m à 630 bits, nous obtenons les mêmes résultats tout en gardant une complexité de code importante, à savoir : 2^{630}.

3.2.5 Analyse et discussion

Le cadre d'évaluation proposé fournit une manière unifiée pour analyser la robustesse du système biométrique révocable en termes de sécurité et de préservation de la vie privée. De cette étude, nous pouvons tirer les conclusions suivantes :

- Le choix du seuil de décision du système révocable est une tâche importante qui peut affecter la robustesse du système par rapport à certaines attaques.
- La valeur de la métrique A_2 à 1 indique que le BioHashing, en supposant qu'aucune attaque n'est opérée, apporte de meilleurs résultats que le système biométrique de base. Néanmoins, lorsque le scénario *Worst* est considéré, la valeur de A_2 est à -0.1 ce qui indique une dégradation des performances par rapport au système de base.
- Les valeurs des métriques $\{A_{13}, A_{14}, A_{15}, A_{16}, A_{17}\}$, en relation avec l'étude de diversité, soulignent cette propriété forte due à la projection aléatoire.
- Le système est vulnérables aux attaques suivantes : *(i)* attaque par vol de clé, *(ii)* inversion partielle à partir des références multiples et *(iii)* estimation du pseudo-inverse. Néanmoins, il est à noter que l'inversion totale n'est pas possible.

L'attaque par vol de clé reste une attaque préjudiciable en terme du taux d'intrusion possible. Nous entreprenons de comparer dans le tableau 3.7 nos résultats avec ceux obtenus dans la littérature et qui représentent l'empreinte, aussi par des descripteurs globaux (non-minutiae fingerprint based representation). Les critères de comparaison sont : l'EER du système sans protection, l'EER du système avec protection en considérant le cas *Worst* et la taille du code obtenu après protection.

	EER sans protection	EER avec protection (Worst case)	Taille du code	Base de données
Teoh *et al.* [TNG04]	5.66%	Non reporté	80 bits	FVC2002-DB2 (partielle)
Lumini and Nanni [LN06]	7.3%	10.9%	Non reporté	FVC2002-DB2(partielle)
Teoh *et al.* [TKKA10]	14.84%	16.21%	80 bits	FVC2002-DB2 (partielle)
Teoh *et al.* [TKKA10]	14.84%	2.39%	250 bits	FVC2002-DB2 (partielle)
Lumini and Nanni [LN07]	15%	7.5%	180 bits	FVC2002-DB2 (partielle)
Tuyls *et al.* [TAK+05]	1.4%	5.3%	40 bits	FVC2000-DB1 (partielle)
Notre approche	10.25%	11.40%	630 bits	FVC2002-DB2(partielle)

TABLE 3.7: Comparaison des méthodes de protection basées descripteurs globaux

Les résultats qui retiennent l'attention sont ceux où l'EER apparait le plus faible dans le cas *Worst*. Dans Tuyls *et al.* [TAK+05], l'EER est à 5.3%. Néanmoins, la taille du code obtenu est de 40 bits ce qui est très faible en termes de complexité. Dans l'approche de Teoh *et al.* [TKKA10], l'EER est très appréciable puisque il est de 2.39%. Cette approche utilise du BioHashing avec une binarisation multi-échelle. Elle nécessite une phase d'apprentissage puisque les seuils de binarisation sont estimés à partir de la distribution du vecteur W (Rappel : $W = XR$). Dans le cas des références multiples et en connaissant tous les paramètres de l'algorithme, cette transformation peut être totalement inversible puisque les seuils sont liés à W.

Dans cette section, un système révocable à partir du descripteur de texture de l'empreinte a été proposé. Le descripteur est déduit de l'analyse de l'image par les filtres de Gabor. Il utilise le point core comme repère de référence et une phase d'apprentissage pour gérer les problèmes de rotation. Un vecteur binaire de taille importante (630 bits) est généré lors de la phase de transformation. Il est sauvegardé sur la base de données comme référence. Les résultats expérimentaux ont montré que l'approche, principalement, la projection aléatoire, réunit les propriétés de révocabilité et de diversité souhaitées. C'est là sont point fort. Cependant, elle présente d'autres insuffisances et a besoin d'être améliorée. Les points à soulever sont les suivants :

1. Comme toutes les approches basées descripteurs globaux, ce schéma repose sur la détection d'un point de référence. Les tests ont montré que ce point peut ne

pas exister ou ne pas être correctement détecté sur toutes les images. Cela influe négativement sur le taux d'échec à l'enrôlement. Un point important qui peut être contraignant pour l'utilisateur qui est face à la borne biométrique.

2. Le risque d'intrusion dans le cas de vol de clé reste trop important pour envisager un système applicable dans la réalité.
3. Les risques de réversibilité (partielle heureusement) doivent être supprimés du système révocable.

Nous considérons plusieurs solutions pour faire face à ces limitations. Nous commençons par améliorer la représentation de l'empreinte. Le système biométrique, tel que proposé opère à 10.25% de taux d'erreur. Avoir un meilleur système de base est en lui même une solution, de facto, pour améliorer les performances du système après protection. Cette option constitue l'objectif de la prochaine section.

3.3 Schéma révocable basé descripteurs locaux d'empreintes digitales

On constate facilement que la représentation de l'empreinte, à partir de ses descripteurs globaux, même si elle opère mieux sur certaines images et est assez souhaitée pour certaines fonctions de transformation, souffre de performances inférieures par rapport à la reconnaissance utilisant les minuties. Nous allons considérer dans la suite un schéma de protection des empreintes à partir de la représentation des minuties.

En considérant les minuties, les problèmes classiques à traiter sont les suivants :

1. Les distorsions géométriques entre deux ensembles de minuties.
2. L'ensemble des minuties est désordonné et de taille variable.
3. La création de fausses minuties et le non-chevauchement des empreintes.

L'utilisation du BioHashing avec les minuties est assez limitée dans la littérature. Cela n'est pas très étonnant vu leurs natures assez opposées (vecteur *vs.* ensemble de points). Les seules contributions faites dans ce sens, sont à notre connaissance, celles de Yang *et al.* dans [BYB10] puis dans [YHSB10]. Dans ce travail, chaque minutie est traitée par rapport à son voisinage local. En sélectionnant ses 3 plus proches voisins en terme de distance euclidienne, un descripteur X de 36 valeurs réelles est déduit (il comprend, entre autres, les

orientations relatives entre les minuties du quadruplet en question). Chaque descripteur est ensuite transformé par BioHashing pour générer un biocode de 36 bits. Le modèle protégé consiste donc en un ensemble de M biocodes, de 36 bits chacun, M étant le nombre de minuties. Lors de la vérification, le score de décision est calculé en trouvant le meilleur appariement entre les différents biocodes.

Le système, sans le module de protection, opère avec un EER=12%, sur la FVC2002-DB2. Après le BioHashing, la référence [YHSB10] indique un EER=5.99% en supposant le scénario de vol de clé. Sur la référence [BYB10] le même système est présenté mais en indiquant un EER à 5.64%, en considérant le scénario classique (sans vol de clé donc). Il est vraiment curieux de voir que le système, opère presque de la même manière pour les deux scénarios Worst case et Best case, et qu'il donne de meilleurs résultats que le système de base en supposant le pire cas de vol de clé. En passant outre ces imprécisions, nous notons deux améliorations intéressantes pour relever le problème de complexité du biocode, qui est de 36 bits, ce qui est assez faible en terme de sécurité :

- Les auteurs proposent d'avoir des codes de taille plus grande en générant plus de matrices de projection. Cela est utile mais en même temps, il devient plus facile à l'attaquant de trouver l'inverse partielle, puisque le système d'équations linéaires défini dans l'équation 3.7 serait complètement déterminé.
- Afin d'augmenter la complexité actuelle de 36 bits, les auteurs proposent que le choix des colonnes de la matrice de projection R soit dépendant de la distribution des vecteurs biométriques X. Pour chaque dimension j, $j = 1, ..., 36$, un choix parmi 8 colonnes possible est proposé. La complexité est dans ce cas égale à $8^{36} = 2^{108}$. Les auteurs proposent 7 seuils $[T_1, ..., T_7]$ pour sélectionner la colonne adéquate. Les seuils proposés sont centrés autour de la moyenne des éléments de tous les vecteurs X extraits de la base de données. Cette moyenne a été définie à 0 et les seuils suivants ont donc été proposés : $[-75, -50, -24, 0, 25, 50, 75]$. Nous pensons que cette proposition apporte des nouveautés par rapport à l'approche classique du BioHashing. Néanmoins, il aurait été intéressant d'étudier la distribution des vecteurs X, en considérant par exemple la variance, pour garantir que le choix des colonnes est complètement aléatoire et que par rapport aux seuils choisis, ce ne sont pas les mêmes colonnes qui ont tendance à être sélectionnées à chaque fois. Dans ce cas, la complexité devient inférieure à 2^{108}.

Nous proposons dans ce chapitre, une extension de l'approche texture présentée dans la section 3.2, pour les minuties. L'empreinte digitales est représentée par deux descripteurs : un descripteur de texture autour de la minutie pour capturer l'information structurelle

avoisinante et un descripteur basé minutie qui définit les relations entre chaque minutie et son voisinage local. Nous proposons de protéger le descripteur de texture par BioHashing tout en gardant le descripteur basé minutie en clair. Même s'il reste en clair, l'analyse de sécurité montre que cela n'engendre aucune menace en terme d'attaques.

Nous utilisons, pour l'extraction des minuties, la librairie *libcubs* disponible sur l'URL [*http* : *//www.codeforge.com/article/*189863]. Elle est fournie par le CENTER FOR UNIFIED BIOMETRICS AND SENSORS DE L'UNIVERSITÉ DE NEW YORK. Dans ce qui va suivre, nous explicitons les détails de notre approche.

3.3.1 Création du descripteur biométrique

Le création du descripteur local d'empreintes digitales consiste en deux modules principaux : (1). la sélection du descripteur biométrique, et (2). la gestion des problèmes d'alignement, détaillés dans ce qui suit :

1. **Sélection du descripteur biométrique**
   Comme illustré sur la figure 3.26, le descripteur biométrique est composé de MinuCodes et de K-plets.

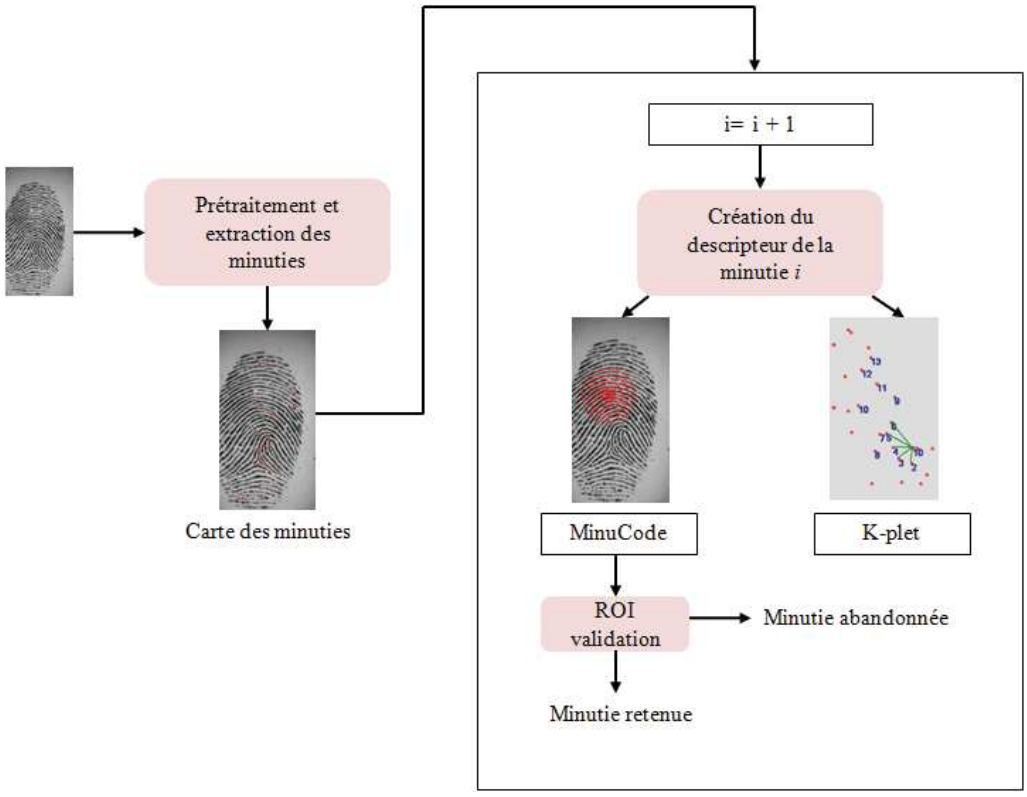
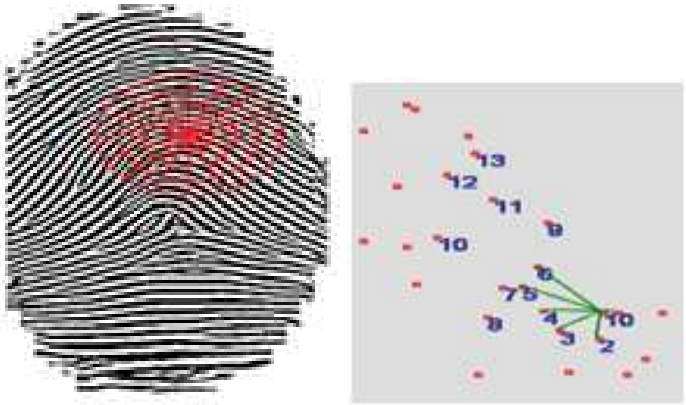


FIGURE 3.26 – Création du descripteur local d'empreinte digitale

Nous définissons autour de chaque minutie, une région d'intérêt circulaire et sectorisée de la même manière que pour les descripteurs globaux. La ROI consiste en B bandes concentriques, chacune large de b pixels. Chaque bande est découpée en 16 secteurs à angle égal. Le calcul du descripteur de texture est conforme à notre étude présentée dans la section 3.2.2. Les étapes reprises sont : Amélioration de l'image, localisation et validation de la région d'intérêt (autour de la minutie au lieu du point core) et extraction du vecteur caractéristique. Ce vecteur est de taille $B \times 16 \times 8$ et sera nommé *MinuCode*. La figure 3.27 *(Fig.a)* illustre cette région d'intérêt.



(a) (b)

FIGURE 3.27 – Descripteurs locaux (a) ROI autour de la minutie (b) *6-plet* de la minutie de référence labélisée *1*

A ce stade, il est important d'argumenter le choix de la forme circulaire par rapport à la forme carrée, pourtant plus performante. Principalement, pour les deux raisons suivantes :

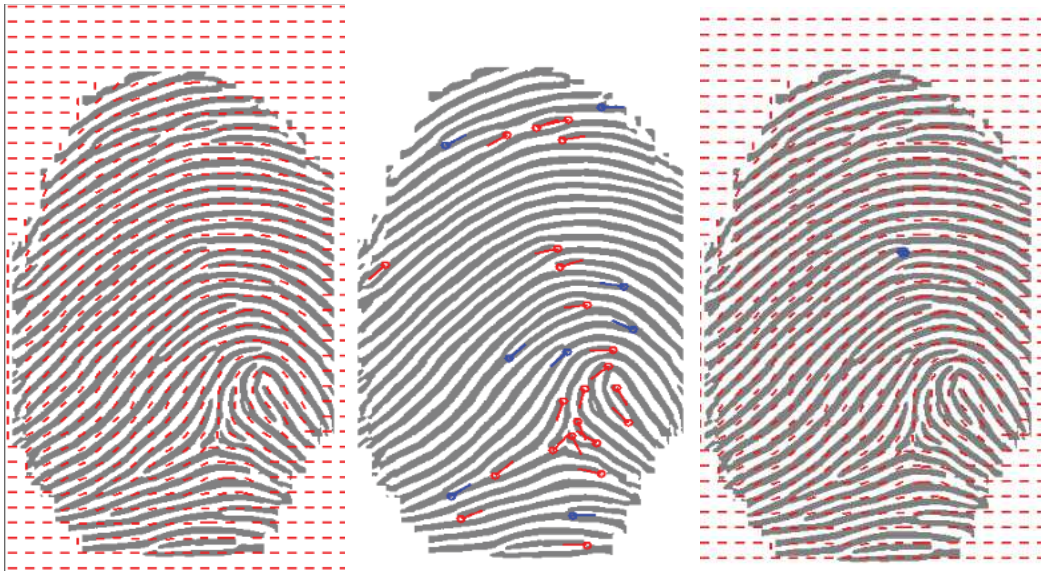
– La région carrée occupe plus d'espace que la région circulaire. Beaucoup de minuties risqueraient de voir leur ROI carrée déborder.
– La gestion de la rotation de la région carrée est plus contraignante. Nous avons utilisé une base d'apprentissage alors que les bases de tests disponibles et largement utilisées par la communauté sont déjà assez réduites (8 échantillons par personne).

Il est aussi important de noter qu'entre des minuties voisines, il y aura certainement des secteurs de la ROI qui vont se chevaucher. Techniquement, cela n'engendre pas de redondance dans le vecteur caractéristique car : les secteurs n'ont pas la même position dans le vecteur caractéristique et possèdent par rapport à leur minutie de référence, des orientations opposées impliquant des réponses de Gabor différentes.

De surcoît, nous représentons les relations locales entre une minutie et son voisinage par une structure appelée *K-plet*. Nous considérons le *K-plet* comme une information de renforcement, en vue de vérifier si l'appariement des descripteurs de texture est consistant sur le niveau global. Par conséquent, un *K-plet* est formé à partir de la minutie de référence en considérant ses K plus proches voisins en terme de distance euclidienne. Les voisins sont ensuite triés suivant l'ordre croissant de leurs directions relatives à la minutie de référence. La direction relative $dir_{relative}$ d'une minutie m_i de direction dir_i par rapport à une minutie m_j de direction dir_j est calculée suivant l'équation 3.8.

$$dir_{relative} = (dir_i - dir_j + 2\pi) \mod 2\pi \tag{3.8}$$

Il est important de noter les différences entre une orientation qui est dans l'intervalle $[0, \pi[$ et une direction qui est plutôt dans l'intervalle $[0, 2\pi[$. La figure 3.28 montre une image d'orientation (*Fig. a*) telle que calculée par l'algorithme 4. La carte des minuties extraites par la librairie *libcubs* est aussi représentée (*Fig. b*). La minutie sélectionnée (i.e. en bleu sur la figure *Fig. c*) possède une orientation égale à 17.88° par contre elle présente une direction de 195°. L'orientation et la direction dépendent de la crête sur laquelle se positionne la minutie.



(a) (b) (c)

FIGURE 3.28 – Différentes images résultats. (a) Image d'orientation, (b) Carte des minuties, (c) Minutie sélectionée

Un *K-plet* de la minutie de référence min_{ref} est l'ensemble des minuties $\{a_j\}_{j=1}^{K}$ spatialement les plus proches et dont les directions relatives à min_{ref} sont croissantes. Il est bien connu que l'inconvénient des structures basées voisinage, tel que notre

K-plet, est la possibilité de manquer ou d'ajouter un voisin (à cause des problèmes d'oubli ou de fausses minuties du module d'extraction). Une technique pour les comparer serait alors d'accomplir un appariement élastique en utilisant la programmation dynamique, par exemple. Néanmoins, la programmation dynamique est sensible à l'ordre des chaines en entrée. Plus cet ordre est précis, plus nous évitons les minima locaux. D'après nos tests, l'ordre proposé, suivant le critère de *direction relative* reste le plus pertinent.

Par la suite, seules les minuties dont la région d'intérêt a été validée par l'algorithme 6, présenté précédemment, seront retenues dans le modèle final (Rappel du principe : chaque secteur est dans les limites de l'image et présente une alternance entre crêtes et vallées) . En fait, il y a habituellement 30-60 minuties dans une empreinte digitale. Dans notre algorithme, nous ne sélectionnons que les minuties avec une ROI valide (implicitement les plus centrales), générant un modèle de 10-20 minuties, considéré néanmoins comme suffisant s'agissant de structures locales comme précisé par Zhang *et al.* [WZ02]. Par conséquent, nous pouvons ajuster la surface de la ROI selon le nombre des minuties, qui à son tour, dépend de la résolution du capteur.

Soit $M = \{m_i\}_{i=1}^{\mu}$ l'ensemble des minuties extraites de l'image d'empreinte. Le modèle final est noté :

$$M = \{m_i\}_{i=1}^{\nu} = \{MinuCode_i + \{a_j\}_{j=1}^{K}\}_{i=1}^{\nu} \text{ avec } \nu \leq \mu$$

.

2. **Gestion des problèmes d'alignement**
   Alors que la translation est implicitement gérée par l'utilisation d'un point de référence, le problème de rotation reste plus critique à traiter. Nous adoptons une méthode basée sur la *correction de l'orientation de référence* de l'image d'empreinte. Pour chaque image, nous calculons l'orientation de référence θ_{ref}. Cette orientation de référence est calculée à partir du point core et elle n'est pas à confondre avec l'orientation que nous pouvons déduire de l'image d'orientation.

   Nous proposons une méthode basée sur l'ajustement par les moindres carrées, mais en incorporant les pixels des régions grises de le figure 3.29 au lieu d'un bloc $w \times w$ comme dans l'algorithme 4. Le but est de calculer la moyenne des orientations des crêtes concaves à la confluence du point core. Les tests empiriques ont démontré plus de stabilité avec cette région de calcul. L'algorithme 10 résume les détails de cette approche.

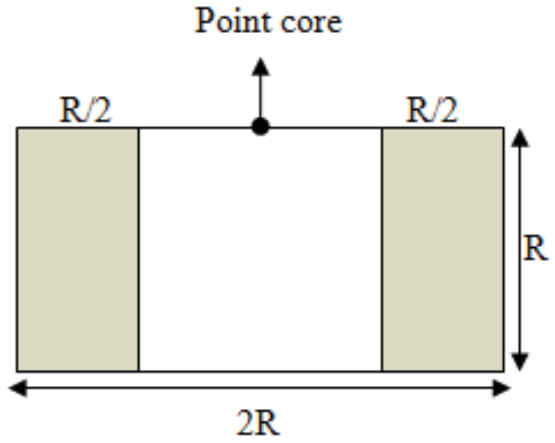


FIGURE 3.29 – Région d'intérêt pour la calcul de l'orientation de référence

---

Algorithme 10 : Détection de l'orientation de référence de l'image d'empreinte

---

Input : Région d'intérêt : la région autour du point core de la figure 3.29 de rayon R (R est pris à 25 pixels)

Output : θ_{ref}

1 Estimer, en utilisant les masques de Sobel, le vecteur gradient $\begin{pmatrix} G_x \\ G_y \end{pmatrix}$ de chaque pixel (i, j) des régions grises dans la figure 3.29

2 Estimer par la méthode des moindres carrées, l'orientation de référence θ_{ref} à partir des formules suivantes :

$$V_x = \sum_{v=0}^{R} 2(\sum_{u=0}^{\frac{R}{2}} G_x(u,v)G_y(u,v) + \sum_{u=\frac{3R}{2}}^{2R} G_x(u,v)G_y(u,v))$$

3

$$V_y = \sum_{v=0}^{R} 2(\sum_{u=0}^{\frac{R}{2}} (G_x^2(u,v) - G_y^2(u,v)) + \sum_{u=\frac{3R}{2}}^{2R} (G_x^2(u,v) - G_y^2(u,v)))$$

4

$$\theta_{ref} = \frac{1}{2}\tan^{-1}(\frac{V_x}{V_y})$$

Un exemple de l'orientation de référence est donné sur la figure 3.30 où nous pouvons aussi déduire un repère de référence invariant pour l'empreinte.

FIGURE 3.30 – Axes de référence d'une image d'empreintes

Pour faciliter la phase de test, nous pouvons approximer θ_{ref} par l'une des orientations fondamentales appartenant à l'intervalle suivant :

$$\theta_{ref} \in \{(k-1)\frac{\pi}{period}, k = 1, 2, ..., period\}$$

period exprime la précision de l'angle d'orientation. En choisissant *period* à 16 ou à 32, nous pouvons gérer les décalages de ±11.25° ou de ±5.625°.

Pour comparer deux empreintes, une phase d'enregistrement relatif est d'abord accomplie suivant l'algorithme 11.

Algorithme 11 : Enregistrement relatif par correction de l'orientation de référence

Input : θ_{1ref} : l'orientation de référence de l'image enrôlée
θ_{2ref} : l'orientation de référence de l'image en entrée

```
diff ← min(| θ1ref − θ2ref |, 360− | θ1ref − θ2ref |)
if θ1ref < θ2ref then
    diff ← −diff
    imrotate(input image, diff)
```

Nous évaluons sur la figure 3.31, de façon qualitative, ce processus d'enregistrement en considérant plusieurs types d'images. Nous remarquons que la correction fonctionne bien. Cependant, son principal inconvénient est que l'estimation dépend

du point core. Les séries d'expériences de la section 3.2.2 montrent que ce point n'est pas nécessairement bien détecté dans toutes les images. Le taux d'échec à l'enrôlement (FTE) devient dans ce cas important.

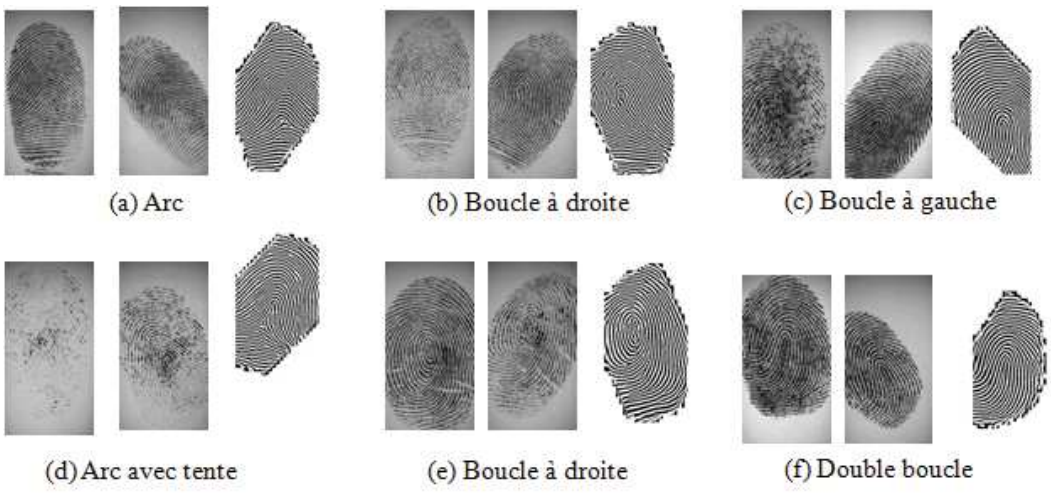


FIGURE 3.31 – Appréciation qualitative de la gestion de rotation par orientation de référence sur différents types d'empreintes (A chaque fois nous avons l'image de référence, l'image en entrée et l'enregistrement de l'image en entrée relativement à l'image de référence)

Pour pallier ce problème, nous proposons une gestion de la rotation sur deux niveaux. La figure 3.32 en résume le principe. Nous commençons par lancer le processus de détection du point core, déjà décrit dans l'algorithme 5. Ce point core est validé suivant le même principe qu'une région d'intérêt, c'est à dire : *(i)* le point core est à l'intérieur de l'empreinte et *(ii)* Une région circulaire de rayon r autour du point core représente une alternance entre crêtes et vallées. Si le point core est validé, une inscription basée sur l'orientation de référence est utilisée. Si, au contraire, ce point n'est pas validé, un MinuCode auto aligné est alors créé. L'auto-alignement des descripteurs locaux a été proposé dans différents travaux, principalement dans[TK03, BAH+07]. La direction de la minutie est utilisée comme axe de référence pour la ROI à la place de l'axe horizontal. L'idée d'utiliser deux alternatives pour la gestion de la rotation, alors que la seconde aurait suffit, a comme but de diminuer les risques d'erreur. En utilisant systématiquement la direction des minuties comme référence, il y aurait autant de possibilités d'artefacts de calcul que de minuties. En utilisant le point le core, les tests ont montré que l'algorithme ne commettait presque aucune erreur de détection (5 images sur 800) par contre il avait un FTE assez élevé qui est de 9%. En combinant les deux approches, il y a une optimisation aussi bien sur les incertitudes de détection que sur les échecs à l'enrôlement.

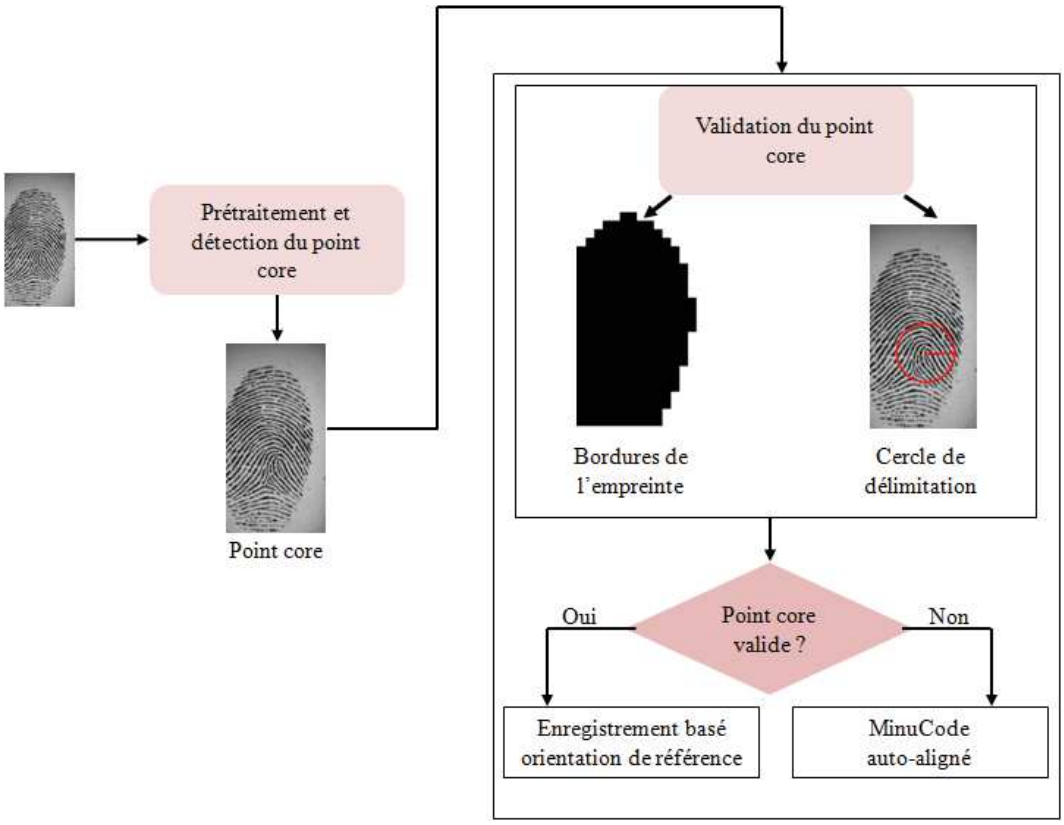


FIGURE 3.32 – Principe de gestion de la rotation

3.3.2 Protection du descripteur biométrique

Chaque minutie $m_i, i = 1, ..., \nu$ avec ν le nombre de minuties sélectionnées dans l'empreinte est représentée par son *MinuCode* et son *K-plet*. Ce modèle, comme il est basé sur des descripteurs de texture autour de la minutie et non sur l'information de minuties, apporte déjà plus de préservation de vie privée dans le sens où il masque l'information initiale des minuties. Il n'est donc pas possible de faire une reconstruction de l'image d'empreintes à partir des minuties en appliquant les attaques citées dans [CLMM07b, FJ09]. L'attaque par mascarade n'est donc pas possible. Néanmoins, les contraintes de révocabilité et d'intraçabilité demeurent. Pour réunir ces exigences, nous décidons d'appliquer le BioHashing avec sa $configuration4$ (cf. page 109) aux différents MinuCodes. Par conséquent, nous obtenons un MinuCode protégé noté *PMinuCode*.
Le modèle local protégé MLP sera représenté par la notation suivante :

$MLP = \{PMinuCode_i + \{a_j\}_{j=i}^{K}\}_{i=1}^{\nu}$, avec a_j un label pour représenter les minuties voisines.

Nous rappelons dans l'algorithme 12 le processus général d'extraction d'un modèle local d'empreintes digitales protégé.

Algorithme 12 : Processus de génération d'un descripteur local d'empreintes digitales révocable

Input : I : Image d'empreinte en entrée
B : L'image I binarisée
M : l'ensemble de taille μ des minuties extraites de l'image d'empreintes I
S : la clé attribuée à l'utilisateur U sous forme de nombre aléatoire
Output : MLP : le modèle local protégé

$OrientationField \leftarrow GetSmoothedOrientation(I)$ Extraction de l'image d'orientation et son adoucissement suivant l'algorithme 4
$pt \leftarrow CoreDetect(I)$ Extraction du point core et de ces axes de référence
$valid \leftarrow CoreValidation(B, pt.x, pt.y)$ Validation du point core
if *valid* **then**
    Sauvegarder(pt.orientation) Sauvegarder l'orientation de référence de l'image
for $i \leftarrow 1$ **to** μ **do**
    $cpt \leftarrow 0$ pour compter le nombre de minuties sélectionnées
    $directionH \leftarrow 0$ direction de l'axe pour l'extraction de la ROI circulaire (initialement, il s'agit de l'axe horizontal)
    if *not valid* **then**
        $directionH \leftarrow M[i].direction$
    $MinuCode \leftarrow$ new byte $[16 \times 8 \times B]$
    $MinuCode \leftarrow ExtractMinuCode(B, directionH)$
    Utilisation des filtres de Gabor sur chaque secteur de le ROI et normalisation dans l'intervalle $[0, 255]$
    if *MinuCode valide* **then**
        $MinutiaSet[cpt].descriptor1 \leftarrow MinuCode$
        $graph \leftarrow AddMinutia(M[i].x, M[i].y)$
        $cpt \leftarrow cpt + 1$
$v \leftarrow cpt$ le nombre de minuties du modèle final
if $v < K$ **then**
    return $FailToEnroll$
else
    for $j \leftarrow 1$ **to** v **do**
        $MinutiaSet[j].descriptor2 = ConstructKPlet(graph, K)$ Construction du K-Plet
for $j \leftarrow 1$ **to** v **do**
    $MLP[j].descriptor1 \leftarrow$ Lancer algorithme 8 en utilisant la configuration 4 pour calculer le biocode du vecteur $MinutiaSet[j].descriptor1$ à partir de la clé S
    $MLP[j].descriptor2 \leftarrow MinutiaSet[j].descriptor2$

Il est question maintenant, de présenter dans la section suivante, le processus de

comparaison entre deux modèles protégés qui consiste à trouver le meilleur alignement entre elles. Cette tâche est plus critique s'agissant de descripteurs locaux, alors qu'elle se basait sur une simple distance de Hamming en utilisant le descripteur global.

3.3.3 Comparaison des empreintes

Il est question de développer un algorithme de comparaison, dans le domaine de la transformation, qui comptabilise le nombre de minuties appariées entre deux empreintes. L'algorithme reçoit en entrée deux modèles protégés et retourne un score de similarité. Soient MLP_A, le modèle protégé de référence contenant N minuties et MLP_B le modèle en entrée contenant M minuties. Nous cherchons à identifier les paires de minuties $C = \{(a_i, b_j)/a_i \in MLP_A \text{ and } b_j \in MLP_B\}$.
Intuitivement, deux minuties sont appariées si elles satisfont des contraintes locales et globales : *(i)* la contrainte locale est satisfaite par la paire (a_i, b_j) si la distance de Hamming $D(a_i, b_j)$ entre leurs *PMinuCode* est suffisamment petite relativement à un certain seuil. *(ii).* la contrainte globale est satisfaite si la minutie a_i est géométriquement proche de la minutie b_j (le K-plet est utilisé pour gérer cette contrainte).
Le processus de comparaison est maintenant divisé en trois étapes :

1. **Phase 1** : il s'agit de sélectionner la paire de minuties $(root_A, root_B)$ la plus probable. Deux minuties a_i et b_j sont appariées si la distance de Hamming entre leur PMinuCodes est assez petite. Cependant, a_i peut aussi présenter une petite distance avec une autre minutie $\grave{b}_j \in MLP_B$ différente de b_j. Ainsi, pour trouver la paire de minuties la plus signifiante, l'idée est de minimiser la probabilité de l'équation 3.9 comme dans [TK03]. Le numérateur définit une petite distance entre a_i et b_j et le dénominateur définit de larges distances pour a_i et b_j par rapport aux autres minuties dans MLP_B et MLP_A respectivement.

$$P(a_i, b_j) = \frac{D(a_i, b_j)^2}{\sum_{\grave{i}=1}^{N} D(a_{\grave{i}}, b_j) + \sum_{\grave{j}=1}^{M} (D(a_i, b_{\grave{j}}) - D(a_i, b_j))} \tag{3.9}$$

   La paire $(root_A, root_B)$ est validée seulement si $P(a_i, b_j) < threshold$. Si $(root_A, root_B)$ est validée alors aller à $phase2$ sinon aller à $phase3$.

2. **Phase 2** : Considérons $root_A$ et $root_B$ les premiers nœuds à explorer dans MLP_A et MLP_B respectivement. Nous développons un algorithme de recherche de type *BFS* (Best First Search) inspiré des travaux de Chikkerur *et al.* [CCG06]. En utilisant la programmation dynamique, nous comparons le *K-plet* de la minutie $root_A$ avec

celui de la minutie $root_B$, comme explicité par l'algorithme 13. Chaque paire de minuties appariées sera empilée dans une queue et marquée comme visitée. Ce schéma est récursivement répété jusqu'à ce que la pile devienne vide. A chaque itération, une nouvelle paire de minuties est dépilée de la queue et ses K-plets respectifs sont comparés. Pour éviter le problème des minima locaux, aller à la *phase 1* en considérant uniquement les nœuds non visités.

Algorithme 13 : Appariement de deux K-plets par programmation dynamique

Input : $K - plet_A = \{a_1, a_2, ..., a_K\}$, un ensemble de minuties chacune définie par son PMinuCode
$K - plet_B = \{b_1, b_2, ..., b_K\}$, un ensemble de minuties chacune définie par son PMinuCode

1 Les minuties du $K - plet_A$ ou du $K - plet_B$ sont respectivement ordonnées par ordre croissant de leur direction relative par rapport à leur minutie de référence
2 Initialiser la matrice de programmation dynamique C_{KK} à 0
3 **for** $i \leftarrow 1$**to**K **do**
4 **for** $j \leftarrow 1$**to**K **do**
5 Calculer la fonction coût :
 $X \leftarrow Distance_{Hamming}(PMinuCode_{a_i}, PMinuCode_{b_j})$
6 $$C(ij) \leftarrow \min \left\{ \begin{array}{l} C(i-1, j) + OHM \\ C(i, j-1) + OHM \\ C(i-1, j-1) + X \end{array} \right\}$$
 OHM est une constante pour mesurer le coût d'une insertion ou d'une délétion
7 Faire un retour arrière pour trouver le chemin minimum

3. **Phase 3** : Il s'agit de calculer le score de similarité. Dans un système de reconnaissance automatique, le nombre de minuties appariées MA est d'abord converti, par normalisation, en un score de similarité. D'après nos tests, nous trouvons que le score suivant : $\frac{MA}{\min(N,M)}$ est le plus approprié avec N le nombre de minuties de référence et M le nombre de minuties en entrée.

Il est important de mentionner que cet algorithme de comparaison est quelque peu modifié suivant les cas d'utilisation du système, en *mode réel* ou en *mode test* :

– *En mode réel* : Lors du processus d'enrôlement, un modèle local protégé (MLP) est généré et sauvegardé comme référence. Si le point core de l'image d'enrôlement existe alors l'orientation relative est calculée suivant l'algorithme 11 et est sauvegardée en même temps avec le MLP de référence. Si ce point n'est pas validé alors

des PMinuCodes auto-alignés sont extraits. Lors de la vérification, si le point core existait dans l'image de référence alors il devrait aussi exister dans l'image en entrée. Si au bout de X tentatives, l'utilisateur devant la borne biométrique ne fournit pas une empreinte avec un point core valide alors le système va systématiquement le refuser. La figure 3.33 illustre le fonctionnement général du système d'authentification.

– *En mode test* : comme les images font partie d'une base de données universelle et afin de maximiser le nombre d'images enrôlées, nous calculons même pour les images avec des points cores valides, un autre MLP avec des PMinuCodes alignés. Ainsi lors des tests, chaque image enrôlée peut être comparée avec toutes les images de la base de données.

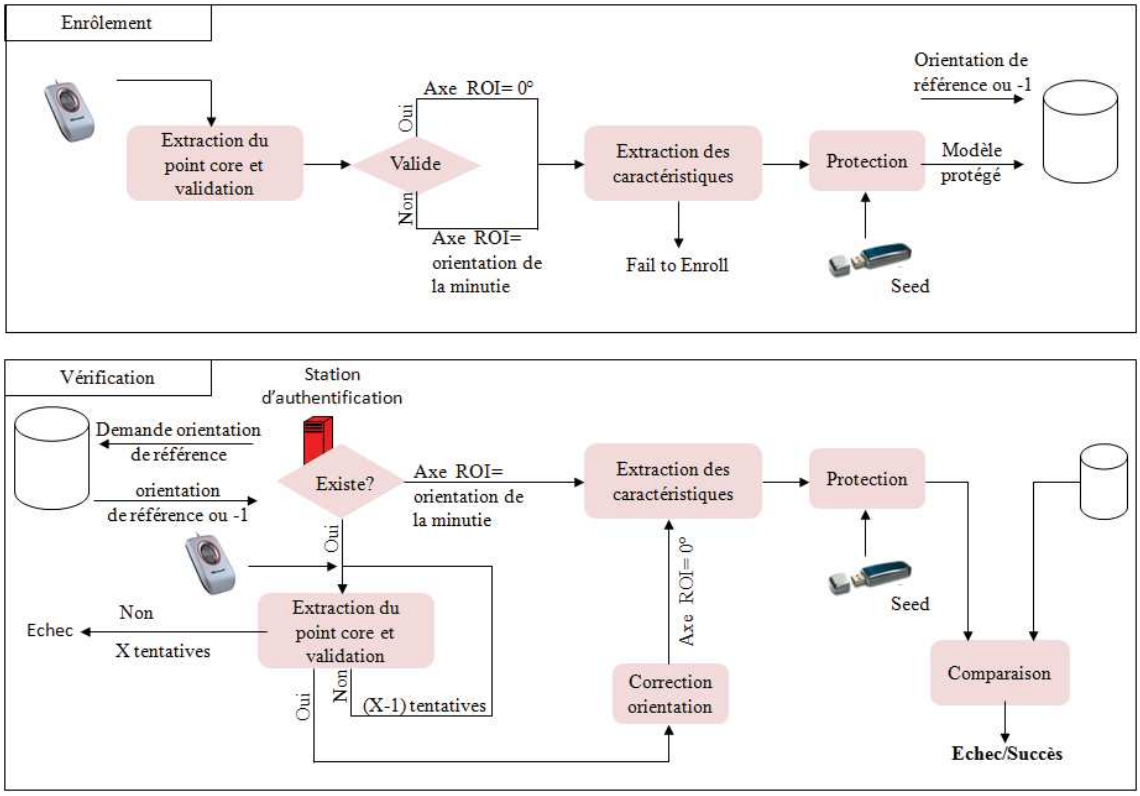


FIGURE 3.33 – Fonctionnement du système de biométrie révocable proposé

3.3.4 Résultats expérimentaux

Comme précédemment, nous utilisons la FVC2002-DB2 comme base de test. Cet intérêt pour cette base est dû à sa large utilisation par la communauté qui travaille sur la même thématique.

Nous posons $B = 3$, le nombre de bandes de la région d'intérêt circulaire impliquant des MinuCodes de taille $B \times 16 \times 8 = 384$ éléments.

La longueur du PMinuCode, c'est à dire le biocode du vecteur caractéristique est maintenue à $m = 180$ bits. D'après nos tests, chaque K-plet est mieux représenté lorsque $K = 6$.

Nous commençons par évaluer les performances de la vérification biométrique concernant trois scénarios : le système de vérification de base sans le module de protection (*SOLE Biometric*), le système de vérification avec protection lorsqu'aucune attaque n'est rencontrée (*BEST case*) et lorsque l'imposteur connait toujours la clé du client (*Worst case*).
Pour simuler le cas *Best*, nous attribuons à chaque sujet de la base de données une clé différente. Pour le cas *Worst*, nous utilisons la même clé pour toute la base de données.
Nous suivons, pour les tests, le même protocole que la FVC2002-DB2. Pour calculer le FAR, le premier échantillon de chaque empreinte est comparé avec le premier échantillon des autres empreintes. Pour calculer le FRR, chaque échantillon de l'empreinte est comparé avec les autres échantillons du même individu. La performance sera exprimée en terme d'EER (Equal Error Rate).
Le FTE (Fail To Enroll) est à 1.25% alors qu'il était de 9% pour le système à base du descripteur global. Cela constitue une nette amélioration.
La performance du système de base (SOLE Biometric) est reportée sous forme de courbe DET sur la figure 3.34. L'EER est égal à 4.56%.

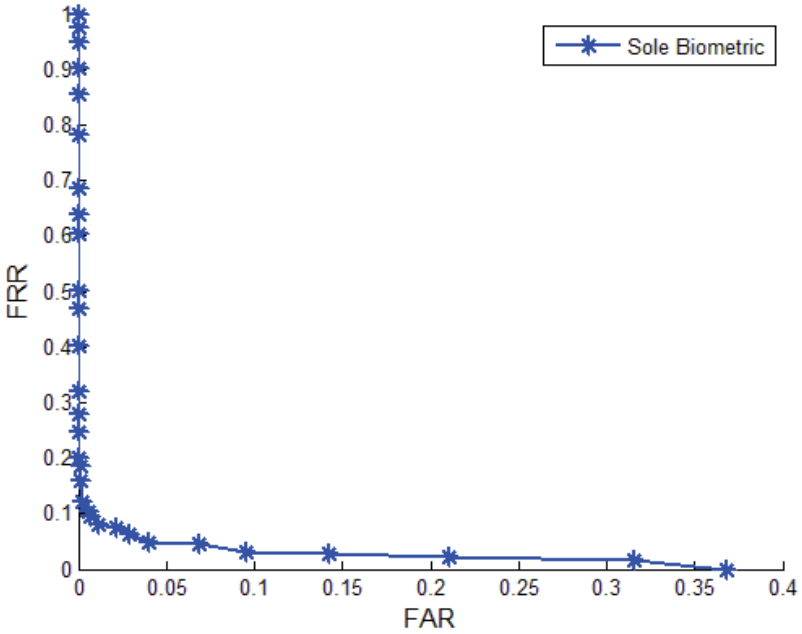


FIGURE 3.34 – Courbe DET du système de vérification par descripteurs locaux (sans protection)

En considérant le système avec protection, nous illustrons sur la figure 3.35, la distribution des scores du FAR et du FRR pout tous les seuils de décision possibles. Nous reportons les deux cas *Worst* et *Best* sur la même figure pour analyser l'impact du choix du seuil de décision. Le point d'intersection entre les deux courbes FAR et FRR représente l'EER.

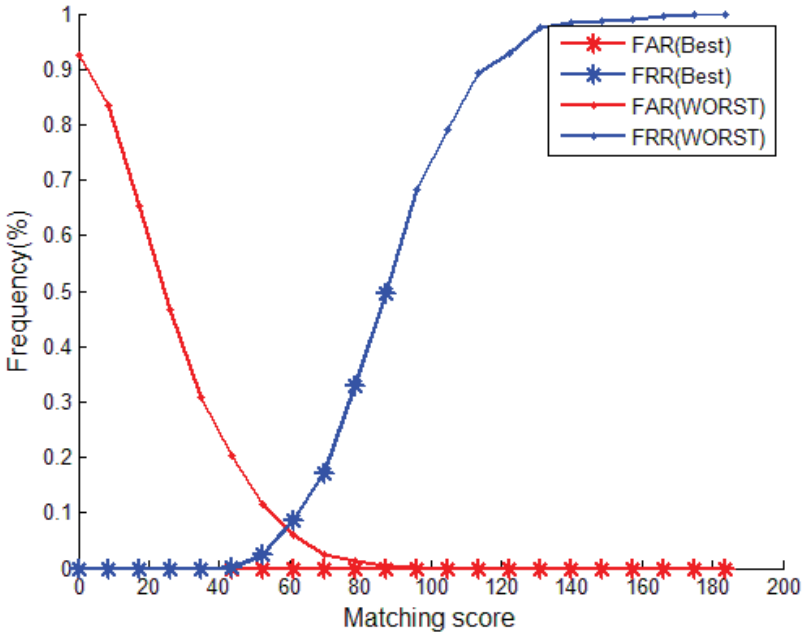


FIGURE 3.35 – Evolution des distributions du FAR et du FRR pour le Best case et le Worst case

A partir de la figure 3.35, nous observons que la performance du système de vérification biométrique par protection atteint un EER de 0% dans le cas *Best* et un EER de 6.68% pour le cas *Worst*.

Cependant, ces taux d'erreur sont faisables mais en utilisant des seuils de décision différents : 40% pour le cas *Best* et 60% pour le cas *Worst*. En pratique, il est inconcevable de mettre deux seuils de décision différents, un compromis doit donc être fait. Si nous ajustons le système pour obtenir $EER = 0\%$ sous l'hypothèse que la clé n'est jamais volée, le FAR dans le cas *Worst* est de 22%, ce qui est exagéré pour un système d'authentification. Il est alors plus approprié de mettre le seuil de décision à 60%. Les performances sont alors : $FAR = 0\%$ si la clé utilisateur n'est jamais volée, il est de 7.16% sinon. Le FRR est

dans les deux cas égal à 6.21%.
Pour fournir des résultats sur un grand ensemble de données, nous faisons des tests sur les quatre bases de la FVC2002. Le tableau 3.8 résume les résultats obtenus.

	Sole Biometric (%)	Best case (%)	Worst case (%)
FVC2002-DB1	1.91	1.75	3.78
FVC2002-DB2	4.56	3.1	6.68
FVC2002-DB3	8.95	4.89	10.87
FVC2002-DB4	10.94	5.38	12.39

TABLE 3.8: Performance du système de vérification par protection en terme d'EER sur toutes les base de la FVC2002

3.3.5 Evaluation du système révocable par descripteurs locaux

L'évaluation en termes de sécurité et de respect de vie privée se fait en fonction du schéma proposé dans le chapitre 2. Ce schéma a l'avantage d'être applicable quelque soit le modèle de transformation révocable utilisé.
Comme pour le système à base de descripteurs globaux, nous commençons par paramétrer le seuil de décision du système révocable ϵ_D. Nous posons $\epsilon_D = 60\%$ par rapport au scénario *Worst*. Le tableau 3.9 résume les valeurs des différentes métriques d'évaluation $A_1, ..., A_{17}$.

A1	A2	A3	A4	A5	A6	A7	A8	A9
6.68%	1	0%	0%	7.16%	0%	true	18432 bits	NI
A10	A11	A12	A13	A14	A15	A16	A17	
IPP	OF	ONF	0%	0%	0%	0	180 bits	

TABLE 3.9: Evaluation du système révocable par descripteurs locaux d'empreintes digitales

La figure 3.36 illustre l'évolution du FAR en fonction des seuils de décision possibles pour les métriques : A_3 (attaque à zéro effort), A_4 (attaque par force brute), A_5 (attaque par vol de clé), A_6 (attaque par vol de biométrie), A_{14} (attaque par écoute de 3 modèles différents d'un même utilisateur), A_{15} (attaque par écoute de 11 modèles différents d'un même utilisateur).
Nous remarquons qu'aucune attaque n'est effective, quelque soit le seuil de décision, le FAR est à 0% mis à part l'attaque par vol de clé qui peut réussir dans 7.16% des cas. Il est aussi à noter l'amélioration par rapport à l'approche par descripteurs globaux où la valeur de A_5 était à 13.11%.

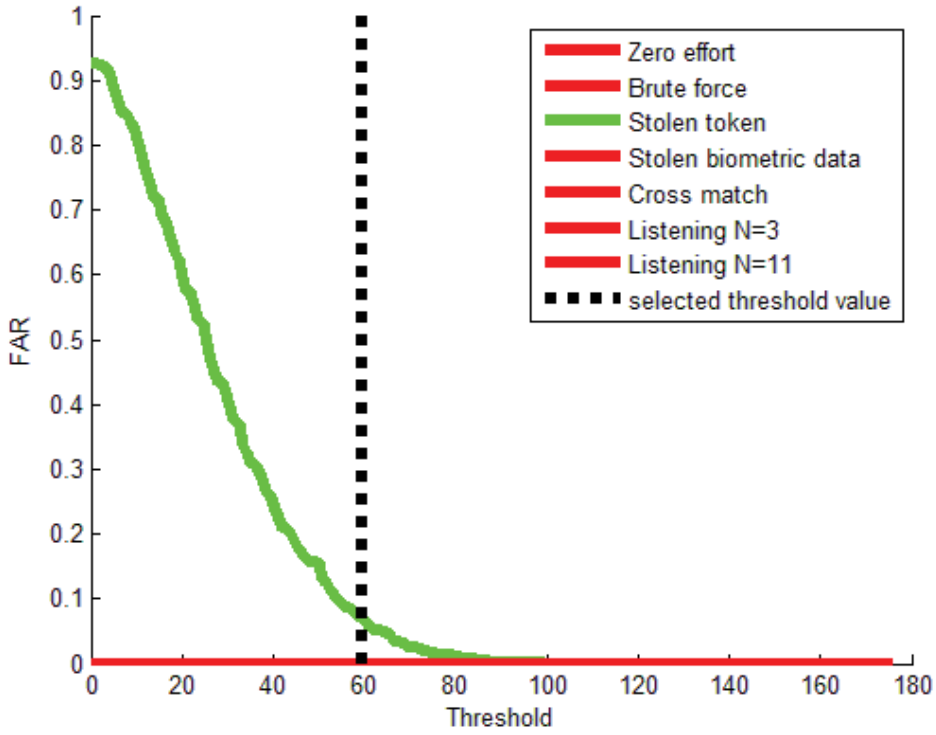


FIGURE 3.36 – Evolution du FAR pour différents scénarios d'attaque

Analyse de diversité : Comme préconisé, l'un des points forts de la méthode de transformation est lié à sa diversité. Le taux de compatibilité croisé (A_{13}) est à chaque fois à 0% pour tous les seuils de décision possibles. Cette indépendance du seuil de décision est un point important qui assure la non traçabilité des modèles révoqués ou inscrits sur différentes applications. Généralement, ce seuil de décision n'est pas discuté sur les différents papiers traitant de la question, alors qu'il est essentiel car en changeant d'applications, les paramètres du système peuvent aussi changer et donc être indépendant du seuil de décision est un point critique.

Le paramètre d'entropie de diversité a été calculé à partir de la distribution pseudo-imposteur. Comme pour le calcul du taux de compatibilité croisé, nous attribuons à chaque échantillon de la base de données, t différents modèles (nous faisons varier t entre 2 et 10 dans cette étude) en attribuant des clés S différentes. Nous calculons à chaque fois le score de similarité entre ces modèles et nous répétons cela de façon récursive sur toute la base de données pour obtenir la distribution pseudo-imposteur.

L'histogramme de la figure 3.37 qui représente la distribution pseudo-imposteur a une moyenne $P = 49.58\%$. Ceci indique que la probabilité de prédire une valeur correcte par bit est $1 - P = 50.52\%$, ce qui satisfait une ambiguïté presque totale.

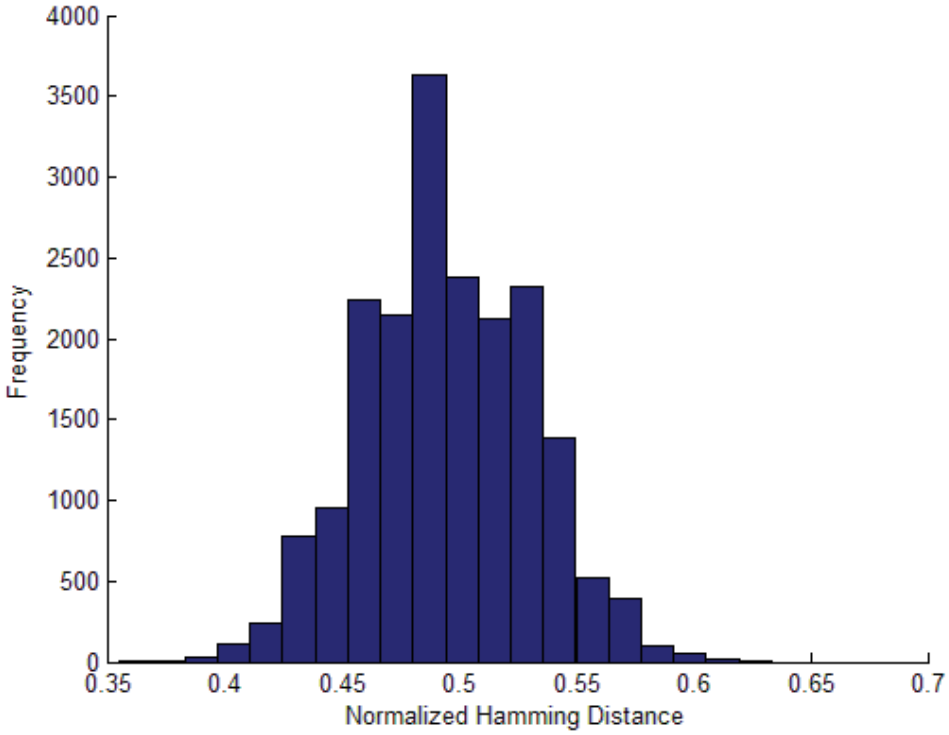


FIGURE 3.37 – Histogramme de la distribution pseudo-imposteur

Analyse de la non-inversibilité : Nous discutons maintenant des risques d'inversibilité de la méthode.

Nous supposons que l'attaquant est en possession de la matrice de projection $R(n, m)$ et de $(K + 1)$ *PMinuCodes* $(y_1, y_2, ..., y_{K+1})/y_i \in \{0, 1\}^m, i = 1, ..., K + 1$ qui correspondent à un K-plet et souhaite estimer les MinuCodes d'origine $(x_1, x_2, ..., x_{K+1})/x_i \in \Re^n$.

D'un point de vue mathématique, l'inversibilité peut être considérée comme le fait de trouver une solution au système d'équations linéaires défini par l'équation 3.10.

$$\begin{cases} R.x_1 = y_1 \\ \vdots \\ R.x_{K+1} = y_{K+1} \end{cases} \tag{3.10}$$

Dans ce système, il y a $(K + 1)n$ inconnues et $(K + 1)m$ équations, $(m < n)$. Nous avons donc plus d'inconnues que d'équations linéaires indépendantes. En partant du principe que les solutions possibles sont infinies, la projection aléatoire est considérée comme non inversible. Cependant, si l'attaquant possède le vecteur Médiane $(M_1, M_2, ..., M_m)$ des

seuils de décision, il pourrait de la même façon que pour les descripteurs globaux (voir section 3.2.4), faire une estimation des vecteurs $(\tilde{x}_1, \tilde{x}_2, ..., \tilde{x}_{K+1})$ qui peuvent générer des biocodes acceptables. La valeur de la métrique A_{11} est dans ce cas OF (Optimisation Faisable).
Il est cependant facile pour le module de vérification de remédier à cette attaque. Il suffit pour cela de compter le nombre d'éléments identiques dans le vecteur $\tilde{x}_i$. Si ce nombre est supérieur à une certaine limite (déduite à partir d'une phase d'analyse de la base de données) alors il va refuser ce vecteur d'entrée.

Nous considérons maintenant le cas des références multiples où l'attaquant pourrait connaître le lien entre I différents modèles diversifiés en même temps que leur matrice de projection $R_i, i = 1, ..., I$. Nous supposons que le rang des matrices concaténées $[R_1...R_I]$ est égal à n. Le système d'équations linéaires devient déterminé mais sa construction dans le cas des références multiples est difficile.

En effet, pour $n = 384$ et $m = 180$, I devrait être égal à 3. On suppose que l'attaquant est en possession de 3 modèles diversifiés du même individu. Pour résoudre le système d'équations, il faudrait qu'il associe dans le système les PMinuCodes de la même minutie. Le seul moyen de le faire est de procéder par force brute. En supposant, M le nombre de minuties moyen, l'attaquant devrait construire M^3 systèmes d'équations. Pour M=20, il s'agira d'une moyenne de 8000 systèmes d'équations. Dans notre système, il est impossible de trouver 3 modèles révoqués qui soient relatifs au même individu en plus de la difficulté de trouver le bon système linéaire, néanmoins nous mettons dans l'absolu, la valeur de A_{10} à IPP (Inverse Partielle Possible).

3.3.6 Etude comparative

Dans le but de situer notre contribution, nous présentons dans cette section une étude comparative entre notre approche et celles de la littérature qui traitent des minuties.
Mais d'abord, nous commençons par présenter quelques éléments de comparaison qui renforcent les différents choix de l'approche proposée.
En prenant comme base d'étude, le pire scénario *de vol de clé*, nous avons entrepris de faire des comparaisons où à chaque fois notre approche est confrontée aux différents tests suivants :

- *Test 1* : dans ce test, nous utilisons un processus d'appariement entre les deux modèles, de référence et en entrée, sans prendre en compte l'information de voisinage. Le but est d'analyser l'algorithme d'appariement proposé avec un algorithme par recherche gourmande dont le fonctionnement est explicité par l'algorithme 14.

Algorithme 14 : Appariement par recherche gourmande

Input : $A = \{a_1, a_2, ..., a_N\}$, un ensemble de N PMinuCodes associés au modèle de référence
$B = \{b_1, b_2, ..., b_M\}$, un ensemble de M PMinuCodes associés au modèle de référence

Output : *score* : score de similarité

Phase 1 : Calcul des distance de Hamming
$P \leftarrow$ new int $[N \times M]$
$index \leftarrow$ new int $[N \times M]$
$current \leftarrow 0$
for $i \leftarrow 1$**to**N **do**
  for $j \leftarrow 1$**to**M **do**
    $P[current] \leftarrow DistanceHamming(a_i, b_j)$
    $index[current] \leftarrow i \times M + j$
    $current++$

Phase 2 : tri du tableau P et élimination des redondances
Phase 3 : calcul du score de comparaison

– *Test 2* : dans ce test, nous utilisons uniquement les MinuCodes auto-orientés sans le second élément de gestion de la rotation qui est basé sur l'orientation relative à partir du point core.

– *Test 3* : dans ce test, on assigne au MinuCode la valeur de son descripteur de Tico proposé dans [TK03] au lieu du descripteur de Gabor. Le descripteur de Tico est basé sur une quantification des pixels autour de la minutie de référence comme illustré par la figure 3.38. Chaque pixel est ensuite représenté par son orientation relative à la minutie de référence. L'axe de référence du descripteur est conforme à la direction de la minutie.
– *Test 4* : ici, le MinuCode est issu de la fusion (i.e. par concaténation) entre le descripteur de Tico et le descripteur de Gabor.

En évaluant à chaque fois la valeur de l'EER, le tableau 3.10 résume les résultats obtenus. Cela démontre que les choix dans notre méthode restent les plus performants.

Notre approche	Test 1	Test 2	Test 3	Test4
6.68%	7.54%	7.46%	8.88%	7.77%

TABLE 3.10: Différents tests de comparaison en terme d'EER dans le cas de vol de clé

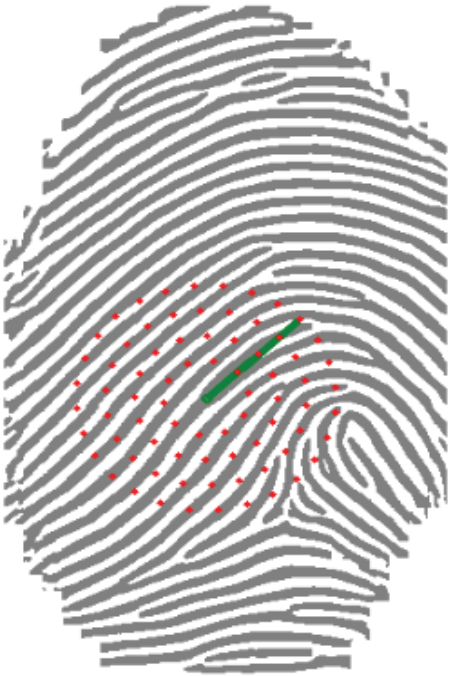


FIGURE 3.38 – Points de quantification du descripteur de Tico autour de la minutie de référence

Dans le tableau 3.11, nous présentons une étude comparative entre notre approche et quelques méthodes de la littérature basées minuties. Nous constatons que le problème de protection des modèles d'empreintes digitales n'est pas une tâche simple à réaliser. Par rapport aux méthodes existantes, nous estimons que nous sommes assez bien placés même si nous sommes conscients que des efforts restent à faire pour réaliser le meilleur compromis entre sécurité et préservation de vie privée.

3.4 Conclusion

La principale contribution de ce chapitre est la présentation d'une méthode révocable d'empreintes digitales qui est compétitive par rapport aux différentes approches existantes selon les critères suivants : 1/ Les EERs obtenus sur différentes bases de données par rapport au scénario de vol de clé sont d'actualité. 2/ La méthode offre une diversité quasi parfaite, indépendante des seuils de décision, ce qui lui confère un avantage certain par rapport aux autres méthodes.

Méthode	Base de données	EER(%)	Diversité (taux de compatibilité croisée)
Notre approche	FVC2002-DB2	6.68	0% et indépendante du seuil
	FVC2002-DB3	10.87	0% et indépendante du seuil
Ang *et al.* [ARM05]	NIST (80 images)	16.8	70%
Kumar *et al.* [KTG10]	FVC2002-DB2	4.98	Non mesurée
Lee and Kim [LK10]	FVC2004-DB2	10.3	0% mais dépendante du seuil
Ratha *et al.* [RCCB07]	IBM (188 paires)	10	91.5%
Jin *et al.* [JTOT12]	FVC2002-DB2	6.94%	0% mais dépendante du seuil
Ahmed *et al.* [AHW11]	FVC2002-DB2	6	0% mais dépendante du seuil
Ahmed *et al.* [AHW11]	FVC2002-DB3	27	0% mais dépendante du seuil
Wang *et al.* [WH12]	FVC2002-DB2	5	0% mais dépendante du seuil
Wang *et al.* [WH12]	FVC2002-DB3	7.5	0% mais dépendante du seuil

TABLE 3.11: Comparaison entre différents schémas de protection des minuties lorsque tous les paramètres sont connus

Par ailleurs, la construction de notre schéma révocable s'est consolidée par une argumentation des différents choix entrepris en faisant de multiples études comparatives. Finalement, nous avons confronté ce schéma aux métriques d'évaluation proposées dans le chapitre 2. Essentiellement, le système peut être sensible aux deux attaques suivantes :

– **L'attaque par vol de clé** : Si le *FAR* n'est pas suffisamment bas, l'attaquant peut chercher une empreinte vraisemblable (pas nécessairement la bonne) susceptible de passer la borne de vérification. De plus, s'il est en possession du modèle de référence et du seuil de décision, l'attaquant peut vouloir trouver cette empreinte de façon *Offline* sans accéder au serveur d'authentification. Ainsi, cette attaque devient beaucoup plus critique que l'attaque *en ligne* qui peut être évitée en bloquant le système après un nombre alarmant d'échecs consécutifs à l'authentification.

– **L'inversion partielle à partir des références multiples lorsque la matrice de projection devient pleine** : Même si nous obtenons une bonne diversité, l'attaque

par références multiples doit être considérée. Dans l'absolu, même si elle est difficile, cette attaque peut être effective. Elle est possible à cause du caractère contigu du processus de binarisation dans le BioHashing. En effet, soit $W = RX$ le vecteur obtenu après projection aléatoire. W est ensuite binarisé par seuillage pour obtenir le biocode. Il est certain qu'une binarisation multi-échelle, comme proposé par Nagar et Jain [NJ09] renforcerait d'avantage l'irréversibilité de la méthode. En contrepartie, cela réduirait les performances.

Comme nous pouvons le constater, les attaques sont effectives lorsque la clé utilisateur ainsi qu'un ou plusieurs modèles de références appartenant au même individu sont connus au même temps. Ces attaques sont généralement récurrentes lorsqu'il s'agit de transformations révocables [LH13]. La gestion de la clé utilisateur est critique. Celle-ci doit impérativement être stockée séparément, du modèle de référence pour éviter qu'ils soient compromis en même temps. Si, de plus, nous assurons la confidentialité du modèle de référence (après transformation), ces menaces peuvent être complètement évitées. Cela est l'objet de notre prochaine étude. Le chapitre suivant présente une méthode décentralisée pour la gestion des créances utilisateur dans le cas d'un système révocable d'empreintes digitales.

Chapitre 4

Gestion d'identité biométrique décentralisée

Ce chapitre s'intéresse principalement à l'étude d'architectures décentralisées pour la gestion des données biométriques. Il se scinde en deux parties. Dans la première partie, nous présentons une solution d'un système MatchOnCard avec biométrie révocable, compatible avec la norme PKCS15, dans le but de proposer des solutions de cartes d'identité biométriques. Dans la seconde partie, nous nous intéressons à l'application de masse des passeports biométriques. Nous étudions principalement les menaces d'invasion de vie privée dans le traitement des données sensibles.

Sommaire

4.1 Introduction

Récemment, dans le but de garantir la sécurité des applications et des données, différentes architectures basées sur des éléments sécurisés sont apparues.
D'après Madlmayr *et al.* [MDLS07], un élément sécurisé est défini comme :

Un environnement dynamique où plusieurs applications peuvent être chargées, personnalisées, gérées et supprimées indépendamment l'une de l'autre.

Un élément sécurisé est un ensemble de composants : un microprocesseur, des mémoires, un système d'exploitation et des applications.
Les éléments de sécurité de ces dispositifs sont implémentés aussi bien au niveau matériel que logiciel. Les microprocesseurs sont, par exemple, dotés de mécanismes pour lutter contre les attaques invasives. Le système d'exploitation est, quant à lui, capable de garantir efficacement le contrôle d'accès aux données sensibles.
Certains dispositifs mobiles comme les cartes à puce sont considérés comme éléments sécurisés. Ils stockent des données et des applications hautement sensibles comme les applications de payement. Les langages pour les programmer sont apparus vers le milieu des années 90. Ces technologies sont connues autant que MULTOS, ZeitControl, SmartCard .Net, et la JavaCard qui constitue la technologie la plus répondue. JavaCard est l'adaptation du langage Java prenant en compte les contraintes matérielles des cartes à puce.

Dans un système biométrique typique, les modèles de référence sont stockés dans des bases de données centrales. Avec ce stockage centralisé, de possibles menaces de sécurité se posent. En même temps, pour des raisons sociales, culturelles ou autres, la perception d'un tel système peut négativement influer sur son acceptation. Une solution possible parmi toutes les autres que nous avons présentées dans le chapitre 1, est la gestion décentralisée du système biométrique à travers de multiples cartes à puce.
Dans le chapitre 1, nous avons vu que dans un système SoC (Secure on Card), la carte est uniquement utilisée comme un dispositif de stockage. Le passeport électronique est l'exemple concret des systèmes SoC. Nous nous intéressons, *à fortiori*, à cette application de masse. A la fin de ce chapitre, nous présentons une étude sur la sécurité et l'invasion de la vie privée impliquées dans les passeports biométriques.

Un système MoC (Match On Card), comme le montre le figure 4.1, est une extension du système de stockage (SoC) vers un système où de plus le processus d'appariement est effectué sur la carte. Parallèlement, un point faible de la biométrie révocable est lié à la gestion de la clé utilisateur. Si cette clé est compromise au même temps que le modèle de référence, des failles de sécurité peuvent exister. L'objectif principal de ce chapitre est d'étudier comment les deux technologies de biométrie révocable et de JavaCard peuvent être complémentaires et contribuer à résoudre les problèmes de sécurité et de confidentialité concernant la biométrie.

Notre objectif est d'embarquer aussi bien les données personnelles que l'application de vérification biométrique sur la carte. Il s'agit de générer un MoC révocable sur JavaCard. Le système hybride résultant a les propriétés suivantes : il confère à la biométrie les caractéristiques de révocabilité et de diversité souhaitées, fournit à la carte à puce un lien fort à l'utilisateur par biométrie et garantit à l'utilisateur le respect de sa vie privée et une manipulation sereine de ses données personnelles.
Dans la section 4.2 de ce chapitre, nous explicitons plus en détails nos principales motivations. Dans les sections 4.4, 4.3 et 4.5, nous présentons la conception de la solution MoC. La section 4.6 est dédiée à l'étude des passeports biométriques. La section 4.7 conclut ce chapitre.

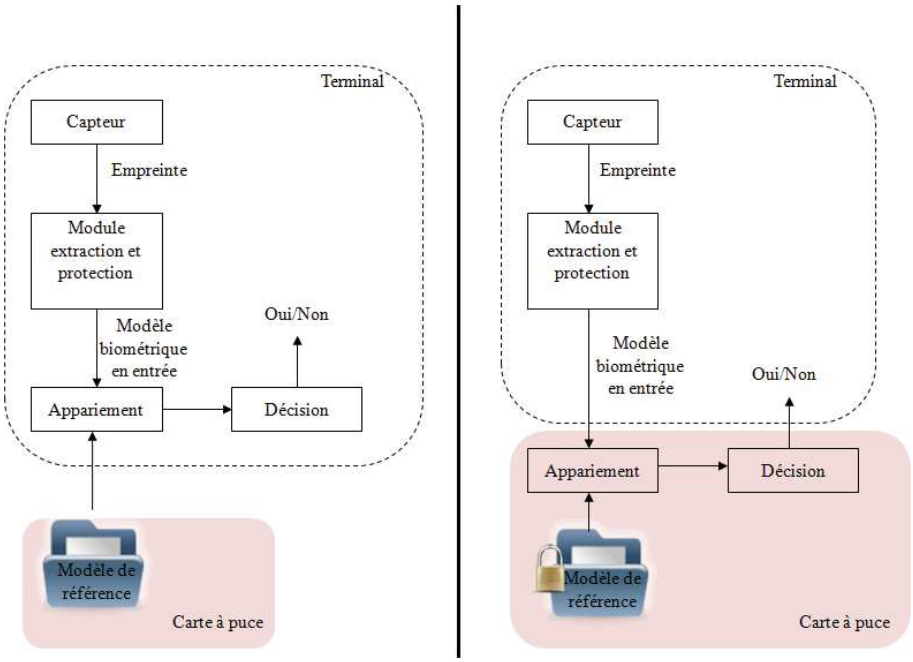


FIGURE 4.1 – Différence entre un système SoC (à gauche) et un système MoC (à droite)

4.2 Vers des cartes d'identité personnelles : Idées de base

La recherche concernant les systèmes MoC est principalement accaparée par les grands leaders de la sécurité numérique comme Sagem ou Gemalto. Pour cette raison, nous trouvons peu de publications académiques. Nous citons, à titre d'exemple, les quelques travaux mentionnés dans [SS02, GB07, SB08].

Notre intérêt pour les MoCs constitue un enjeu de taille. Il est généralement assez difficile de concevoir un tel système à cause des contraintes matérielles réduites. Néanmoins, la sécurité offerte peut être inviolable. Ainsi, la proposition d'une architecture MoC révocable est basée sur certaines idées clés énumérées comme suit :

1. L'évaluation du schéma révocable (par descripteurs locaux), dans le chapitre précédent, a souligné la propriété de diversité du système. Cependant, si les modèles transformés d'un même individu sont révélés au même temps que leurs clés, le système peut être vulnérable. Une solution consiste à assurer la confidentialité du modèle transformé par un système MoC. Par conséquent, il ne sera jamais exposé à l'extérieur de la carte. Cela est de plus motivé par la comparaison des biocodes qui se base sur une simple distance de Hamming. Ainsi, une JavaCard raisonnablement peu chère peut être suffisante pour faire les calculs.
2. Une authentification conjointe à 3 facteurs, comme le montre la figure 4.2, offre une meilleure gestion du scénario de vol de clé. Si la clé est volée, elle sera inutile sans la carte correspondante. Si les deux créances sont volées au même temps, une gestion plus rigoureuse du processus de post-personnalisation de la carte est faite, principalement, en raison de la propriété de révocabilité du système biométrique. En révoquant les cartes perdues, il est maintenant possible de changer la biométrie de référence pour éviter tout abus de ces cartes.

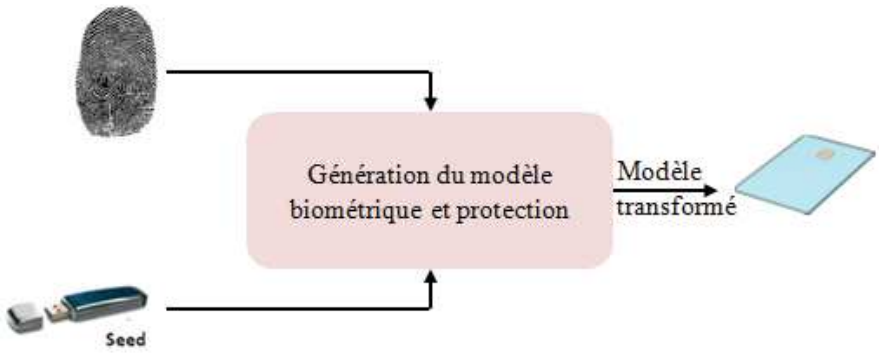


FIGURE 4.2 – Principe du système d'authentification à 3 facteurs

3. Nous proposons une gestion des données d'authentification à l'intérieur de la carte conforme à la norme **PKCS15** [PKC00b]. Les spécifications PKCS15, sont largement utilisées par l'industrie, spécialement pour les cartes d'identité électroniques. Son adoption permet donc une interopérabilité avec les solutions d'identités existantes, une facilité d'intégration par rapport aux autres normes comme le standard PKCS11 [PKC00a] et enfin une meilleure gestion des données personnelles à l'intérieur de la carte tout le long de son cycle de vie.

Nous proposons donc une solution d'authentification biométrique révocable qui soit mobile (sans intervention du serveur d'authentification), interopérable et embarquable sur des cartes multi-applicatives.

4.3 Architecture globale de la solution MoC

La solution MoC est un **ensemble de composants** qui interagissent de différentes manières suivant les **différentes phases du processus**.
Comme le montre la figure 4.3, les différents composants du système sont principalement :

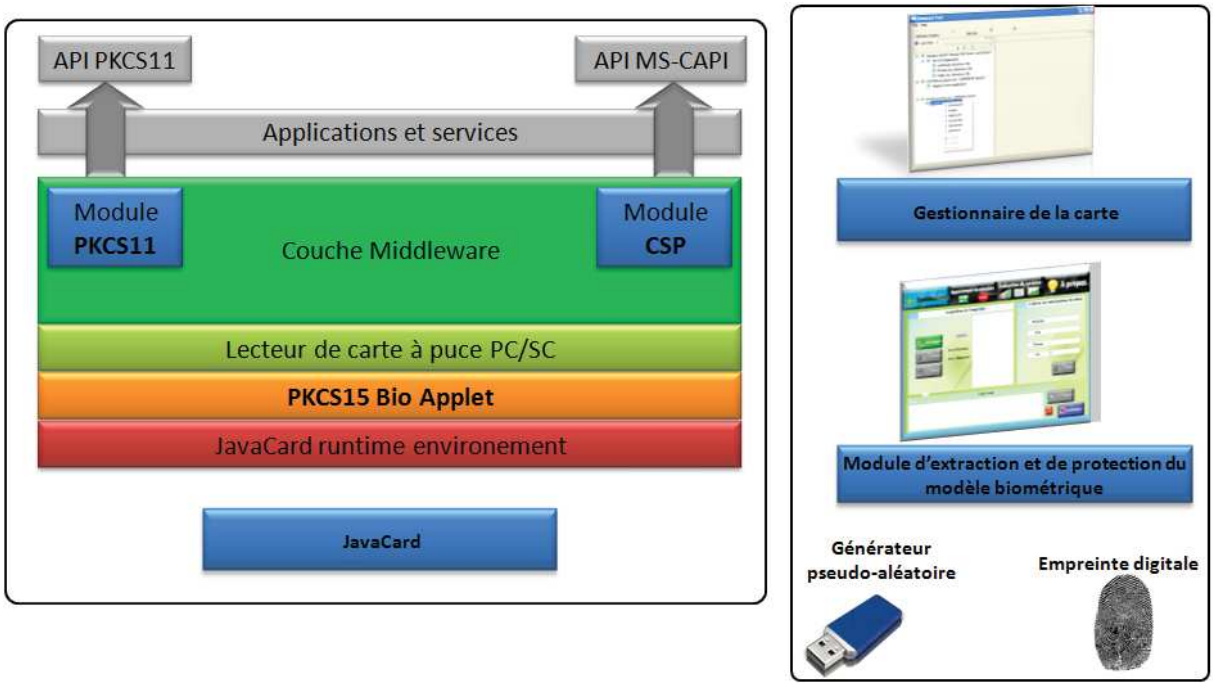


FIGURE 4.3 – Composants principaux de la solution MoC

– *L'application On Card* : il s'agit de l'applet PKCS15, nommée **PKCS15 Bio Applet**, qui stocke les données biométriques et implémente le module de comparaison sur la carte. Celui-ci retourne une réponse de type oui/non.

– *Le gestionnaire de la carte à puce* : il s'agit du système d'administration qui accomplit des fonctions de gestion de la carte le long de son cycle de vie, principalement : l'interaction avec les sources de données pour la préparation des informations à stocker sur la carte, la personnalisation et la post-personnalisation (comme le blocage, l'expiration, la désactivation ou la révocabilité de la carte).

– *Le module d'extraction et de protection du modèle biométrique* : il s'agit du module de génération du modèle biométrique transformé à partir de l'empreinte digitale de l'utilisateur.

La chaine de traitement comprend principalement :
– La phase d'enrôlement.
– La phase d'initialisation et de personnalisation de la carte.
– La phase d'usage (i.e. la vérification biométrique).
– La phase de post-personnalisation.

Elle implique les principales entités que nous synthétisons par : le client, la station d'enrôlement, la station d'émission et les environnements d'acceptation de la carte (terminaux).

La figure 4.4 illustre les interactions entre les différents composants/entités du système :

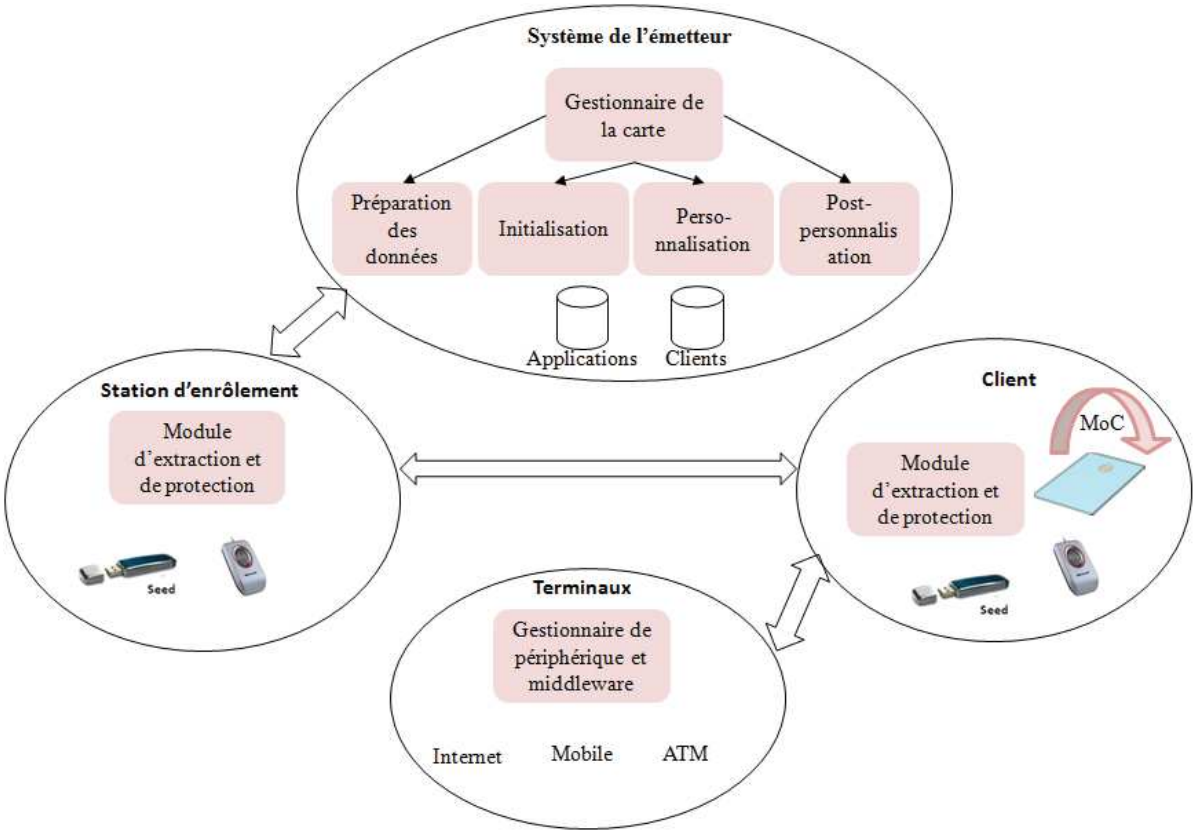


FIGURE 4.4 – Interaction entre entités et composants de la solution MoC

– *La station d'émission* qui, grâce au gestionnaire de la carte, peut initialiser les cartes en chargeant l'application PKCS15 Bio Applet (cette application est fournie par une entité tierce responsable du développement), elle peut aussi les personnaliser à partir

des données préparées par le gestionnaire et temporairement stockées dans une base de données client.

– *La station d'enrôlement*, il s'agit d'un service qui collecte les données utilisateurs, biométriques entre autres, et génère grâce au module d'extraction et de protection, le modèle de référence. Il envoie ensuite les données à la station d'émission.

– *Le client* interagit avec la station d'enrôlement pour fournir son identité, récupérer sa carte et sa clé et éventuellement faire une révocation. La révocation peut aussi être gérée de façon périodique comme pour les certificats numériques, au gré d'échéanciers réguliers.

– *Les terminaux* gèrent l'accès au service après interaction avec le client pour authentification.

Avant de détailler quelques éléments de ces interactions, nous nous concentrons d'abord sur le module PKCS15 Bio Applet en présentant sa conception détaillée.

4.4 Conception de l'applet PKCS15 Bio

Nous détaillons dans cette section la gestion interne de la carte. A cet effet, nous proposons une gestion conforme à la norme PKCS15. PKCS15 est une norme définissant les structures de données pour stocker des informations relatives à la sécurité des périphériques cryptographiques comme les HSM ou les cartes à puce. Ces dispositifs sont capables de stocker des informations d'authentification comme les certificats, ainsi que des clés cryptographiques et d'effectuer des opérations de génération de clé, de chiffrement ou déchiffrement. Des standards comme le PKCS15 sont mis au point pour assurer la structuration de ces données à l'intérieur de la carte. Il permet aux utilisateurs de s'identifier auprès des applications, indépendamment de l'implémentation sur le support cryptographique. Il est donc neutre de plateforme, neutre d'application, conforme aux autres normes comme ISO7816 ou PKCS11, et à structure modulaire flexible.
PKCS15 gagne un intérêt croissant surtout dans les applications des cartes d'identité électroniques (eID). Parmi les pays qui adoptent des eIDs au format PKCS15, nous comptons : Autriche, Belgique, Estonie, Finlande, Italie, Malte, Slovénie, Espagne, Suède et Allemagne. Cela constitue une preuve suffisante sur son utilité.

4.4.1 Modèle objet

La conception PKCS15 consiste en une approche orientée objet qui définit la structure dont les certificats, les clés de chiffrement, les objets d'authentification (PIN et biométrie) sont stockés sur la carte.
La figure 4.5 illustre la hiérarchie des objets définis par PKCS15.

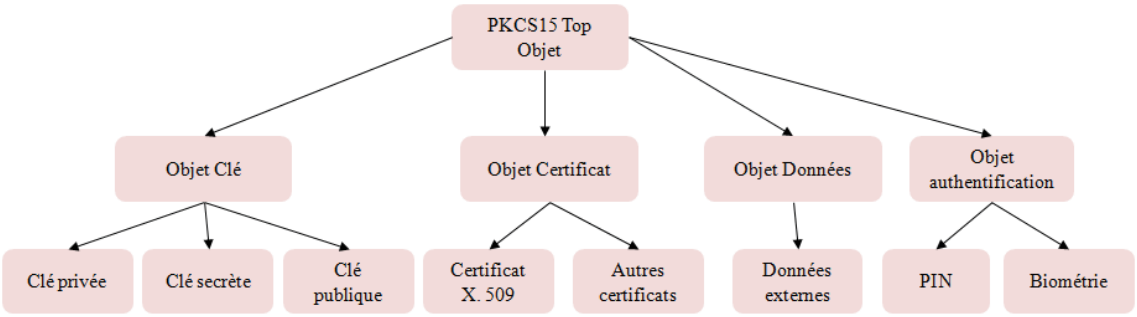


FIGURE 4.5 – Modèle objet PKCS15

Il existe quatre classes d'objets : clés, certificats, objets d'authentification et objets de donnée. Ces classes sont décomposées à leur tour en sous-classes, par exemple, la classe clés peut être divisée en trois classes : clés publiques, clés privées et clés secrètes. Tout objet instancié comporte un certain nombre d'attributs hérités depuis sa classe mère.
Un objet peut être privé (protégé contre les accès non autorisés) ou bien public. Tout type d'accès aux objets privés est défini par les objets d'authentification (PIN, Biométrie, etc.), où il est nécessaire d'authentifier le porteur de la carte avant d'effectuer n'importe quelle opération. Tandis que les objets publics sont uniquement protégés contre les modifications et non pas contre les accès en lecture.

A chaque objet de cette hiérarchie, est associé un fichier dont le but est de répertorier l'identité des objets disponibles. La figure 4.6 illustre une hiérarchie de ces fichiers. Voir l'annexe B pour une description des fichiers PKCS15.

Sur une plateforme JavaCard, la conception d'une applet PKCS15 est basée sur la construction des modules suivants :

- Le module de gestion de la mémoire, idéalement avec une option de ramasse-miettes.
- Le module de gestion des fichiers PKCS15.
- Le module de gestion des objets PKCS15.
- Le module de gestion de la sécurité.

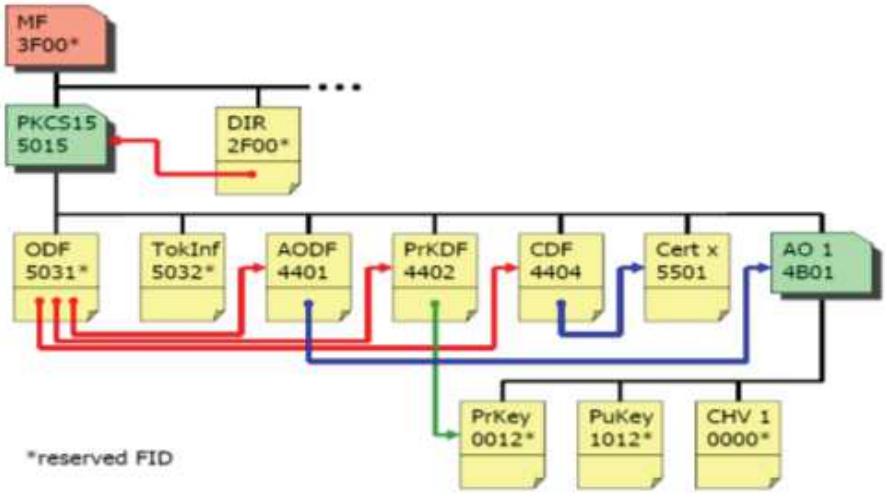


FIGURE 4.6 – Hiérarchie des fichiers PKCS15

La sécurité étant l'un des points critiques de l'applet, nous proposons de la gérer sur trois niveaux :

1. L'utilisation d'une *LCA* (Liste de Contrôle d'Accès) pour restreindre les permissions d'accès à un objet ou un fichier.
2. L'utilisation des objets d'authentification comme les PINs, la biométrie ou les clés de chiffrement.
3. L'interaction avec le monde extérieur à travers un nombre prédéfini de commandes.

Nous modélisons maintenant l'applet PKCS15 par le diagramme de classes défini sur la figure 4.7. La tableau 4.1 donne une description générale de ces classes.

La conception détaillée de ces classes (comme la gestion de la mémoire) n'est pas l'objet de ce chapitre. Nous nous contentons d'en donner quelques éléments dans l'annexe B. Il est clair que l'implémentation d'une applet PKCS15 est un travail assez conséquent, qui est plus rattaché à l'industrie qu'à la recherche. Néanmoins, une applet PKCS15 biométrique révocable est inexistante commercialement parlant et fait encore partie du monde de la recherche. Ainsi, dans ce qui suit, nous présentons uniquement les détails de conception concernant la partie biométrique.

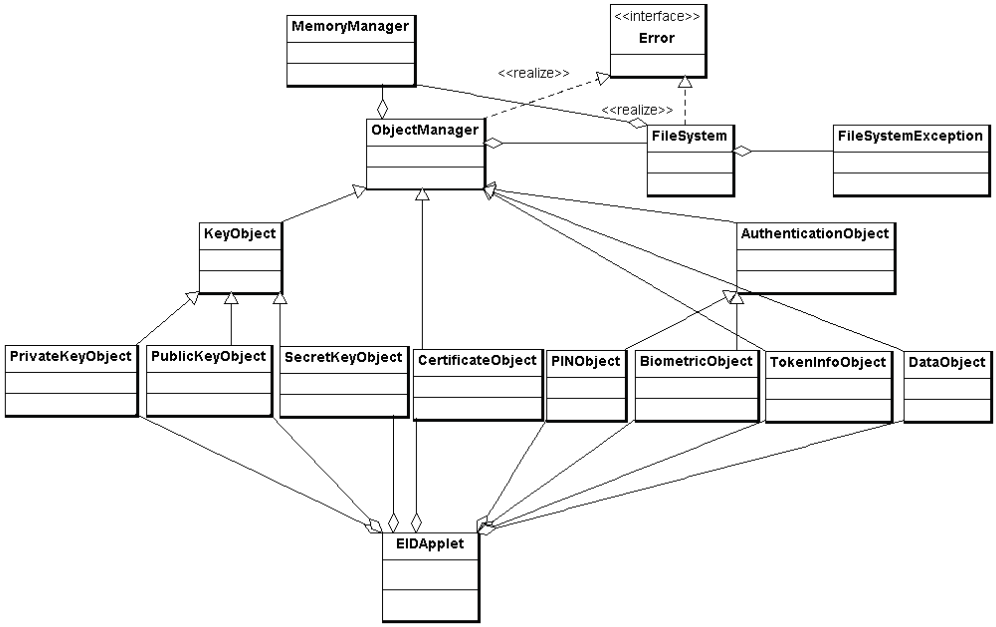


FIGURE 4.7 – Diagramme de classes de l'applet PKCS15

Nous utilisons dans cette applet, 16 identités par lesquelles il est possible au client de s'authentifier :

- Les identités de 0 à 3 : pour l'authentification par code PIN.
- Les identités de 4 à 12 : pour les protocoles à clé publique de type Défi/Réponse.
- Les identités 13 et 14 : pour l'authentification par biométrie.
- L'identité 15 est réservée.

Le modèle biométrique (version transformée) est sécurisé par des restrictions d'accès. Il est considéré comme un objet privé, il ne peut donc jamais être lu. Dans le cas d'une révocation, il peut être supprimé et modifié uniquement par un administrateur. Relativement à la figure 4.4, il s'agira de l'autorité d'émission de la carte. L'administrateur est associé à l'identité 4 et il est authentifié par la carte suivant le protocole RSA Défi/Réponse.

Le modèle biométrique est considéré comme un objet d'authentification (classe AuthenticationObject) et sera reconnu à l'intérieur de la carte par son identité stockée dans le fichier PKCS15 associé, à savoir, le fichier AODF (Authentication Object Data File). Il est caractérisé par un descripteur qui va constituer la classe BiometricObject.
Comme il est montré sur la figure 4.8, un descripteur biométrique comprend un ensemble de champs communs à tous les types d'objets et un ensemble de champs spécifiques à l'objet biométrie.

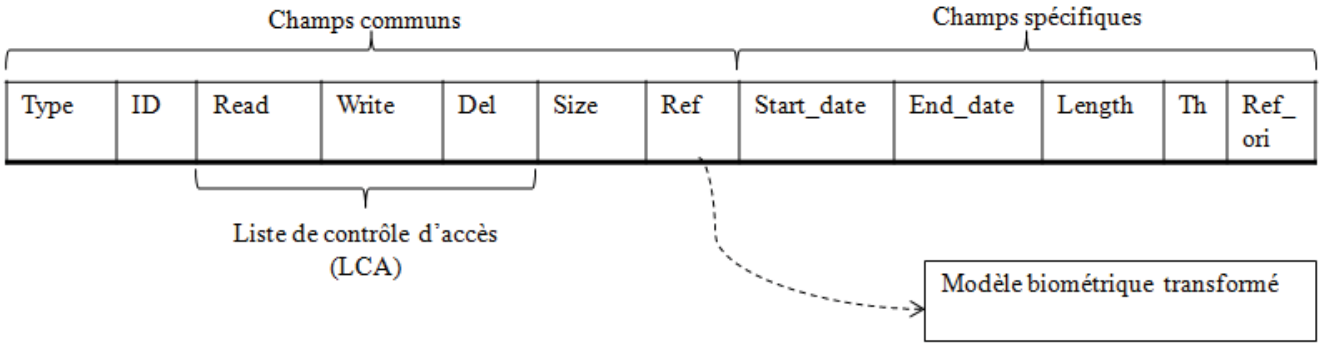
Classes	**Descriptions**		
MemoryManager	Gestion de la mémoire de l'applet		
FileSystem	Gestion des fichiers PKCS15		
ObjectManager	Classe abstraite. Gestion des objets PKCS15		
	Sous-classes	**Descriptions**	
	CertificateObject AuthenticationObject	Gérer les certificats X.509 Gérer les objets d'authentification	
		Spécialisation	**Description**
		PINObject BiometricObject	Gérer les objets PIN Gérer les objets biométriques
	DataObject	Gérer les objets des données	
	TokenInfo	Contient des informations sur la carte et l'applet	
	KeyObject	Classe abstraite. Gérer les objets clé	
		Spécialisation	**Description**
		PrivateKeyObject PublicKeyObject SecretKeyObject	Gérer les clés privées Gérer les clés publiques Gérer les clés secrètes
FileException	Génère une exception s'il y a une erreur dans la classe FileSystem		
Error	Permet aux autres classes de retourner des exceptions en cas d'erreur		
EIDApplet	La classe qui reçoit et exécute les commandes envoyées à la carte		

TABLE 4.1: Description des classes de l'applet PKCS15

Les opérations associées à un objet sont : Read, Write et Delete. Concernant l'objet biométrique, la valeur du Read est à NEVER et les valeurs de Write et Delete sont à 4 pour les associer à l'identité de l'administrateur. Le champ Ref_ori relatif à l'orientation de référence de l'empreinte doit être lu au moment de la vérification par le module externe *d'extraction et de protection* (i.e. pour savoir quel type d'enregistrement doit-il faire : basé sur l'orientation de référence ou sur les MinuCodes auto-alignés). Nous pouvons permettre cela grâce à une commande spéciale de type *GetRefOrientation()* que la classe principale *EIDApplet* pourrait traiter.
Dans le prochain paragraphe, nous explicitons plus en détail cette interface carte-terminal qui fait aussi partie de la sécurité de la carte.

4.4.2 Interface carte-terminal

L'interface de communication entre une carte et un terminal se base sur un échange de commandes appelées APDU (Application Protocol Data Unit). Les échanges, comme le montre la figure 4.9, entre la carte et le terminal se font selon le modèle client/serveur.



Champ	Description
TYPE	Objet d'authentification (PIN, Biométrie), Objet données, Objet certificats, etc.
ID	Un numéro unique qui identifie l'objet (entre 0 et 15)
LCA	Gestion des permissions d'accès à l'objet
Size	Taille totale de l'objet
Ref	L'adresse mémoire des données effectives contenues dans l'objet
Start_date et End_date	L'objet est associé à une date d'activation et d'expiration pour permettre sa mise à jour
Length	Taille des données de l'objet (le modèle biométrique)
Th	Le seuil de décision du module de vérification
Ref_ori	L'orientation de référence si elle existe, -1 sinon

FIGURE 4.8 – Descripteur de l'objet biométrie de l'applet PKCS15

Le processeur de la carte reste inactif tant qu'il n'a pas reçu une commande APDU du terminal (C-APDU) et répond par un APDU de réponse (R-APDU).

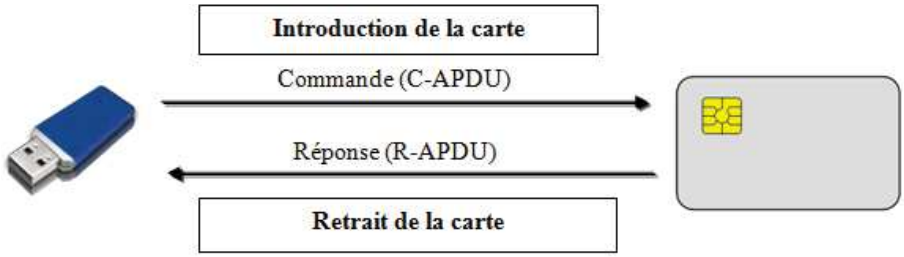


FIGURE 4.9 – Communication carte/Terminal

Nous définissons pour notre applet PKCS15 Bio, un ensemble de commandes APDU représentées par le tableau 4.2.

Les commandes relatives à l'objet biométrique sont : CreateTemplate, WriteTemplate, VerifyTemlate, DeleteTemplate et GetReferenceOrienation.

Commandes	**APDU autorisés**			
Commande de création	CreatePKCS15			
Commandes reliées à l'objet PIN	CreatePIN	ChangePIN	VerifyPIN	UnblockPIN
Commandes reliées à l'objet Biométrie	CreateTemplate	WriteTemplate	DeleteTemplate	Verifytemplate
	GetReferenceOrientation			
Commandes reliées à l'objet clé	GenKeyPair	DelKeyPair	EncDATA	DecDATA
	ImportKey	ExportKey	External Authenticate	Internal Authenticate
Commandes reliées aux données	CreateData Object	WriteData Object	GetData Object	DeleteData Object
Autres commandes	...			

TABLE 4.2: Ensemble des commandes APDU

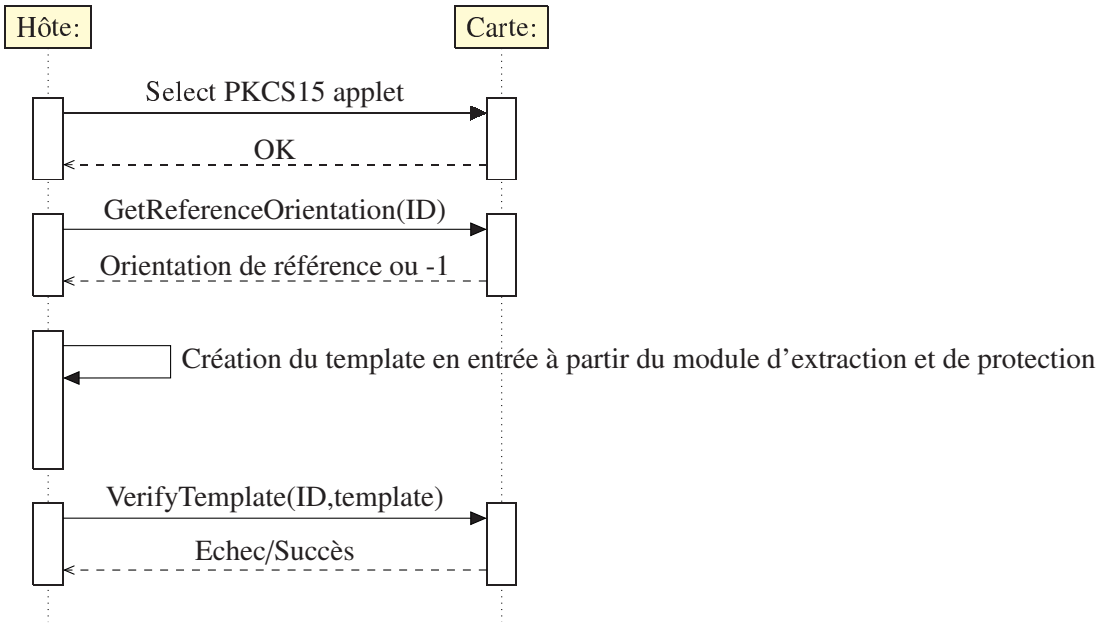


FIGURE 4.10 – Principales commandes échangées lors de l'authentification

La JavaCard est une architecture ouverte qui supporte le chargement dynamique de code. Lors de l'initialisation de la carte, l'émetteur va charger, sous forme de *bytecode*, l'applet PKCS15 Bio sur la carte. Une fois chargée, l'applet peut être sélectionnée et exécutée. Suivant la phase de traitement : personnalisation, authentification ou post-personnalisation (révocation), les interactions entre la carte et la machine hôte sont respectivement représentées par les figures 4.10, 4.11, et 4.12.

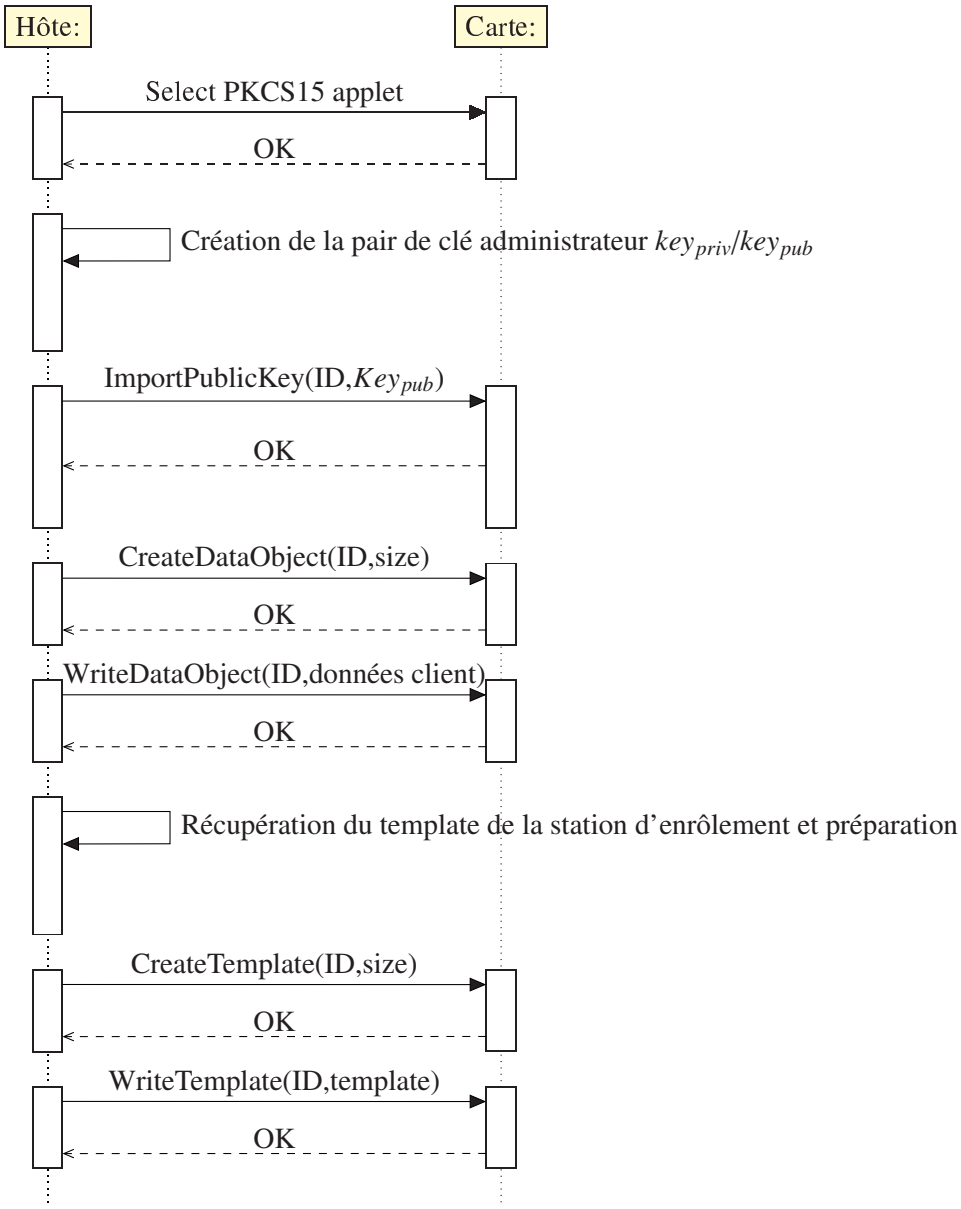


FIGURE 4.11 – Principales commandes échangées lors de la personnalisation de la carte

Lors d'une opération de personnalisation ou de post-personnalisation, la machine hôte se trouve au niveau de la station d'émission. Lors de l'utilisation de la carte en mode authentification, la station hôte est au niveau du terminal d'acceptation. Pour plus de sécurité, un canal sécurisé peut éventuellement être établi entre la carte et le terminal.
La phase d'authentification nécessite une commande de type *VerifyTemplate()* pour accomplir la comparaison à l'intérieur de la carte. Cette commande est exécutée par la classe principale de l'applet qui fait intervenir le module de vérification que nous présentons dans le prochain paragraphe.

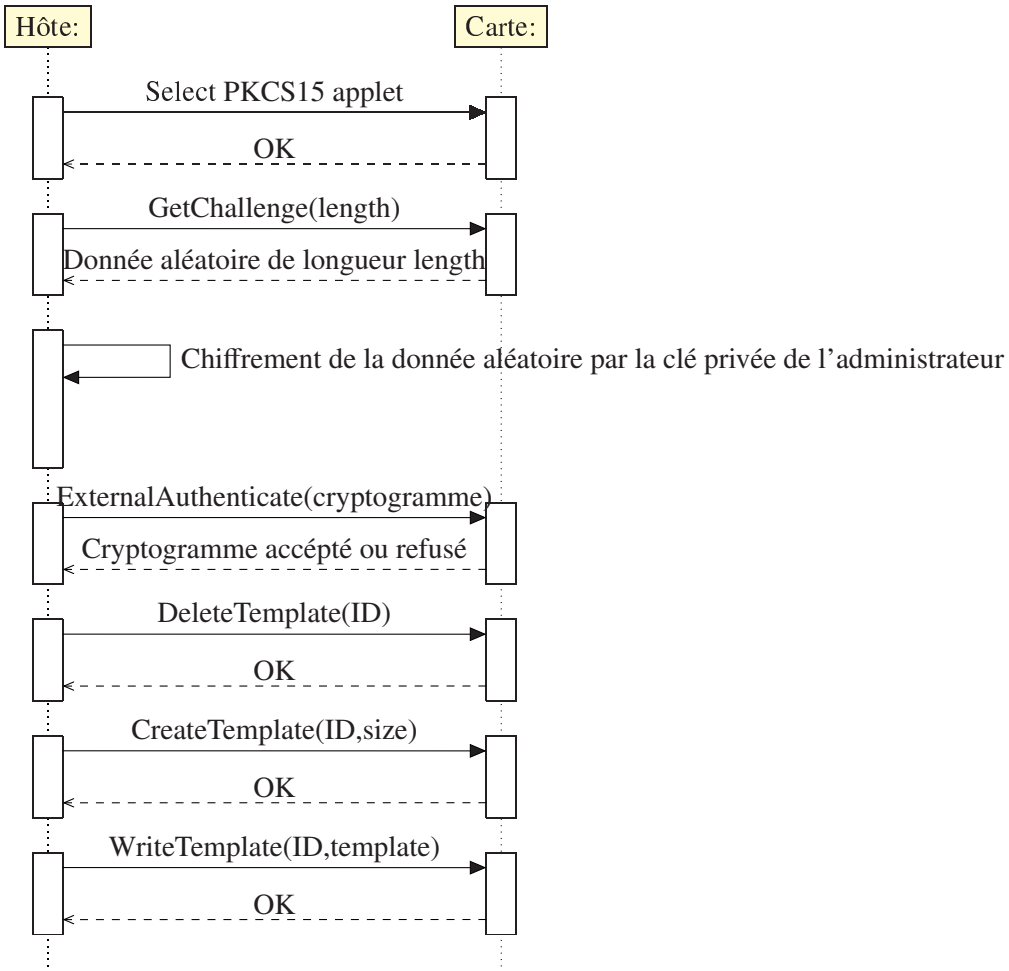


FIGURE 4.12 – Principales commandes échangées lors de la révocation du modèle biométrique

4.4.3 Comparaison biométrique sur carte

Une carte à puce est un objet fortement contraint. Aujourd'hui sur une JavaCard, la configuration matérielle peut être résumée comme suit : un processeur de faible puissance (généralement de 4 à 200 Mhz) et des mémoires de petites tailles (entre 1 et 4KO pour la RAM, 32 à 128 KO pour la ROM et 16 à 72 KO pour la mémoire de stockage ou l'EEPROM).

Notre modèle de référence occupe un espace mémoire assez limité, estimé à 500 octets. Sur les JavaCards d'aujourd'hui, cela est largement suffisant.
Contrairement à beaucoup de schémas de protection, notre algorithme de vérification est bien adapté aux architectures limitées pour les raisons suivantes :

– La slélectin des minuties diminue le nombre de points à traiter. Le nombre maximum

de minuties extraites est par exemple estimé à 20 sur la FVC2002-DB2
– Chaque minutie est représentée par un descripteur de 180 bits et la comparaison entre ces descripteurs se fait par une simple distance de Hamming.

Nous utilisons une JavaCard de type JCOP3.1 qui implémente l'API Javacard V2.2.1. Cette puce possède une EEPROM à 36KO, une ROM à 96KO et une RAM à 2304 octets. La fréquence de son horloge interne est de 30 Mhz. Cette puce possède un crypto-processeur qui implémente l'algorithme de chiffrement RSA avec des clés de 2048 bits.

Nous avons implémenté sur la carte, le calcul de la distance de Hamming entre deux biocodes à 180 bits. Le temps de calcul est de 70 ms. Les temps d'exécution restants sont ensuite estimés. Dans le tableau 4.3, nous résumons ces temps d'exécution. Nous considérons deux versions de l'algorithme : la variante par recherche gourmande sans prise en compte du voisinage et la variante par Best First Search (BFS) avec prise en compte du voisinage. Les résultats montrent que la recherche par BFS détrône de loin la recherche gourmande. Néanmoins, des efforts restent à faire pour améliorer les temps de traitement. L'une de nos perspectives immédiate est de proposer un code plus optimisé pour diminuer le temps de traitement à l'intérieur de la carte.

	Appariement par recherche gourmande	Appariement par BFS
Complexité	$O(N^4)$	$O(N^2)$
Nombre d'opérations maximum	$N^2 \times DistanceHamming + (11 \times N^4 + 4 \times N^2 + 3)$ opérations élémentaires	$N^2 \times DistanceHamming + (10 \times N^2 + 3)$ opérations élémentaires
Temps de calcul	7 mn	30 s

TABLE 4.3: Estimation du temps de calcul du module de vérification sut la carte

4.4.4 Analyse et discussion

En se basant sur les critères d'évaluation préalablement définis, l'analyse du système MoC est résumée dans le tableau 4.4 où nous remarquons une amélioration des critères A_{10} et A_{11} par rapport au système sans carte à puce.

La proposition d'une solution d'authentification révocable sur carte à puce élimine ainsi tous les risques d'inversibilité du modèle biométrique de référence. Elle offre, de plus, une meilleure gestion du scénario de vol de clé tout en gardant la propriété de diversité de la biométrie. La carte est gérée en conformité avec la norme PKCS15. Il s'agit donc d'une carte d'identité électronique qui supporte l'authentification par biométrie

A1	A2	A3	A4	A5	A6	A7	A8	A9
6.68%	1	0	0%	7.16%	0%	true	18432 bits	NI
A10	A11	A12	A13	A14	A15	A16	A17	
NI	ONF	ONF	0%	0%	0%	0	180 bits	

TABLE 4.4: Evaluation du système MoC

révocable. La faiblesse de cette solution est rattachée au temps d'exécution qui reste assez conséquent pour certaines applications comme le payement par carte. Il est donc souhaitable d'améliorer ce temps. C'est l'une de nos principales perspectives.

4.5 Cas d'utilisation de la carte d'identité proposée

Cette carte d'identité peut être facilement intégrée dans plusieurs contextes comme pour sécuriser d'une manière simple les accès web. L'accès à un service web met en jeux plusieurs processus s'exécutant sur des machines distantes. Il s'agit d'applications structurées en plusieurs niveaux :

- Niveau interfaçage : postes clients accédant à des ressources distantes.
- Niveau serveur de données : implémentation de la logique métier rendue consommable par l'utilisation de standards comme TCP/IP ou HTTP.
- Niveau bases de données : intégration de l'infrastructure informationnelle.

Dans de telles applications, l'authentification est devenue un requis prioritaire. Le but est de permettre à la personne authentifiée de faire les opérations autorisées comme : l'accès à une page web, l'appel à une méthode ou la manipulation d'un objet du domaine.
La vision d'une solution à ce problème réside dans la sécurisation de la communication TCP/IP à travers le protocole SSL/TLS. Il s'agit d'un protocole ayant pour but de créer un canal de communication authentifié. Il est ensuite possible pour le serveur d'authentifier le client en lui demandant de présenter son certificat numérique et de signer les messages échangés.
SSL/TLS utilise une authentification à infrastructure publique où le client doit être en possession d'une clé privée et d'un certificat numérique. Pour garantir la protection de ses créances, nous proposons d'utiliser notre applet PKCS15 Bio comme fournisseur des services cryptographiques (i.e. *FSC*).
Elle offrira des services de chiffrement, de signature, de génération aléatoire et des capacités de stockage d'objets sensibles (certificats numériques, clés privées, données biométriques, etc.). L'invocation de ce *FSC* ou son chargement dynamique est supportée par l'utilisation d'API standards comme MS CAPI ou PKCS11, implémentées sous forme de middelwares

comme nous l'avons préalablement illustré sur la figure 4.3. L'utilisation d'objets comme les certificats ou les clés privées peut être contrôlée par l'authentification de l'utilisateur en utilisant sa biométrie, comme le montre la figure 4.13, au lieu du traditionnel code PIN. Cela renforce la sécurité dans le cas du vol de la carte ou de son partage. L'idée est donc

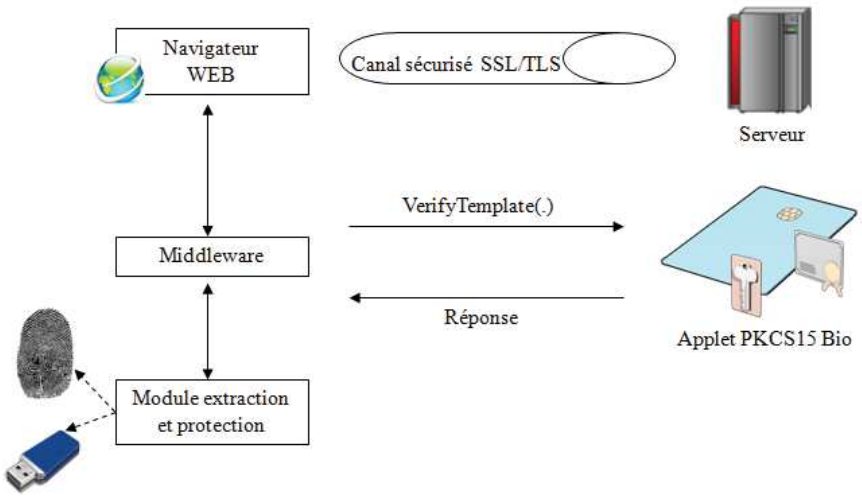


FIGURE 4.13 – Utilisation de la carte d'identité proposée dans le contexte du web sécurisé

de simplifier le rôle du client en lui proposant plusieurs services basés sur un seul objet sécurisé, comme :

- La consultation de son compte bancaire.
- L'accès à des services gouvernementaux en ligne : vote, demandes de fiches d'états civils, etc.
- L'accès à des services de santé en ligne : consultation du dossier médical, etc.

4.6 Sécurité et vie privée dans les passeports biométriques

L'idée des passeports biométriques est récente. Le contrôle des frontières aux Etats-Unis les adopte en 2006. Depuis, le nombre des pays qui délivrent de tels passeports est en perpétuelle augmentation. Le nombre de passeports biométriques dans le monde est estimé à 100 millions. Il s'agit, sans doute, de l'application biométrique de masse la plus répandue. Pour empêcher l'utilisation frauduleuse des passeports, l'ICAO (International Civil Aviation Organization) prévoit en plus de la photo, l'intégration des empreintes digitales sur le support de stockage. Pour ces raisons, nous étudions dans cette section les risques d'atteinte à la vie privée et au droit de protection des données sensibles, engendrés par l'utilisation des passeports biométriques.

4.6.1 Manipulation des données personnelles

Le passeport électronique est un document de voyage qui contient une puce. Celle-ci stocke les informations personnelles et la donnée biométrique de son porteur. Une technologie sans contact (ou RFID) a été choisie lors du processus d'authentification pour la lecture des données.
Conformément aux premières spécifications d'ICAO [Org04] en 2004, le visage numérique est obligatoire. Les autres modalités comme l'empreinte ou l'iris sont optionnelles. Toutefois, certains pays comme l'Union Européenne exigent le stockage des empreintes digitales dans la puce du document.
D'après les spécifications d'ICAO, la mémoire de la puce est composée de 16 groupes de données nommés DG1,...,DG16, qui correspondent à une structure de données logique. Cette structure est divisée comme suit :

- DG1 est la copie numérique de la zone de lecture du passeport électronique (MRZ ou Machine Readable Zone).
- DG2 contient la photo du visage du porteur (format jpeg ou jpeg2000).
- DG3 contient les empreintes digitales.
- DG4 est réservé pour l'iris.
- DG14 et DG15 sont utilisés pour les clés publiques.

Pour contrôler l'intégrité de ces données, une fonction de hachage est utilisée et la valeur du résultat est stockée dans l'objet de sécurité du document (SOD : Security Object of the Document). Au même temps, l'autorité d'émission signe numériquement cette valeur de hachage et stocke aussi la signature dans cette zone de sécurité. Finalement, la puce contient des clés privées, stockées dans une région sécurisée de la mémoire.
La puce du passeport est uniquement utilisée pour le stockage. La donnée biométrique doit quitter la puce à travers une communication sans contact pour effectuer la comparaison. Ce processus peut donc être soumis à des considérations concernant la vie privée.
Entre autres, la technologie sans contact implique les attaques suivantes :

- L'attaque par écoute : l'attaquant espionne une communication légitime entre le lecteur et le passeport.
- L'attaque par substitution : l'attaquant communique avec la puce sans le consentement du porteur en utilisant un lecteur à proximité du passeport.

D'autres attaques, sur les passeports électroniques, relatives à la technologie sans fil existent. Par exemple, Hlavac et Rosa [HR07] développent une attaque par relais contre le passeport tchèque. Chotia and Smirnov présentent en 2010, un procédé de suivi du titulaire du passeport [CS10]. Richter, Mostowski et Poll montrent dans [RMP08] comment

détecter la présence d'un passeport et récupérer sa nationalité.
Les attaques par substitution ou par écoute peuvent être utilisées pour récupérer les données stockées dans la puce. Par conséquence, un protocole de contrôle d'accès est nécessaire sur le canal de communication.

4.6.2 Politique de sécurité

Le standard d'ICAO [Org04] prévoit trois protocoles cryptographiques dont seul le premier est obligatoire :

1. L'authentification passive pour vérifier l'authenticité des données à l'intérieur de la puce en se basant sur le contrôle de la signature numérique de toutes les données stockées.
2. L'authentification active pour prouver l'authenticité de la puce elle-même. La puce doit prouver la connaissance d'une clé privée délivrée par l'autorité d'émission à travers un protocole Défi/Réponse.
3. Le contrôle d'accès basique (BAC : Basic Access Control) qui est utilisé pour anticiper les accès non autorisés. Le protocole BAC produit des clés de session pour empêcher les attaques par écoute.

Les protocoles d'authentification passive et active ne concernent pas le contrôle d'accès. Nous nous intéressons donc au protocole BAC uniquement.
Dans le protocole BAC de base, le lecteur prouve à la puce qu'il connait les informations de la zone MRZ, typiquement imprimées sur la première page du passeport. Au niveau du terminal, ces informations sont lues pas un capteur de reconnaissance de caractères (un OCR). Une clé K est automatiquement dérivée à partir de ces données, en utilisant les 128 bits les plus significatifs, générés par la fonction de hachage Sha-1, comme suit :

$$K = 128msb(Sha-1(MRZ)) \text{ msb : most significant bit}$$

Ensuite, deux clés de 128 bits K_{enc} et K_{mac} sont dérivées de la clé K ($||$ est le symbole de concaténation) :

$$K_{enc} = 128msb(Sha-1(K||1)),$$

$$K_{mac} = 128msb(Sha-1(K||2)).$$

Ces deux clés sont contenues dans la mémoire de la puce et sont utilisées dans un mécanisme Défi/Réponse à base d'un chiffrement 3DES/MAC comme le démontre l'algorithme 15. Nous dénotons Enc, la fonction de chiffrement et MAC la fonction de génération de l'empreinte MAC.

Algorithme 15 : Protocole BAC de base

La puce génère un défi de 64 bits C_c et l'envoie au terminal
Le lecteur génère deux mots K_r et C_r de 64 bits et calcule avec les clés K_{enc} et K_{mac} la valeur :

$$MAC(Enc(C_r||C_c||K_r))||Enc(C_r||C_c||K_r)$$

Cette valeur est envoyée à la puce
La puce récupère le défi C_c et extrait la clé K_r. Elle génère ensuite une clé K_c de 64 bits et envoie au lecteur la valeur :

$$MAC(Enc(C_c||C_r||K_c))||Enc(C_c||C_r||K_c)$$

Le lecteur récupère le défi C_r et extrait la clé K_c
Le lecteur et la puce calculent de nouvelles clés de session avec $\grave{K} = K_r \oplus K_c$ et :

$$\grave{K}_{enc} = 128msb(Sha - 1(\grave{K}||1)).$$

$$\grave{K}_{mac} = 128msb(Sha - 1(\grave{K}||2)).$$

L'objectif du protocole BAC est de s'assurer que l'accès aux données de la puce ne peut se faire que par un lecteur qui a effectivement lu la zone MRZ. En effet, le lecteur dérive directement les clés K_{enc} et K_{mac} à partir des données MRZ. Les clés de session $\grave{K}_{enc}$ et $\grave{K}_{mac}$ sont ensuite utilisées pour établir un canal sécurisé entre le terminal et la puce.

En 2006, l'organisme allemand de normalisation BSI réalise une nouvelle spécification reportée dans [fSidI08], avec un nouveau protocole de contrôle d'accès, appelé PACE, pour remplacer le protocole BAC.
Le protocole PACE fonctionne en deux temps indépendants, de la manière suivante : premièrement, le lecteur lit, grâce à un capteur OCR, une donnée spécifique sur le passeport. Cette donnée est sous forme d'un nombre aléatoire α indépendant des données personnelles du propriétaire du passeport. Le nombre α doit avoir suffisamment d'entropie pour éviter l'attaque par force brute. En second lieu, le terminal et la puce réalisent un protocole d'échange de clés, de type Diffie-Hellman par exemple, pour dériver un secret commun. Dans la spécification, le protocole d'échange de clés est noté KA et le domaine des paramètres est noté D_c. Finalement, une clé de session est dérivée du secret commun en utilisant l'algorithme 16.

Algorithme 16 : Protocole PACE

La puce génère un défi R_c, calcule la clé $K_\alpha = Sha-1(\alpha||3)$, chiffre le défi R_c avec la clé K_α et envoie le résultat au lecteur en même temps avec les paramètres D_c
Le lecteur calcule la clé $K_\alpha = Sha-1(\alpha||3)$ et déchiffre le défi R_c
La puce et le lecteur calculent un domaine éphémère $\grave{D}$ à partir du défi R_c et des paramètres D_c
La puce et le lecteur calculent une clé commune K en utilisant le protocole d'échange de clés KA et les paramètres $\grave{D}$: premièrement, la puce et le lecteur calculent chacun des paires de clés *privée/publique* nommées repectivement $(PuK_{c,pace}, PrK_{c,pace})$ et $(PuK_{r,pace}, PrK_{r,pace})$. Ils dérivent ensuite une clé commune K comme suit :

$$K = KA(PuK_{c,pace}, PrK_{c,pace}, \grave{D}) = KA(PuK_{r,pace}, PrK_{r,pace}, \grave{D})$$

La puce et le lecteur calculent ensuite les clés de session :

$$K_{enc} = 128msb(K||1)$$

$$K_{mac} = 128msb(K||2)$$

La puce et le lecteur réalisent ensuite une authentification mutuelle. Le lecteur calcule et envoie le facteur d'authentification $T_r = MAC(K_{mac}, PuK_{c,pace})$ à la puce qui à son tour le recalcule et le vérifie
La puce calcule le facteur $T_c = MAC(K_{mac}, PuK_{r,pace})$ et l'envoie au lecteur. Le lecteur recalcule et vérifie T_c.

4.6.3 Discussion et proposition

De nombreux travaux pour analyser le protocole BAC ont été réalisés. L'une des premières analyses de sécurité a été présentée par Juels *et al.* [JMW05] où plusieurs problèmes ont été identifiés.
Les données MRZ concernent précisément la date de naissance du titulaire, le numéro de passeport et sa date d'expiration. Tout d'abord, les informations MRZ peuvent être directement lues sur le passeport par une personne tierce ou partiellement connues comme la date de naissance. De plus, l'entropie des données MRZ est très faible, spécialement, pour les pays qui génèrent le numéro de passeport séquentiellement. Des possibilités d'attaque par force brute ont été clairement effectuées sur certains passeports tels que les passeports belges ou allemands [AKQ08].
Le protocole PACE corrige les principales faiblesses du protocole BAC. La sécurité de ce mécanisme est basée sur deux points : premièrement, les données personnelles de la zone

MRZ ne sont pas utilisées et l'entropie du nombre aléatoire α n'est pas surestimée.
D'autre part, la génération des clés de session utilise un crypto-système à clé publique (et non un crypto-système symétrique où la clé secrète est écrite sur la zone MRZ).
Malheureusement, des faiblesses de cette nouvelle spécification ont été signalées par Chaabouni et Vaudenay dans [CV09]. Le nombre α est simplement imprimé sur le passeport, il peut donc être facilement lu, principalement lorsque le passeport est volé. Dans ce cas, un attaquant avec un faux lecteur peut successivement exécuter le protocole PACE. Par conséquent, un lecteur qui exécute avec succès le protocole d'authentification du terminal (même avec un certificat révoqué) a accès à toutes les données sensibles.

Les protocoles d'accès aux données sensibles, dans les passeports biométriques, ne sont pas complètement aboutis et le risque de vol de l'identité biométrique est ainsi possible. Le passeport biométrique, peut dans ce cas, comporter des risques de violation de vie privée. Avec cette étude sur les passeports électroniques, nous avons mis en évidence les risques liés à la protection du modèle biométrique, avec des protocoles cryptographiques. Dans certains cas, des risques palpables de violation de vie privée peuvent exister.
Comme solution possible à ce problème, nous proposons de stocker dans le passeport, un modèle d'empreinte digitale révocable. Au lieu de stocker une image JPEG de l'empreinte digitale (qui pourrait être volée et utilisée par un attaquant), nous proposons de stocker dans la zone DG3 de la puce, le modèle protégé à base de descripteurs locaux. En utilisant la biométrie révocable, le problème d'invasion de vie privée va être complètement résolu. Il s'agit donc d'une application possible de notre système à biométrie révocable.

4.7 Conclusion

L'idée principale de ce chapitre est de fédérer la biométrie révocable et les cartes à puce à l'égard de la protection des données biométriques.
Ce chapitre ouvre la voie à un domaine d'étude peu traité dans la littérature. Il met l'accent sur les enjeux et en même temps sur les difficultés. L'objectif a été en partie atteint. D'autres aspects pourront être explorés ultérieurement, principalement pour améliorer les temps de calcul.
En parallèle, nous avons étudié l'application de masse des passeports électroniques. Cette application n'est pas dépourvue de menaces. Il est critique de garantir, pour les différents usagers, une manipulation pérenne à l'égard du traitement de leurs données personnelles. Nous avons pu constater que la biométrie révocable pourrait s'avérer nécessaire dans ce contexte.

Chapitre 5

Conclusion générale

« Il est facile de manquer le but et difficile de l'atteindre »

Aristote

L'empreinte digitale est une technologie biométrique très attrayante, principalement, en raison de ses performances et de son acceptabilité. Malheureusement, de nombreux problèmes lui sont aussi associés. Il existe des préoccupations majeures pour des raisons d'éthique, de sécurité et d'invasion de la vie privée. L'empreinte digitale est une modalité à trace, elle n'est donc pas secrète. En cas de compromission, comme il est impossible de la révoquer, elle devient inutilisable. Les méthodes de protection de la biométrie sont des solutions prometteuses pour les problèmes de révocabilité et de confidentialité qui concernent les modèles biométriques en général. Pour l'heure, les propositions apportées ne sont pas totalement satisfaisantes en terme de performance. On observe une dégradation des taux de reconnaissance par rapport aux systèmes sans protection. Dans la plupart des cas, le critère de non-inversibilité, souhaité pour garantir le respect de la vie privée n'est que partiellement atteint.

Dans cette thèse, nous avons présenté un état de l'art sur les différents schémas de protection. Nous nous sommes par la suite intéressés aux transformations révocables. La recherche de telles transformations est un processus complexe, qui nécessite une certaine imagination pour explorer et bousculer l'ordre établi dans les systèmes classiques de vérification biométrique. Nous avons abordé ce thème de recherche par différentes contributions que nous détaillons dans la section 5.1. Des améliorations et des pistes pour des travaux futurs sont ensuite envisagées dans la section 5.2.

5.1 Bilan et principales contributions

En substance, les principales contributions de cette thèse sons les suivantes :

- La définition d'une méthode d'évaluation complète pour les systèmes par transformation révocable.
- La construction de schémas révocables d'empreintes digitales.
- La construction d'une solution de gestion des données biométriques révocables par un système décentralisé de type MatchOnCard conforme aux normes internationales.
- L'étude et l'analyse des problèmes de violation de vie privée dans l'application biométrique de masse, celle des passeports électroniques.

L'intérêt pour le problème d'évaluation des systèmes de protection biométriques s'est tout de suite avéré important et nécessaire. L'absence d'un processus de tests et d'évaluation opérationnel empêche de correctement mesurer leur fiabilité. Des initiatives existent, néanmoins, elles concernent principalement la définition de critères et peu de travaux existent sur la quantification de ces critères. Dans le chapitre 2, une méthodologie d'évaluation a été proposée. Dans ce modèle, le choix des métriques est conditionné par la sélection de la fonction de transformation en entrée. En effet, comme les points d'attaques diffèrent d'une fonction à une autre, les métriques dépendent fortement de cette fonction. A partir de notre étude bibliographique sur les schémas de protection présentés dans le chapitre 1, nous avons pu déduire quatre constructions génériques pour l'ensemble des méthodes présentées. Nous instancions ensuite ce modèle d'évaluation, pour les transformations révocables, l'un des objectifs de cette thèse. Pour mesurer les critères des risques d'intrusion, de révocabilité, d'inversibilité, de divulgation partielle de l'information biométrique, d'intraçabilité et enfin de diversité, nous avons défini 17 métriques.
La définition des métriques est basée sur le modèle de l'adversaire. D'emblée, nous avons associé les métriques de sécurité à un *adversaire malicieux* et les métriques relatives à la vie privée à *l'adversaire honnête mais curieux*. Deux types de métriques ont été définies, pratiques et théoriques. Les métriques pratiques mesurent la complexité des différentes attaques. Les métriques théoriques évaluent la faisabilité de la construction. Nous avons validé ces métriques en les confrontant à une base commune d'attaques. Ces métriques mesurent toutes ces attaques, ce qui démontre leur complétude. Cette complétude est aussi visible lorsque nous situons nos métriques par rapport au travail précurseur de Nagar *et al.* [NJ09] qui traite aussi de l'évaluation des transformations révocables. Enfin, ce modèle d'évaluation nous a permis de tester et d'analyser les méthodes proposées dans le cadre de cette thèse.

Questions	Réponses
Comment le modèle biométrique est-il représenté ?	Représentation globale de l'empreinte par descripteurs de texture. Sections 3.2.1 et 3.2.2 Structures locales des minuties et consolidation par l'information de voisinage. Section 3.3.1
Quel est le schéma de transformation appliqué ?	Projection aléatoire et binarisation. Section 3.2.3
Comment l'alignement géométrique des empreintes a-t-il été traité ?	Modèle global : point core, apprentissage du modèle moyenne. Section 3.2.2 Modèle local : minutie de référence, pré-alignement par correction de l'orientation relative et structures locales auto-alignées. Section 3.3.1
Comment le modèle peut-être révoqué ?	Attribution d'une nouvelle clé utilisateur à chaque révocation. Section 3.2.3
Comment évaluer le nouveau système biométrique ?	Modèle d'évaluation proposé dans le chapitre 2

TABLE 5.1: Résumé des propositions concernant les schémas révocables

Les seconds travaux sont entièrement consacrés à la formulation des modèles révocables d'empreintes digitales et ont été présentés dans le chapitre 3. Un schéma de transformation révocable est une chaine de traitement complexe qui implique différents modules : pré-traitement de l'image, extraction des caractéristiques, protection de ces caractéristiques et enfin comparaison et prise de décision.

Dans cette thèse, nous nous sommes intéressés à chacune de ces phases. Les réponses apportées aux différentes questions posées dans l'introduction générale sont résumées dans le tableau 5.1.

Nous nous sommes intéressés à l'étude de texture pour les deux raisons suivantes : *(i).* L'information de texture est représentée sous forme d'un vecteur, souhaité pour certaines fonctions de transformation. *(ii).* Le descripteur de texture est plus sécurisé que les minuties car il ne permet pas la reconstruction de l'image initiale. Une étude comparative entre un certain nombre de descripteurs de texture a permis de sélectionner le plus approprié.

En se basant sur le point core, le descripteur global de l'empreinte a été créé. Un schéma révocable par projection aléatoire a ensuite été utilisé. L'évaluation de la méthode a montré un taux d'erreur important, non envisageable en pratique.
Pour améliorer les résultats, l'exploration des minuties s'est avérée nécessaire. Le modèle de texture a été projeté sur chacune des minuties et a été transformé aussi par projection aléatoire pour générer le modèle révocable. L'évaluation de la méthode a démontré plusieurs points forts : l'inversibilité totale est impossible, le taux d'erreur en considérant le pire scénario de vol de clé est à 6.68% sur la FVC2002-DB2 et 3.78% sur la FVC2002-DB1, ce qui est assez acceptable. La diversité de la méthode a été prouvée et c'est là son point le plus fort.
Il reste à enrayer la possibilité de divulgation partielle de l'information biométrique dans le cas des références multiples. Plusieurs pistes étaient envisageables. Dans cette thèse, nous nous sommes intéressés à un terrain peu exploré, à savoir l'intégration de la biométrie révocable dans un élément sécurisé.

L'étude de la gestion décentralisée des modèles biométriques a été présentée dans le chapitre 4. Pour éviter les attaques, actuelles et possibles, il est important de gérer en toute sécurité la clé utilisateur et le modèle transformé. Un système MoC le permet. Le modèle des minuties Bio-hashées est compact, il est de 500 octets, ce qui permet son stockage dans la carte à puce. De plus, l'algorithme de comparaison proposé est compatible avec les traitements des JavaCards : pas de virgule flottante, structures de données linéaires, pas d'algorithme de tri (assez gourmand en temps de calcul) et comparaison des structures locales facilitée par une simple distance de Hamming. Le temps de calcul estimé est de 30 secondes. Une conception objet basée sur la norme PKCS15 a été choisie. Le but est d'être plus pratique en générant une carte d'identité personnelle compatible et facile à intégrer. En offrant ce niveau d'abstraction et d'uniformité, l'utilisation de la carte dans diverses applications comme le *e-government* devient évidente.
Un autre côté pratique de la biométrie révocable pourrait être celui des passeports biométriques. Une étude faite dans le chapitre 4 a démontré que les protocoles cryptographiques initiés pour contrôler l'accès aux données sensibles dans la puce du passeport ne sont pas complètement aboutis. Des risques d'attaques et de violation de vie privée peuvent exister. L'usager sera sûrement plus réconforté en sachant que son empreinte n'est pas directement stockée sur le passeport mais une version non inversible et de plus révocable de cette empreinte.

5.2 Perspectives et futures recherches

La réponse à une question ouvre toujours la voie à de nouvelles perspectives de recherche. Ce travail peut être amélioré et étendu de différentes manières.

La programmation des JavaCards est tout un art qui requiert plus de temps pour en maîtriser les subtilités. Comme perspective immédiate, nous entendons améliorer le temps d'exécution du module de comparaison sur la JavaCard. Un code optimisé, qui prend en compte les particularités de la carte, devrait être considéré. Voici un exemple de bonnes pratiques que nous n'avons pas encore testées : factoriser le code redondant, réduire le nombre de méthodes et de classes, allouer les tableaux de manière plus rigoureuse, limiter l'accès à la mémoire EEPROM, réutiliser les variables locales, etc.

De meilleures performances de reconnaissance sont souhaitées pour le schéma révocable par descripteurs locaux. En analysant de plus près les raisons de cette insuffisance, nous trouvons que notre méthode a du mal à apparier les empreintes du même individu lorsque le chevauchement entre elles n'est pas suffisant. En effet, dans la création du descripteur, une sélection des minuties est faite ce qui réduit leur nombre. Si le chevauchement est partiel, le nombre des minuties est insuffisant pour effectuer l'appariement. Il faudrait réfléchir à une méthode qui inclut toutes les minuties même lorsque leur descripteur de texture est incomplet en utilisant une approche dynamique de type *composants les plus fiables* par exemple. D'après nos différents tests et expérimentations, une partie non négligeable de cette thèse pour ne pas dire la plus envahissante, nous estimons qu'une bonne solution de transformation révocable serait de créer un descripteur pour la minutie de référence le plus informatif possible, pas uniquement sur son voisinage, mais en considérant toutes les autres informations qui existent autour d'elle et que nous avons à mettre en évidence. Il est aussi important de proposer des représentations binaires qui sont devenues communes dans les schémas de protection. C'est là notre point de départ pour proposer de meilleurs schémas révocables d'empreintes digitales.

Par rapport à la fonction de transformation, nous retenons une bonne diversité due à la projection aléatoire. Les taux de reconnaissance diminuent lorsque le processus de binarisation est inclus. Il est donc intéressant d'avoir un processus de binarisation optimal qui est plus fidèle à la distribution statistique du descripteur biométrique mais qui quand

même génère une ambiguïté totale pour empêcher une possible prédiction.

L'un des points faibles des transformations révocables est le problème de vol de clé. Nous avons géré ce problème par l'utilisation d'architectures sécurisées. Dans le futur, nous espérons aborder ce problème au niveau algorithmique. Nous réfléchissons à des solutions qui combineraient une transformation révocable avec un crypto-système biométrique. Ce thème a été très peu exploré par la recherche alors qu'il pourrait être utile d'utiliser des approches de type secure sketch sur la biométrie révocable dans le but de combiner les avantages des deux approches (i.e. les transformations révocables offrent une bonne diversité et les secure sketches éliminent le risque de vol de clé car celle-ci est cachée dans la donnée auxiliaire).

Nous avons présenté dans le chapitre 3 un calcul de distance par facteur de vraisemblance entre deux représentations globales de l'empreinte. Les taux d'erreur étaient assez intéressants. Nous envisageons d'étudier la possibilité de présenter un schéma de vérification biométrique respectueux de la vie privée en effectuant le calcul du facteur de vraisemblance par cryptographie hommomorphique de tel sorte que tout le protocole de vérification se fasse dans l'espace chiffré.

Nous espérons aussi nous intéresser à la sécurité et au respect de la vie privée des modèles biométriques dans les bases de données d'identification et non uniquement de vérification ou d'authentification. Il faudrait analyser les contraintes et les exigences en relation, avant d'explorer de possibles solutions.

L'authentification biométrique sur les plateformes Androïde constitue un enjeu de taille. La sécurité doit être impérativement garantie car il s'agit de dispositifs personnels, appelés à être impliqués dans des applications sensibles comme le payement mobile. Les derniers appareils sur le marché sont dotés de différents capteurs et d'outils pour effectuer des login par la reconnaissance du visage, des empreintes digitales ou de l'iris. Dernièrement, certains de ces capteurs ont été facilement trompés par des moulages de doigts. Dans ce contexte, l'utilisation de la biométrie révocable peut s'avérer nécessaire pour protéger l'utilisateur contre le vol d'identité. Nous espérons donc prospecter dans ce sens aussi.

Une nouvelle génération de la reconnaissance biométrique est en cours. Nous restons enthousiastes à l'idée de contribuer à cette transition.

Publications de l'auteur

1 Chapitre de livre

[1] **R. Belguechi**, V. Alimi, E. Cherrier, P. Lacharme and C. Rosenberger. "An Overview on Privacy Preserving Biometrics". In *Recent Application in Biometrics*. Sous la dir. de Jucheng Yang et Norman Poh. Intech. Chap. 4, p. 65-84. Juillet 2011.
URL http ://cdn.intechopen.com/pdfs/17038/InTech-An-overview-on-privacy-preserving-biometrics.pdf

2 Article de revue internationale avec comité de rédaction

[1] **R. Belguechi**, P. Lacharme, C. Rosenberger and S. Ait-Aoudia. "Enhancing the privacy of electronic passports". In *International Journal of Information Technology and Management (IJITM). Special Issue on : "Advances and Trends in Biometrics"*. Inderscience. vol. 11, p. 1-16, 2012.

[2] **R. Belguechi**, E. Cherrier, C. Rosenberger and S. Ait-Aoudia. "Operational Bio-Hash to Preserve Privacy of Fingerprint Minutiae Templates". In *IET journal on Biometrics*. Vol. 2(2), p. 76-84, June 2013.

[3] **R. Belguechi**, E. Cherrier, C. Rosenberger C and S. Ait-Aoudia. "An integrated framework combining Bio-Hashed minutiae template and PKCS15 compliant card for a better secure management of fingerprint cancelable templates". In *Elsevier Journal on Computers & Security*. Vol. 39, p. 325-399, November 2013.

3 Conférence internationale avec actes et comité de sélection

[1] **R. Belguechi**, E. Cherrier, C. Rosenberger. "Texture based Fingerprint BioHashing : Attacks and Robustness". In *IEEE/IAPR International Conference on Biometrics (ICB)*. New Delhi. p. 196-201, 2012.

[2] **R. Belguechi**, E. Cherrier, C. Rosenberger. "How to Evaluate Transformation Based Cancelable Biometric Systems ?". In *NIST International Biometric Performance Testing Conference (IBPC)*. Gaithersburg, United states. 2012.

[3] **R. Belguechi**, E. Cherrier, M. El Abed and C. Rosenberger. "Evaluation of Cancelable Biometric Systems : Application to Finger-Knuckle-Prints". In *IEEE International Conference on Hand-Based Biometrics (ICHB)*. Hong Kong. p. 1-6, 2011.

[4] **R. Belguechi**, C. Rosenberger, S. Ait-Aoudia. "BioHashing for securing fingerprint minutiae templates". In *the 20th International Conference on Pattern Recognition (ICPR)*. Istanbul. p. 1168-1171, 2010.

[5] **R. Belguechi**, C. Rosenberger, S. Ait-Aoudia. "Cancelable authentication based on fingerprints texture". In *International congress on models, optimization and security of systems (ICMOSS)*.Tiaret, Algérie. 2010.

4 Conférence nationale avec actes et comité de sélection

[1] **R. Belguechi**, T. Le Gof, E. Cherrier, C. Rosenberger. "Etude de la robustesse d'un système de biométrie révocable". In *Conférence sur la sécurité des architectures réseaux et des systèmes d'information (SAR SSI)*. La Rochelle, France. 2011.

[2] **R. Belguechi**, V. Alimi, C. Rosenberger. "Secure and Privacy Preserving Management of Biometric Templates". In *The third Norsk Information security conference (NISK)*. Norway. 2010.

[3] **R. Belguechi**, B. Hemery, C. Rosenberger. "Authentification révocable pour la vérification basée texture d'empreintes digitales". In *Congrès Francophone en Reconnaissance des Formes et l'Intelligence Artificielle (RFIA)*. France, 2010.

[4] D. Petrovska-Delacrétaz, S. Kanade, **R. Belguechi**, C. Rosenberger, B. Dorizzi. "L'usage de la biométrie est il contradictoire avec le respect de la vie privée ?". In *Workshop Interdisciplinaire sur la Sécurité Globale (WISG)*. France. 2010.

5 Logiciel

[1] **R. Belguechi**. "Plateforme de reconnaissance d'empreintes digitales". Logiciel sous C♯ Microsoft Visual Studio 8.

[2] **R. Belguechi**. "Applet PKCS15 et gestionnaire de la carte". Logiciel sous Eclipse, Javacard 2.2.1 et JCOP SDK.

Bibliographie

[Adl07] Andy Adler. *Handbook of biometrics*, chapter Biometric system security. 2007. [cité p. 2]

[AHW11] T. Ahmad, J. Hu, and S. Wang. Pair-polar coordinate-based cancelable fingerprint templates. *Pattern recognition*, 44(1), January 2011. [cité p.50, 51, 52, 56, 57, 132, 178]

[AI08] ANSI-INCITS. Ansi-incits fingerprint minutia format for data interchange, 2008. [cité p. 21]

[AJH07] A. Arakala, J. Jeffers, and K. J. Horadam. Fuzzy extractors for minutiae-based fingerprint authentication. In *2nd International Conference on Biometrics*, 2007. [cité p. 39]

[AKQ08] G. Avoine, K. Kalach, and J.J. Quisquater. Epassport : Securing international contacts with contactless chips. In Springer-Verlag, editor, *Proceedings of FC'08, Lecture Notes in Computer Sciences*, volume 5143, pages 141–155, 2008. [cité p. 155]

[ARM05] R. Ang, S.N. Rei, and L. McAven. Cancellable key-based fingerprint templates. In *Information Security and Privacy : 10th Australasian Conference, ACISP 2005, Brisbane, Australia*, pages 242–252, 2005. [cité p. 47, 56, 132]

[BAH+07] F. Benhammadi, M.N. Amirouche, H. Hentous, K. Bey Beghdad, and M. Aissani. Fingerprint matching from minutiae texture maps. *Pattern recognition*, 40 :189–197, 2007. [cité p. 118]

[BBC+10] M. Barni, T. Bianchi, D. Catalano, M. D. Raimondo, R. D. Labati, P. Failla, D. Fiore, R. Lazzeretti, V. Piuri, F. Scotti, and A. Piva. Privacy-preserving fingercode authentication. In *12th ACM Multimedia and Security Workshop*, 2010. [cité p. 34]

[BBGK08] J. Breebaart, C. Busch, J. Grave, and E. Kindt. A reference architecture for biometric template protection based on pseudo identities. In *Proceedings of the Special Interest Group on Biometrics and Electronic Signatures (BIOSIG), Germany*, 2008. [cité p. 35]

[BC09] J. Bringer and H. Chabanne. An authentication protocol with encrypted biometric data. In *AfricaCrypt'09*, 2009. [cité p. 41]

[BCP+03] R. Bolle, J. Connell, S. Pankanti, N. Ratha, and A. Senior. *Guide to Biometrics*. 2003. [cité p. 27]

[BCA10] R. Belguechi, C. Rosenberger, S. Ait-Aoudia. "BioHashing for securing fingerprint minutiae templates". In *the 20th International Conference on Pattern Recognition (ICPR). Istanbul*. p. 1168-1171, 2010.

[BCR02] R. M. Bolle, J. H. Connell, and N. K. Ratha. Biometric perils and patches. *Pattern Recognition*, 35(12) :2727–2738, 2002. [cité p. 2, 48]

[BCRA13a] R. Belguechi, E. Cherrier, C. Rosenberger and S. Ait-Aoudia. "Operational Bio-Hash to Preserve Privacy of Fingerprint Minutiae Templates". In *IET journal on Biometrics*. Vol. 2(2), p. 76-84, June 2013.

[BCRA13b] R. Belguechi, E. Cherrier, C. Rosenberger C and S. Ait-Aoudia. "An integrated framework combining Bio-Hashed minutiae template and PKCS15 compliant card for a better secure management of fingerprint cancelable templates". In *Elsevier Journal on Computers & Security*. Vol. 39, p. 325-399, November 2013.

[BD10] J. Bringer and V. Despiegel. Binary feature vector fingerprint representation from minutiae vicinities. In *Fourth IEEE International Conference on Biometrics: Theory Applications and Systems (BTAS),* 2010. [cité p. 45]

[Bel06] R. Belguechi. Reconnaissances des empreintes digitales par une approche hybride. Master's thesis, Ecole nationale superieure d'informatique, 2006. [cité p. 21, 177]

[BKS10] I. Buhan, E. Kelkboom, and K. Simoens. A survey of the security and privacy measures for anonymous biometric authentication systems. In *Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)*, 2010. [cité p. 58, 64, 72]

[Buh08] I. Buhan. *Cryptographic keys from Noisy Data Theory and Applications*. PhD thesis, University of Twente, Netherlands, 2008. [cité p. 27]

[Cav99] A. Cavoukian. Privacy and biometrics. In *Proceedings of International Conference on Privacy and Personal Data Protection, Hong Kong, China*, 1999. [cité p. 33]

[Cav08] Ann Cavoukian. Fingerprint biometrics : Address privacy before deployment. Technical report, Information and privacy commissioner, Ontario, November 2008. [cité p. 2, 33]

[CCG06] S. Chikkerur, A.N. Cartwright, and V. Govindaraju. K-plet and coupled bfs : A graph based fingerprint representation and matching algorithm. In *Proc. Int. Conf. on Biometrics, LNCS 3832*, page 309U″ 315, 2006. [cité p. 121]

[CCG07] S. Chikkerur, A.N. Cartwright, and V. Govindaraju. Fingerprint enhancement using stft analysis. *Pattern Recognition*, 40(1) :198–211, 2007. [cité p. 87]

[CFM08] R. Cappelli, M. Ferrara, and D. Maltoni. On the operational quality of fingerprint scanners. *IEEE Transactions on Information Forensics and Security,* 3(2) :192–202, 2008. [cité p. 17]

[CFM10] R. Cappelli, M. Ferrara, and D. Maltoni. Minutia cylinder-code : A new representation and matching technique for fingerprint recognition. *IEEE transactions Pattern Analysis and Machine Intelligence*, 32 :2128–2141, 2010. [cité p. 23]

[CKL03] T.C. Clancy, N. Kiyavash, and D. J. Lin. Secure smartcard-based fingerprint authentication. In *ACM SIGMM 2003 Multimedia, Biometrics Methods and Applications Workshop*, pages 45–52, 2003. [cité p. 40]

[CLMM07a] R. Cappelli, A. Lumini, D. Maio, and D. Maltoni. Evaluating minutiae template vulnerability to masquerade attack. In *5th Workshop on Automatic Identification Advances Technologies (AutoID2007), Alghero*, June 2007. [cité p. 30, 31]

[CLMM07b] R Cappelli, A Lumini, D. Maio, and D. Maltoni. Fingerprint image reconstruction from standard templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(9) :1489–1503, September 2007. [cité p. 30, 31, 76, 119]

[Com10] Privacy Commissioners. Privacy by design resolution. In *32nd international conference of data protection and privacy commissioners*, 2010. [cité p. 33]

[CS10] T. Chotia and V. Smirnov. A traceability attacks against e-passports. In *Proceedings of FC'10, Lecture Note in Computer Sciences, Springer-Verlag*, 2010. [cité p. 152]

[CV09] R. Chaabouni and S. Vaudenay. The extended access control for machine readable travel documents. In *Proceedings of BIOSIG'09*, volume 155, pages 93–103, 2009. [cité p. 156]

[Dau03] J. Daugman. The importance of being random : statistical principles of iris recognition. *Pattern Recognition*, 36(2) :279–291, 2003. [cité p. 72]

[DFM99] G. I. Davida, Y. Frankel, and B. J. Matt. On the relation of error correction and cryptography to an offline biometric based identification scheme. In *WCC99, Workshop on coding and cryptography*, 1999. [cité p. 38]

[DG99] S. Dasgupta and A. Gupta. An elementary proof of the johnson-lindenstrauss lemma. Technical Report TR-99-006, International Computer Science Institute, Berkeley, CA, 1999. [cité p. 100]

[DHS00] R.O. Duda, P.E. Hart, and D.G. Stork. *Pattern classification*. Wiley interscience, 2000. [cité p. 94]

[DKM+07] S.C. Draper, A. Khisti, E. Martinian, A. Vetro, and J.S. Yedidia. Using distributed source coding to secure fingerprint biometrics. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2007. [cité p. 43, 61]

[Dom04] V. Domnesque. Carte d'identit électronique & conservation des données biométriques. Master's thesis, Université de Lille, 2004. [cité p. 31]

[DRS04] Y. Dodis, L. Reyzin, and A. Smith. Fuzzy extractors : How to generate strong keys from biometrics and other noisy data ? *in Adv. in Cryptology Eurocrypt 2004, LNCS*, 3027 :523–540, 2004. [cité p. 37, 61, 71]

[DTL98] C. Domeniconi, S. Tari, and P. Liang. Direct gray scale ridge reconstruction in fingerprint images. In *ICASSP*, 1998. [cité p. 20]

[EA11] Mohammed El-Abed. *Evaluation des systèmes biométriques*. PhD thesis, Université de Caen, 2011. [cité p. 24]

[Fen08] J. Feng. Combining minutiae descriptors for fingerprint matching. *Pattern Recognition*, 41(1) :342–352, 2008. [cité p. 24]

[FJ09] J. Feng and A. Jain. Fm model based fingerprint reconstruction from minutiae template. In *International conference on Biometrics (ICB)*, 2009. [cité p. 30, 31, 119]

[FMC12] M. Ferrara, D. Maltoni, and R. Cappelli. Non-invertible minutia cylinder-code representation. *IEEE Trans. Inf. Forensics Secur.*, 7 :1727–1737, 2012. [cité p. 53]

[fSidI08] Bundesalt fur Sicherheit in der Informationstechni. Advanced security mechanism for machine readable travel documents. Technical report, 2008. [cité p. 154]

[GB07] M. Govan and T. Buggy. A computationally efficient fingerprint matching algorithm for implementation on smartcards. In *in Proc. Int. Conf. on Biometrics : Theory, Applications, and Systems (BTAS 07)*, pages 1–6, 2007. [cité p. 33, 136]

[GCL+08] J. Galbally, R. Cappelli, A. Lumini, D. Maltoni, and J. Fierrez-Aguilar. Fake fingertip generation from a minutiae template. In *19th International Conference on Pattern Recognition (ICPR2008), Tampa, Florida, USA*, pages 1–4. IBM Best Student Award, December 2008. [cité p. 28, 31]

[Gro02] Common Criteria Working Group. Biometric evaluation methodology supplement, 2002. [cité p. 27]

[gro13] International Biometric group. Biometrics market and industry report 2009-2014. Technical report, 2013. [cité p. 5]

[GWMW02] M.D. Garris, C.I. Watson, R.M. McCabe, and C.L. Wilson. Users guide to nist fingerprint image software (nfis). Technical report, National Institute of Standards and Technology, 2002. [cité p. 20]

[Hil01] C.J. Hill. Risk of masquerade arising from the storage of biometrics. Bachelor of science, Australian National University, 2001. [cité p. 59, 74]

[Hof89] W Hoffmann. Iterative algorithms for gram-schmidt orthogonalization. *Computing*, 41(4) :335–348, 1989. [cité p. 100]

[HR07] M. Hlavac and T. Rosa. A note on the relay attacks on e-passport : the case of czech e-passport. Technical report, 2007. [cité p. 152]

[HSK10] O. Henniger, D. Scheuermann, and T. Kniess. On security evaluation of fingerprint recognition systems. In *International biometric Performance testing conference*, 2010. [cité p. 27]

[HWJ98] L. Hong, Y. Wan, and A. Jain. Fingerprint image enhancement : Algorithm and performance evaluation. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 20 :777–789, 1998. [cité p. 20]

[HZ08] A. Hafiane and B. Zavidovique. Local relational string and mutual matching for image retrieval. *Inf. Process. Manage.*, 44 :1201–1213, 2008. [cité p. 83]

[Ign09] Tanya Ignatenko. *Secret-Key Rates and Privacy Leakage in Biometric Systems*. PhD thesis, Eindhoven : Technische Universteit Eindhoven, 2009. [cité p. 56]

[ISO06a] ISO/IEC. Iso/iec 19795. information technology - biometric performance testing and reporting -, 2006. [cité p. 25]

[ISO06b] ISO/IEC. Iso/iec cd 19792 : Information technology - security techniques - security evaluation of biometrics, 2006. [cité p. 27]

[ISO08] ISO/IEC. International standard iso/iec 19794-2 information technology - biometric data interchange formats - part 2 : Finger minutia data, 2008. [cité p. 21]

[ISO09] ISO/IEC. Iso/iec 15408. security techniques - evaluation criteria for it security -, 2009. [cité p. 27]

[ISO11] ISO/IEC. Iso/iec 24745 :2011 information technology - security techniques - biometric information protection., 2011. [cité p. 35]

[JHPB97] A Jain, L Hong, S Pankanti, and R Bolle. An identity-authentification system using fingerprints. *Proceedings of The IEEE*, 85(9), 1997. [cité p. 20, 23]

[JLS04] A.T.B. Jin, D.N.C. Ling, and O.T. Song. An efficient fingerprint verification system using integrated wavelet and fourier-mellin invariant transform. *Image and Vision Computing*, 22(6) :503–513, 2004. [cité p. 76]

[JMW05] A. Juels, D. Molnar, and D. Wagner. Security and privacy issues in e-passports. In *IEEE SecureComm*, 2005. [cité p. 155]

[JNN08] A. K. Jain, K. Nandakumar, and A. Nagar. Biometric template security. *EURASIP Journal on Advances in Signal Processing*, pages 1–17, 2008. [cité p. 27, 37, 58, 74]

[JPHP00] A.K. Jain, S. Prabhakar, L. Hong, and S. Pankanti. Filterbank based fingerprint matching. *IEEE Trans Image Process*, 9(5) :846–859, 2000. [cité p. 22, 76, 89, 90, 92]

[JS02] A. Juels and M. Sudan. A fuzzy vault scheme. In *Proceedings of IEEE International Symposium on Information Theory, Lausanne, Switzerland*. IEEE Press, 2002. [cité p. 40, 57]

[JTOT12] Z Jin, A.B.J. Teoh, T.S. Ong, and C. Tee. Fingerprint template protection with minutiae-based bit-string for security and privacy preserving. *Expert systems with applications, Elsevier*, 39 :6157–6167, 2012. [cité p. 51, 52, 132, 178]

[JW99] A. Juels and M. Wattenberg. A fuzzy commitment scheme. In *Sixth ACM Conference on Computer and Communications Security*, pages 28–36. ACM Press, 1999. [cité p. 38, 39, 56]

[Kas98] S. Kaski. Dimensionality reduction by random mapping. In *Int. Joint Conf. on Neural Networks*, volume 1, pages 413–418, 1998. [cité p. 99]

[KCZ+05] A. Kong, K.H. Cheung, D. Zhang, M. Kamel, and J. You. An analysis of biohashing and its variants. *Pattern Recognition*, 39, 2005. [cité p. 58, 74]

[KTG10] G Kumar, S Tulyakov, and V. Govindaraju. Combination of symmetric hash functions for secure fingerprint matching. In *International conference on pattern recognition*, 2010. [cité p. 48, 56, 132]

[LCT+07] C. Lee, J.Y. Choi, K.A. Toh, S. Lee, and J. Kim. Alignment-free cancelable fingerprint templates based on local minutiae information. *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS*, 37(4), August 2007. [cité p. 57]

[LH13] C. Li and J. Hu. Attacks via record multiplicity on cancelable biometrics templates. *Concurrency Computat. : Pract. Exper*, 2013. [cité p. 53, 57, 59, 74, 133]

[LJY02] E. Lim, X. Jiang, and W. Yau. Fingerprint quality and validity analysis. In *IEEE ICIP*, 2002. [cité p. 17]

[LK10] C. Lee and J. Kim. Cancelable fingerprint templates using minutiae-based bit-strings. *Journal of Network and Computer Applications*, 33 :236–246, 2010. [cité p. 49, 50, 56, 132, 178]

[LN06] A. Lumini and L. Nanni. Empirical tests on biohashing. *NeuroComputing*, 69(16) :2390–2395, October 2006. [cité p. 8, 109]

[LN07] A. Lumini and L. Nanni. An improved biohashing for human authentication. *Pattern recognition*, 40 :1057–1065, 2007. [cité p. 94, 109]

[LPY+12] J. Lei, Q. Peng, X. You, H. Jabbar, and P. Wang. Fingerprint enhancement based on wavelet and anisotropic filtering. *International journal of pattern recognition and artificial intelligence*, 26, 2012. [cité p. 20]

[LT03] J.M.G. Linnartz and P. Tuyls. New shielding functions to enhance privacy and prevent misuse of biometric templates. In *4th International Conference on Audio- and Video-Based Biometric Person Authentication*, volume 2688, pages 393–402, 2003. [cité p. 42, 61]

[MDLS07] G. Madlmayr, O. Dillinger, J. Langer, and C. Schaffer. The benefit of using sim application toolkit in the context of near field communication application. In *ICMB*, 2007. [cité p. 134]

[MM08] E. Mordini and S. Massari. Body, biometrics and identity. *Bioethics journal*, 2008. [cité p. 30, 31]

[MMJP09] D. Maltoni, D. Maio, A.K. Jain, and S. Prabhakar. *Handbook of Fingerprint Recognition*. Springer, 2009. [cité p. 24, 28, 35, 39, 40, 76]

[MMYH02] T Matsumoto, H. Matsumoto, K. Yamada, and S. Hoshino. Impact of artificial gummy fingers on fingerprint systems. *Proceedings of SPIE , Optical Security and Counterfeit Deterrence Techniques IV*, 4677, 2002. [cité p. 28]

[MN07] R.M. McCabe and E.M. Newton. Data format for the interchange of fingerprint, facial and other biometric information, 2007. [cité p. 21]

[NJ09] A. Nagar and A.K. Jain. On the security of non-invertible fingerprint template transforms. In *WIFS*, 2009. [cité p. 48, 59, 60, 71, 74, 107, 133, 158]

[NJP07] K. Nandakumar, A.K. Jain, , and S. Pankanti. Fingerprint-based fuzzy vault : Implementation and performance. *IEEE Transactions on Information Forensics and Security*, 2 :744–757, December 2007. [cité p. 40, 44]

[NL09] L. Nanni and A. Lumini. Descriptors for image-based fingerprint matchers. *Expert Systems With Applications*, 36(10), 2009. [cité p. 76]

[NNJ10a] A. Nagar, K. Nandakumar, and A.K. Jain. Biometric template transformation : A security analysis. *Proceeding of SPIE, Electronic Imaging, Media Forensics and Security, Sans Jose*, January 2010. [cité p. 57, 64, 65]

[NNJ10b] A. Nagar, K. Nandakumar, and A.K. Jain. A hybrid biometric cryptosystem for securing fingerprint minutiae templates. *Pattern Recognition Letters*, 31(8) :733–741, 2010. [cité p. 41]

[NRV10] A. Nagar, S. Rane, and A. Vetro. Privacy and security of features extracted from minutiae aggregates. In *International Conference on Acoustics, Speech and Signal Processing*, 2010. [cité p. 44]

[Org04] International Civil Aviation Organization. Doc 9303 : Machine readable travel documents. Technical report, 2004. [cité p. 152, 153]

[Ots79] N. Otsu. A threshold selection method from gray level histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9 :62–66, 1979. [cité p. 92]

[PKC00a] PKCS11. Pkcs11 : Cryptographic token interface standard, 2000. [cité p. 137]

[PKC00b] PKCS15. Pkcs15 : Cryptographic token information format standard, 2000. [cité p. 9, 137]

[PPJ02] S. Pankanti, S. Prabhakar, and A. Jain. On the individuality of fingerprints. *IEEE Transactions on Pattern Analysis And Machine Intelligence*, 24(8) :1010–1025, 2002. [cité p. 30]

[QFAF08] F. Quan, S. Fei, C. Anni, and Z. Feifei. Cracking cancelable fingerprint template of ratha. In *International Symposium on Computer Science and Computational Technology*, 2008. [cité p. 57]

[QZX06] L. Qiong, L. Zhaoping, and N. Xiamu. Analysis and problems on fuzzy vault scheme. In *International conference on intelligent information hiding and multimedia signal processing*, 2006. [cité p. 40]

[Rao90] A.R. Rao. *A Taxonomy for Texture Description and Identification*. Springer-Verlag, New York, 1990. [cité p. 89]

[RCB01] N. K. Ratha, J. H. Connell, and R. M. Bolle. Enhancing security and privacy in biometrics-based authentication systems. *IBM Systems Journal*, 40(03) :614–634, 2001. [cité p. 2, 27, 28, 177]

[RCCB07] N. K. Ratha, S. Chikkerur, J. H. Connell, and R. M. Bolle. Generating cancelable fingerprint templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(4) :561–572, 2007. [cité p. 48, 50, 56, 57, 132]

[RCJ95] N Ratha, S Chen, and A Jain. Adaptive flow orientation based texture extraction in fingerprint images. *Pattern recognition*, 28(11) :1657–1672, Nov 1995. [cité p. 20]

[RKCJ96] N.K. Ratha, K. Karu, S. Chen, and A.K. Jain. A real-time matching system for large fingerprint databases. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 18(8) :799–813, 1996. [cité p. 22]

[RMP08] H. Richter, W. Mostowski, and E. Poll. Fingerprinting passports. In *Spring Conference on Security*, 2008. [cité p. 152]

[RSJ07] A. Ross, J. Shah, and A.K. Jain. From template to image : reconstructing fingerprints from minutiae points. *IEEE Transactions on Pattern Analysis and Machine Intelligence, Special Issue on Biometrics*, 29(4) :544–560, 2007. [cité p. 31]

[RU11] Christian Rathgeb and Andreas Uhl. A survey on biometric cryptosystems and cancelable biometrics. *EURASIP Journal on Information Security*, 3, 2011. [cité p. 46]

[SB07] W. J. Scheirer and T. E. Boult. Cracking fuzzy vaults and biometric encryption. In *Proceedings of Biometrics Symposium*, 2007. [cité p. 41, 57, 59, 74]

[SB08] W.J. Scheirer and T.E. Boult. Bio-cryptographic protocols with bipartite biotokens. In *Proc. Biometric Symposium*, 2008. [cité p. 136]

[Sch99] B. Schneier. The uses and abuses of biometrics. In *Comm. ACM*, volume 42, page 136, Aug 1999. [cité p. 30]

[SKK01] L.L. Shen, A. Kot, and W.M. Koo. Quality measures of fingerprint images. In *3rd international conference AVBPA*, pages 182–271, June 2001. [cité p. 17]

[Smi04] A. Smith. *Maintaining secrecy when information leakage is unavoidable*. PhD thesis, Massachusetts institute of technology, 2004. [cité p. 41, 56]

[SMM94] B.G. Sherlock, D.M. Monro, and K. Millard. Fingerprint enhancement by directional fourier filtering. *In Visual Image Signal Processing*, 141 :87–94, 1994. [cité p. 87, 88]

[Sol08] D.J. Solove. Understanding privacy. *Harvard university press*, 2008. [cité p. 29]

[SR01] R. Sanchez-Reillo. Including biometric authentication in a smart card operating system. In *AVBPA*, 2001. [cité p. 33]

[SS02] B. Struif and D. Scheuermann. Smartcards with biometric user verification. In *IEEE International Conference on Multimedia and Expo*, 2002. [cité p. 136]

[Sta02] J. Stapleton. American national standard x9.84-2001 biometric informatio, management and security, 2002. [cité p. 33]

[STP09] K. Simoens, P. Tuyls, and B. Preneel. Privacy weaknesses in biometric sketches. *30th IEEE Symposium on Security and Privacy*, pages 188–203, 2009. [cité p. 41, 56]

[SYZ+12] K Simoens, B Yang, X Zhou, F Beato, C Busch, E.M. Newton, , and B. Preneel. Criteria towards metrics for benchmarking template protection algorithms. In *International conference on biometrics*, 2012. [cité p. 57, 58]

[TAK+05] P. Tuyls, A.H.M. Akkermans, T.A.M. Kevenaar, G.J. Schrijen, A.M. Bazen, and R.N.J. Veldhuis. Practical biometric authentication with template protection. In *International Conference on Audio- and Video-Based Biometric Person Authentication*, volume 3546, pages 436–446, 2005. [cité p. 39, 109]

[TFG05] S. Tulyakov, F. Farooq, and V. Govindaraju. Symmetric hash functions for fingerprint minutiae. In *ICAPR*, 2005. [cité p. 48]

[TG04] P. Tuyls and J. Goseling. Capacity and examples of template protection biometric authentication systems. In LNCS, editor, *Biometric authentication workshop*, volume 3087, pages 158–180, 2004. [cité p. 42]

[TH11] K. Takahashi and S. Hirata. Parameter management schemes for cancelable biometrics. In *Computational Intelligence in Biometrics and Identity Management*, 2011. [cité p. 59]

[Thi03] Michael Thieme. Identifying and reducing privacy risks in biometric systems. In *13th annual conference on computers, freedom & privacy*, 2003. [cité p. 5]

[TJ98] M. Tuceryan and A.K. Jain. *The Handbook of Pattern Recognition and Computer Vision (2nd Edition)*, chapter Texture analysis (2.1), pages 207–248. 1998. [cité p. 77]

[TK03] M. Tico and P. Kuosmanen. Fingerprint matching using an orientation-based minutia descriptor. *Pattern Analysis and Machine Intelligence*, 25(8) :1009–1014, 2003. [cité p. 23, 118, 121, 130]

[TKKA10] A.B.J. Teoh, Y.W. Kuan, and T. Kar-Ann. Cancellable biometrics and user-dependent multi-state discretization in biohash. *Pattern analysis & applications*, 13 :301–307, 2010. [cité p. 109]

[TKL07] Andrew B.J. Teoh, YipWai Kuanb, and Sangyoun Leea. Cancellable biometrics and annotations on biohash. *pattern recognition*, 41 :2034–2044, 2007. [cité p. 8]

[TN04] A. B.J. Teoh and D. C.L. Ngo. Cancellable biometerics featuring with tokenised random number. *Pattern Recognition Letters*, 26 :1454–1460, 2004. [cité p. 8]

[TNG04] A. Teoh, D. Ngo, and A. Goh. Biohashing : two factor authentication featuring fingerprint data and tokenised random number. *Pattern recognition*, 37(11), 2004. [cité p. 47, 56, 57, 94, 98, 99, 109]

[TO96] D. Harwood T. Ojala, M. Pietikeinen. A comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*, 29 :51–59, 1996. [cité p. 81]

[TRCB08] A.O. Thomas, N. Ratha, J. Connell, and R. Bolle. Comparative analysis of registration based and registration free methods for cancelable fingerprint biometrics. In *19th International Conference on Pattern Recognition*, 2008. [cité p. 46]

[TWW04] A. Tabassi, C. L. Wilson, and C. I Watson. Fingerprint image quality. Technical report, National Institute of Standards and Technology (NIST), August 2004. [cité p. 17]

[UJ04] U. Uludag and A.K. Jain. Attacks on biometric systems : A case study in fingerprints. In *Proc. Int'l Soc. Optical Eng. (SPIE), Security, Steganography, and Watermarking of Multimedia Contents VI*, pages 622–633, June 2004. [cité p. 31]

[UJ06] U. Uludag and A.K. Jain. Securing fingerprint template : Fuzzy vault with helper data. In *Computer Vision and Pattern Recognition Workshop (CVPR)*, June 2006. [cité p. 40, 56]

[UNSJ10] M Upmanyu, A Namboodiri, K Srinathan, and C. V. Jawahar. Blind authentication : A secure crypto-biometric verification protocol. *IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY*, 5(2), June 2010. [cité p. 42]

[WH12] S. Wang and J. Hu. Alignement-free cancellable fingerprint template design : a densely infinite-to-one mapping (ditom) approach. *pattern recognition, Elsevier*, 45(12) :4129–4137, December 2012. [cité p. 52, 56, 57, 132]

[WHT08] L. Wolf, T. Hassner, and Y. Taigman. Descriptor based methods in the wild. In *Real-Life Images workshop at the European Conference on Computer Vision (ECCV)*, October 2008. [cité p. 82, 178]

[WTWY13] W.J. Wong, A.B.J. Teoh, M.L.D. Wong, and Kho. Y.H. Enhanced multi-line code for minutiae-based fingerprint template protection. *Pattern Recognition Letters*, 34 :1221–1229, 2013. [cité p. 53]

[WZ02] Y. Wang W. Zhang. Core-based structure matching algorithm of fingerprint verification. In *16th International Conference on Pattern Recognition*, 2002. [cité p. 115]

[XVB+09] H. Xu, R. N. J. Veldhuis, A. M. Bazen, T. A. M. Kevenaar, T. A. H. M. Akkermans, and B. Gokberk. Fingerprint verification using spectral minutiae representations. *Transaction on Information Forensics and Security*, 4(3) :397–409, 2009. [cité p. 45]

[YA05] N. Yager and A. Amin. Coarse fingerprint registration using orientation fields. *EURASIP Journal on Applied Signal Processing*, 13 :2043–2053, 2005. [cité p. 23]

[YHSB10] B. Yang, D. Hartung, K. Simeons, and C. Busch. Dynamic random projection for biometric tempalte protection. In *BTAS*, 2010. [cité p. 110, 111]

[YLZC09] S. Ye, Y. Luo, J. Zhao, and S. Cheung. Anonymous biometric access control. *EURASIP J*, 2009. [cité p. 34]

[Zho12] Xuebing Zhou. *Privacy and security assessment of biometric template protection*. PhD thesis, Fachbereich Informatik Universitat Darmstadt, Germany, 2012. [cité p. 57, 62]

[ZKVB11] X. Zhou, A. Kuijper, R. Veldhuis, and C. Busch. Quantifying privacy and security of biometric fuzzy commitment. In *IEEE Proc in International Joint Conference on Biometrics (IJCB)*, 2011. [cité p. 41, 56, 58, 60, 72]

[ZW02] W. Zhang and Y. Wang. Core-based structurematching algorithm of fingerprint verification. In *Proc. 16th IEEE International Conference on Pattern Recognition*, 2002. [cité p. 23]

[ZWBK09a] X. Zhou, S.D. Wolthusen, C. Busch, and A. Kuijper. Feature correlation attacks on biometric privacy protection scheme. In *IEEE Proc in 5th International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, 2009. [cité p. 41]

[ZWBK09b] X Zhou, S.D. Wolthusen, C Busch, and A. Kuijper. A security analysis of biometric template protection schemes. *Lecture Notes in Computer Science, Springer-Verlag*, pages 429–438, 2009. [cité p. 72]

Table des figures

Liste des tableaux

Liste des Algorithmes

Annexe

Annexe A

Fiche sur la théorie bayesienne et la vérification biométrique

Soit V l'espace des modèles de dimension d et $W = w_1, w_2, ..., w_S$ un ensemble de classes disjointes dont les éléments sont dans V.

- $\forall v \in V$ et $\forall w_i \in W$, $P(v/w_i)$ dénote la densité de probabilité conditionnelle pour v sachant w_i.
- $\forall w_i \in W$, $P(w_i)$ dénote la probabilité à priori pour w_i.
- $\forall v \in V$, $P(v)$ dénote la densité de probabilité absolue pour v :

$$P(v) = \sum_{i=1}^{S} P(v/w_i)P(w_i)$$

- $\forall w_i \in W$ et $\forall v \in V$, $P(w_i/v)$ dénote la probabilité à postériori pour w_i sachant v. En utilisant le théorème de Bayes, nous avons :

$$P(w_i/v) = \frac{P(v/w_i).P(w_i)}{P(v)}$$

$$posterior = \frac{Vraisemblance \times priori}{Evidence}$$

En biométrie, chaque personne x est identifiée par son instance biométrique ou son modèle C_x^i. Durant l'acquisition, les modèles $C_x^i, i = 1, 2, ...$ correspondant aux différentes instances du même individu x ne se coïncident pas exactement et forment un ensemble centré sur le modèle noté C_x. Pour chaque individu, une classe w_x est définie, représentée par le modèle C_x.
Nous supposons maintenant que :
Les modèles C_x^i de la classe w_x suivent une distribution normale multi-variée définie par la moyenne C_x et la matrice de covariance Σ_E. La densité de probabilité conditionnelle est alors donnée par la formule suivante :

$$P(C_x^i/w_x) = \frac{1}{2\pi^{d/2} \times |\Sigma_E|^{1/2}} \times \exp(-\frac{1}{2}(C_x^i - C_x)\Sigma_E^{-1}(C_x^i - C_x))$$

Le problème de vérification biométrique peut alors être considéré comme un problème de classification où le but est de calculer la probabilité à postériori $P(w_x/C_x^i)$ en utilisant le théorème de Bayes :

$$P(w_x/C_x^i) = \frac{P(C_x^i/w_x).P(w_x)}{P(C_x^i)}$$

Comme approximation, il est possible d'utiliser la mesure de similarité suivante :

$$\log P(w_x/C_x^i) = -(C_x^i - C_x)\Sigma_E^{-1}(C_x^i - C_x)$$

Le facteur de vraisemblance $L(C_x^i)$ est une autre mesure de similartié pour le problème de classification cité.

$$L(C_x^i) = \frac{P(C_x^i/w_x)}{P(C_x^i/\bar{w}_x)}$$

Avec $P(C_x^i/\bar{w}_x)$: la probabilité de C_x^i sachant qu'il n'est pas membre de la classe w_x.
Le facteur de vraisemblance peut être calculé par l'approximation suivante où Σ_T est la matrice de covariance de toute la population :

$$\log L(C_x^i) = -(C_x^i - C_x)\Sigma_E^{-1}(C_x^i - C_x) + C_x^i\Sigma_T^{-1}(C_x^i)^T$$

Annexe B

Complément sur l'applet PKCS15 Bio

Les modules importants dans la conception de l'applet PKCS15 sont les suivants :

- Gestion de la mémoire
- Gestion des fichiers PKCS15
- Gestion des objets PKCS15
- Gestion de la sécurité

1 Gestion de la mémoire

Dans une JavaCard, la gestion de la mémoire est laissée au soin du développeur. Nous considérons la mémoire comme un tableau d'octets d'une taille maximale *MaxMemSize*.
Nous appelons *objet mémoire* la suite d'octets qui contient deux champs d'en-tête *Size* et *Next*. Le champ *Size* contient la taille de l'objet et le champ *Next* contient l'adresse du prochain objet vide dans la mémoire, −1 sinon.
Au début, le tableau mémoire est un unique objet vide dont le champ *Size* est égal à la taille maximale de la mémoire et le champ *Next* est égal à -1. Au fur et à mesure que des allocations et des suppressions d'objets sont faites, il contiendra un ensemble d'objets vides chainés entre eux. Une variable globale, appelée *FreeLLC Head*, contient l'indice de la première case libre. La figure B.1 illuste la structure de la mémoire interne.

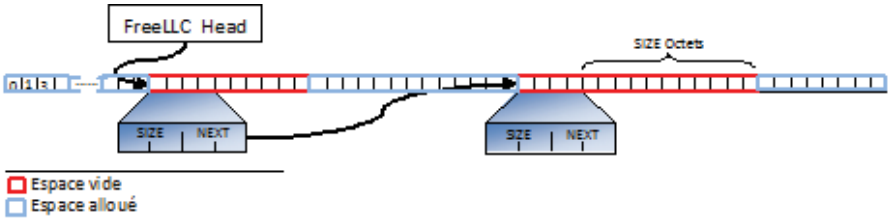


FIGURE B.1 – Structure de la mémoire interne

2 Gestion des fichiers PKCS15

Le standard PKCS15 définit une certaine hiérarchie dont voici les principaux fichiers :

- **Object Directory Files (ODF)** : Ce fichier contient des pointeurs vers les autres fichiers élémentaires (i.e. PrKDF, PuKDF, SKDF, CDF, DODF et AODF). L'ODF a donc une structure orientée enregistrement, où chaque enregistrement est un pointeur vers un autre fichier.
- **TokenInfo File** : Le fichier obligatoire TokenInfo contient des informations génériques sur le support de cryptographie utilisé.
- **Autentication Object Directory Files (AODFs)** : Ce fichier peut être considéré comme un annuaire d'objets d'authentification (par exemple PINs ou données biométriques) connus par l'application PKCS15. Il contient aussi des pointeurs vers les objets d'authentification.
- **PrKDF, SKDF et PuKDF** : Ces fichiers sont considérés comme des annuaires des clés connues par l'application PKCS15. PrKDF contient des informations sur des clés privées, PuKDF contient des informations sur des clés publiques et SKDF contient des informations sur des clés secrètes. Ces fichiers associent les clés aux objets d'authentification et aux certificats, et contiennent des attributs généraux tels que des étiquettes, des restrictions d'utilisation de la clé, la taille de la clé, le type de l'algorithme de chiffrement, l'identifiant de son objet d'authentification, etc. En outre, ils contiennent les pointeurs vers les clés elles-mêmes.
- **Certificate Directory Files (CDF)** : Ce fichier Contient des informations à propos des certificats tels que les identifiants, le type du certificat, etc. Il contient également des pointeurs vers les certificats eux-mêmes. Quand un certificat contient une clé publique correspondant à une clé privée, le certificat et la clé privée partageront un identifiant commun, pour permettre de repérer l'un à partir de l'autre.
- **Data Object Directory Files (DODFs)** : Ces fichiers peuvent être considérés comme des annuaires d'objets de données. Ils contiennent des attributs d'objet et des pointeurs vers les objets de données.

Nous gérons les fichiers PKCS15 conformément à la norme ISO7016-4. Nous adoptons une structure linéaire fixe pour tous les fichiers PKCS15 étant donné que chaque fichier peut être modélisé par un ensemble d'enregistrements de taille fixe. Les principales méthodes de la classe FileSystem relative à la gestion des fichiers PKCS15 sont résumées sur le tableau B.1.

3 Gestion des objets PKCS15

L'applet est capable de manipuler différents types d'objets dont les objets pour l'authentification biométrique. Chaque objet est caractérisé par un descripteur qui se trouve dans le fichier PKCS15 correspondant (i.e. AODF pour les objets biométrie). Un descripteur comprend un ensemble de champs communs à tous les types d'objets (i.e. Type, ID, ACL, Size, Ref) en plus des champs

Méthode	Paramètres	Descriptions
FileSystem	mm	Crée une instance du système de fichier, mm représente la mémoire de l'applet contenant les différents objets et fichiers
CreateFile	FID, maxsize, perms	Crée un fichier dont l'identifiant est FID et la taille maximale est maxsize avec les permissions de lecture, d'écriture et de suppression dans perms
DeletFile	FID	Supprime un fichier si l'opération est accordée
VerifyFilePerms	indexRecord, op	Vérifie si on peut effectuer l'opération op pour un fichier dont l'adresse de son enregistrement dans ODF est indexRecord
SearchFileHeader	FID	Retourne l'adresse mémoire du fichier portant l'identifiant FID

TABLE B.1: Description de la classe FileSystem

spécifiques au type de l'objet en question. La figure B.2 résume le gestion des fichiers et objets PKCS15 au niveau de la mémoire.

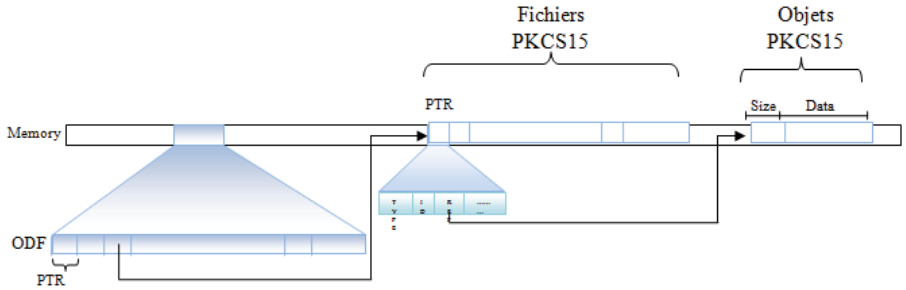


FIGURE B.2 – Fichiers et Objets PKCS15 dans la mémoire

Les principales méthodes de la classe ObjectManager relative à la gestion des objets PKCS15 sont résumées sur le tableau B.2.

Il est exhaustif de décrire tous les éléments de notre applet PKCS15. Nous espérons, au moins, avoir apportés quelques précisions utiles pour le lecteur.

Méthode	Paramètres	Descriptions
ObjectManager	mm, fs	Le constructeur de la classe qui va être exécuté par toutes les sous-classes gérant un type d'objet particulier, mm représente la mémoire, et fs le système de fichier
Create	ptr, size	Crée un objet dans son fichier, ptr l'enregistrement de l'objet à ajouter dans le fichier, size la taille de ptr
Write	ID, offset, OBJ	Ecrit dans un objet d'identifiant ID à partir de l'index OFFSET
VerifyObjectPerms	indexRecord, logged, op	Vérifie si l'opération op est permise pour un objet dont l'adresse de son enregistrement dans son fichier conteneur est indexRecord, logged représente les identités authentifiées

TABLE B.2: Description de la classe ObjectManager