Résumé

Le besoin d'accès sécurisés et automatisés à des environnements physiques ou virtuels, notamment pour des services personnalisés, est en pleine croissance. Ces besoins requièrent des moyens fiables pour reconnaitre la personne qui se présente au système d'accès. C'est ainsi que l'exploitation de caractéristiques ou mesures liées à l'individu est apparue naturellement comme la solution la plus fiable, chacune de ces différentes mesures est appelée "modalité biométrique".

Ce mémoire de fin d'études, nous a conduit à concevoir et à réaliser un système de vérification et d'identification par empreintes digitales, en implémentant l'approche basée minuties qui est très utilisée pour ses différents avantages, principalement ses bons résultats et sa simplicité de mise en œuvre.

Les outils de développement utilisés sont :

→ Visual Studio 2010 (C++)
→ Le Framework Qt
→ La bibliothèque de traitement d'image OpenCv

République Algérienne Démocratique et Populaire
Ministère de L'enseignement Supérieur et de la Recherche Scientifique
Université Mouloud MAMMERI Tizi-Ouzou
Faculté de Génie Electrique & d'Informatique
Département d'informatique

Mémoire

de fin d'études

En vue de l'obtention du diplôme de master recherche

Spécialité : Systèmes informatiques

Thème

Conception et réalisation d'un système automatique de reconnaissance d'empreintes digitales

Proposé et dirigé par :

M^r^ S. Redaoui

Réalisé par :

M^elle^ Djouaher Sabrina

M^elle^ Hocine Djamila

Promotion 2012

Remerciements

Nous tenons à présenter nos sincères remerciements à notre promoteur Mr S. Redaoui pour nous avoir encadrés et guidés tout au long de notre projet et pour tous les conseils judicieux qu'il nous a prodigués.

Aussi nous tenons à lui reconnaître le temps précieux qu'il nous a consacré.

Que les membres du jury trouvent ici nos très vifs remerciements pour avoir accepté d'honorer par leur jugement notre travail, ainsi que pour le temps qu'ils ont consacré pour nous donner leurs avis et corrections.

Nous leurs exprimons ici nos sincères sentiments.

Enfin, merci à tous ceux qui de près ou de loin ont contribué à la réalisation de ce projet : nos enseignants, nos chères familles et nos amis(es).

Sabrina et Djamila

Dédicaces

Je manquerai terriblement d'originalité en commençant par dédier ce travail à mes parents, mais j'ai une bonne excuse à cela : sans eux, mon existence n'aurait pas grand sens. Les mots me manquent pour les remercier comme il se doit.

A mes frères et sœurs dont le soutien sans faille m'a aidé à surmonter toutes les difficultés. A ma grande famille, à mes amis(es) et à tous mes enseignants.

Sabrina

Dédicaces

En signe de reconnaissance et de respect, je dédie ce modeste travail

Aux êtres les plus chers à mon cœur, mes parents, qui ont toujours cru

en moi et encouragée que dieu les protège

A mon cher grand frère Amar qui m'a toujours soutenu

A ma chère sœur et ses adorables petites filles

A mes frères Rachid et Keddour, en leurs souhaitant

beaucoup de réussite

A mes chères belles-soeurs Ouisa et Marie ainsi que leur famille

A tous mes oncles et tantes en particulier Fatima et leur famille

A mes cousins (es) en particulier ma chère Lilia

Djamila

Sommaire

CHAPITRE II : METHODES DE RECONNAISSANCE D'EMPREINTES DIGITALES

Chapitre III : Conception du système

Chapitre IV : Réalisation du système

Liste des figures

Chapitre I :

Chapitre II :

Chapitre III :

Chapitre IV :

Liste des tableaux

Chapitre I

INTRODUCTION GENERALE

INTRODUCTION GENERALE

De nos jours, une reconnaissance fiable des personnes est devenue un problème majeur pour des raisons de sécurité, notamment avec le développement des communications. Jusqu'à présent les méthodes usuelles de reconnaissance sont basées sur ce que l'on possède (carte d'identité, carte à puce, badge magnétique…) ou sur ce que l'on sait (mot de passe, code PIN…) mais ces méthodes posent de gros problèmes de fiabilités (falsification de document, oubli de son code, décryptage du mot de passe via des logiciels spécifiques…). Tous ces problèmes ont ainsi provoqué un développement accru des techniques biométriques, qu'elles soient morphologique (la reconnaissance d'empreinte digitale, la reconnaissance faciale...), comportementale (la reconnaissance vocale, l'identification de la démarche…) ou biologique (ADN, salive…).

Ces techniques sont préférées aux méthodes traditionnelles pour différentes raisons:
-La personne à reconnaitre doit être physiquement présente au point de reconnaissance, évitant ainsi la nécessité de se rappeler d'un mot de passe, ou de se servir d'un badge.
-Elles sont plus commodes pour l'utilisateur contrairement aux mots de passe et codes PIN, et peuvent potentiellement empêcher l'accès non autorisé ou l'utilisation frauduleuse de distributeurs de billets, de systèmes de gestion des horaires, de téléphones portables, de cartes à puce, de postes de travail et de réseaux informatiques.

Différents types de systèmes biométriques sont utilisés pour la reconnaissance en temps réel; l'un des plus populaires est basé sur la reconnaissance de l'empreinte digitale que nous allons traiter dans ce mémoire.

Ce mémoire est composé de quatre chapitres :
Le premier chapitre est consacré à la présentation de la biométrie. Dans le second, nous décrivons les caractéristiques ainsi que les différentes méthodes de reconnaissance d'empreintes digitales, le troisième chapitre est dédié à la conception de notre système et le dernier se rapporte à la réalisation d'une application de vérification / identification par empreintes digitales.

CHAPITRE I

Introduction à la biométrie

I.1 Introduction

Le besoin d'accès sécurisés et automatisés à des environnements physiques ou virtuels, notamment pour des services personnalisés, est en pleine croissance. Ces besoins requièrent des moyens fiables pour reconnaitre la personne qui se présente au système d'accès. C'est ainsi que l'exploitation de caractéristiques ou mesures liées à l'individu est apparue naturellement comme la solution la plus fiable, chacune de ces différentes mesures est appelée "modalité biométrique" [1] [2].

Dans ce chapitre, nous commençons par présenter la biométrie de manière générale ainsi que les diverses applications qui en découlent.

I.2 La biométrie

I.2.1 Définition

La biométrie ou mesure (metron) du vivant (bios) est, d'après le Petit Robert une « science qui étudie à l'aide des mathématiques les variations biologiques à l'intérieur d'un groupe déterminé ».

La biométrie consiste en l'analyse mathématique des caractéristiques biologiques d'une personne, et a pour objectif de déterminer son identité de manière irréfutable. Contrairement à ce que l'on sait ou ce que l'on possède la biométrie est basée sur ce que l'on est et permet ainsi d'éviter la duplication, le vol, l'oubli ou la perte. Les caractéristiques utilisées doivent être universelles (c'est-à-dire communes à tous les individus), uniques (pour pouvoir différencier deux individus), permanentes (c'est-à-dire invariantes dans le temps pour chaque individu), mesurable et enregistrable [25].

I.2.2 Les modes opératoires d'un système biométrique

Un système biométrique peut avoir deux modes opératoires [4]**:** la vérification et l'identification.

La vérification, également appelée authentification, consiste à confirmer ou infirmer l'identité d'une personne (suis-je celui que je prétends être ?). Il s'agit d'une comparaison du type un contre un ; les caractéristiques de l'individu sont comparées à celles présentes dans un enregistrement de référence.

Quant à l'identification, elle permet d'établir l'identité d'une personne (qui suis-je ?) à partir d'une base de données, il s'agit d'une comparaison du type un contre plusieurs.

I.2.3 Les différentes techniques biométriques

Parmi les différentes techniques biométriques existantes, on distingue deux grandes catégories :

I.2.3.1 L'analyse morphologique (physiologique)

Elle est basée sur l'identification de traits physiques particuliers qui, pour toute personne, sont uniques et permanents. Cette catégorie regroupe les empreintes digitales, l'iris de l'œil, la forme de la main, les traits du visage, le réseau veineux de la rétine, les veines de la main, etc.

I.2.3.2 L'analyse comportementale

Elle se base sur l'analyse de certains comportements d'une personne. Cette catégorie regroupe la reconnaissance vocale, la dynamique de frappe au clavier, la dynamique de la signature, l'analyse de la démarche, etc.

Il existe, par ailleurs, une autre catégorie qui est l'étude des traces biologiques telles que : l'ADN, le sang, la salive, l'urine, l'odeur, etc.

Figure I.1 : Les différentes modalités biométriques

I.2.4 Présentation des techniques biométriques

I.2.4.1 L'empreinte digitale [5]

Une empreinte digitale est constituée d'un ensemble de lignes localement parallèles formant un motif unique pour chaque individu. On distingue les stries (ou crêtes, ce sont les lignes en contact avec une surface au toucher) et les sillons (ce sont les creux entre deux stries). Les stries contiennent en leur centre un ensemble de pores régulièrement espacés.
Chaque empreinte possède un ensemble de points singuliers globaux (les centres et les deltas) et locaux (les minuties). Les centres correspondent à des lieux de convergence des stries tandis que les deltas correspondent à des lieux de divergence. L'acquisition des données est faite par un capteur électronique de type optique, thermique, capacitif ou à ultrasons.

Figure I.2 : Capture de l'image d'une empreinte digitale

I.2.4.2 L'iris [6] [7]

L'iris est la zone colorée visible entre le blanc de l'œil et la pupille. Il s'agit d'un réseau de tubes fins dont le diamètre est inférieur à celui d'un cheveu. Ces structures se développent dès le plus jeune âge et sont considérées comme uniques, même chez les vrais jumeaux, et resteront constantes jusqu'à la mort.

C'est une technique extrêmement fiable car il contient une infinité de points caractéristiques (ensemble fractal), la fraude étant néanmoins possible en utilisant des lentilles. L'acquisition de l'iris est effectuée au moyen d'une caméra pour pallier aux mouvements inévitables de la pupille. Elle est très sensible (précision, reflet…) et relativement désagréable pour l'utilisateur car l'œil doit rester grand ouvert et il est éclairé par une source lumineuse pour assurer un contraste correct.

Figure I.3 : Capture de l'image d'un iris

I.2.4.3 La géométrie de la main [8]

Jusqu'à 90 caractéristiques de la main sont mesurées (forme de la main et des articulations, longueur et largeur des doigts, longueur inter articulations…). Le taux d'erreurs dans la reconnaissance est assez élevé, en particulier pour des personnes appartenant à une même famille en raison d'une forte ressemblance. De plus, la forme de la main évolue beaucoup avec l'âge.

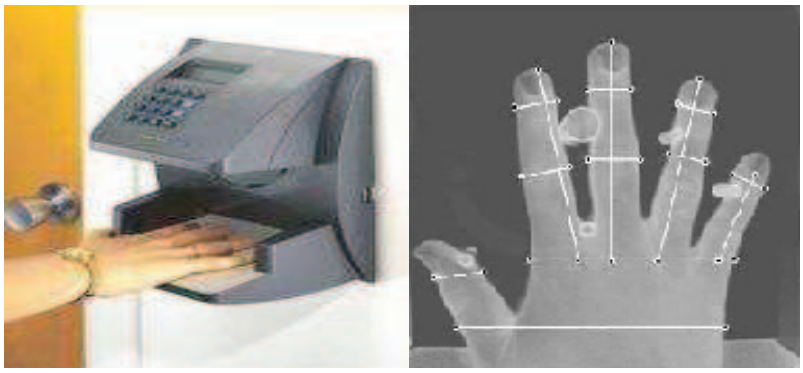


Figure I.4 : Scan de la forme de la main

I.2.4.4 Le visage [9] [10]

Plusieurs parties du visage (joues, yeux, nez, bouche…) sont extraites d'une photo ou d'une vidéo et analysées géométriquement (distance entre différents points, positions, formes…). Le problème de cette méthode vient des possibles perturbations pouvant transformer le visage (maquillage, faible luminosité, présence d'une barbe ou d'une lunette, expression faciale inhabituelle, changement avec l'âge, etc.).

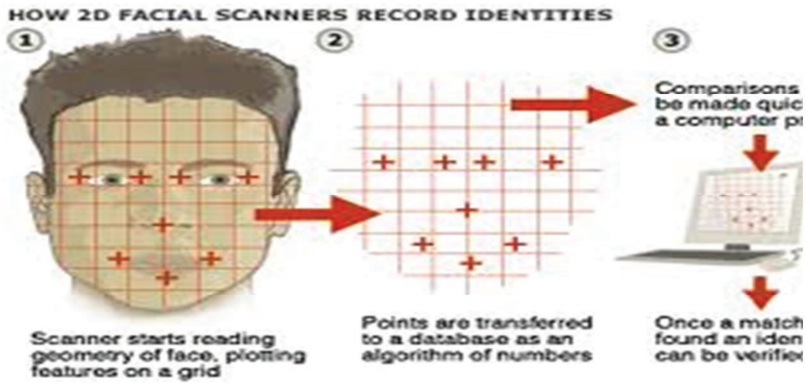


Figure I.5 : Exemple d'un système de reconnaissance facial

I.2.4.5 La rétine [11]

C'est une membrane du fond de l'œil tapissant la choroïde, sensible à la lumière. Cette technique se base sur le fait que les vaisseaux sanguins d'une rétine sont uniques pour chaque personne. L'utilisateur doit placer son œil face à un orifice de capture situé sur le dispositif d'acquisition. Un faisceau lumineux traverse l'œil jusqu'aux vaisseaux sanguins

capillaires de la rétine. Le système localise et capture ainsi environ 400 points de référence. Elle requiert une collaboration étroite de la part de l'utilisateur, car il doit placer son œil extrêmement près de la caméra.

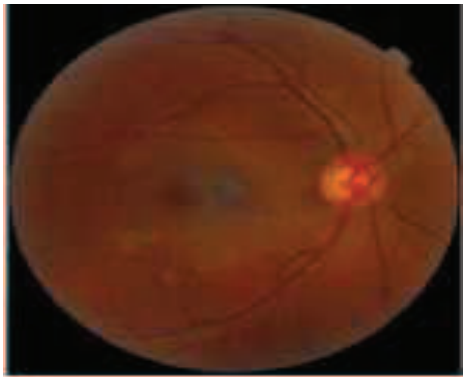


Figure I.6 : Rétine de l'œil

I.2.4.6 Les veines de la main [12]

L'utilisateur place sa main dans une chambre ou un gabarit de lecture. Les caractéristiques des veines sont lues par une caméra infrarouge qui en tire une image en deux dimensions. Cette image est ensuite digitalisée et enregistrée pour comparaison future.

Figure I.7 : Scan des veines de la main

I.2.4.7 La reconnaissance vocale [13] [14]

Les caractéristiques du timbre de la voix et de la prononciation sont analysées. La qualité de l'enregistrement peut poser problème et il est ainsi possible de frauder avec un échantillon vocal préenregistré.

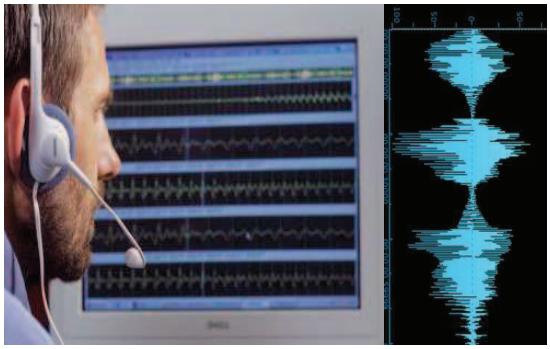


Figure I.8 : Capture de la voix

I.2.4.8 La dynamique de frappe au clavier [15]

Un système basé sur la dynamique de frappe au clavier ne nécessite aucun équipement particulier, chaque ordinateur disposant d'un clavier. Il s'agit d'un dispositif logiciel qui calcule le temps où un doigt effectue une pression sur une touche et le temps où un doigt est dans les airs (entre les frappes). Cette mesure est capturée environ mille fois par seconde. La séquence de frappe est prédéterminée sous la forme d'un mot de passe. Initialement l'utilisateur doit composer son mot de passe à quelques reprises afin que soit constitué un gabarit de référence.

Ce dispositif biométrique est utilisé comme méthode de vérification pour le commerce électronique et comme mécanisme de contrôle d'accès à des bases de données.

Figure I.9 : Dynamique de frappe au clavier

I.2.4.9 La dynamique du tracé de la signature [16]

Il s'agit d'une analyse comportementale où différents éléments (mesure da la vitesse, ordre d'écriture, pression exercée, accélération…) sont mesurés lors de la signature. La falsification est possible en passant par une phase d'apprentissage, la signature peut varier selon le stress de l'utilisateur.

Figure I.10 : Capture d'une signature

I.2.4.10 L'analyse de la démarche [17]

Il s'agit de reconnaître un individu par sa façon de marcher et de bouger (vitesse, accélération, mouvements du corps…), en analysant des séquences d'images. La démarche serait en effet étroitement associée à la musculature naturelle et donc très personnelle. Son inconvénient majeur est qu'elle est sensible aux changements d'habits, chaussures et surface. Ceci rend cette approche limitée au monde de la recherche seulement.

I.2.5 Comparaison des différentes techniques (les plus utilisées)

Il n'y a pas de système biométrique parfait. L'International Biometric Group a procédé à une comparaison des différentes technologies sur la base de 4 critères :

- l'intrusivité : décrit dans quelle mesure l'utilisateur perçoit le contrôle comme étant intrusive
- l'efficacité: efficacité de la méthode (capacité à identifier quelqu'un)
- le coût : coût de la technologie (lecteur, capteur…)
- l'effort : effort requis pour l'utilisateur lors de la mesure

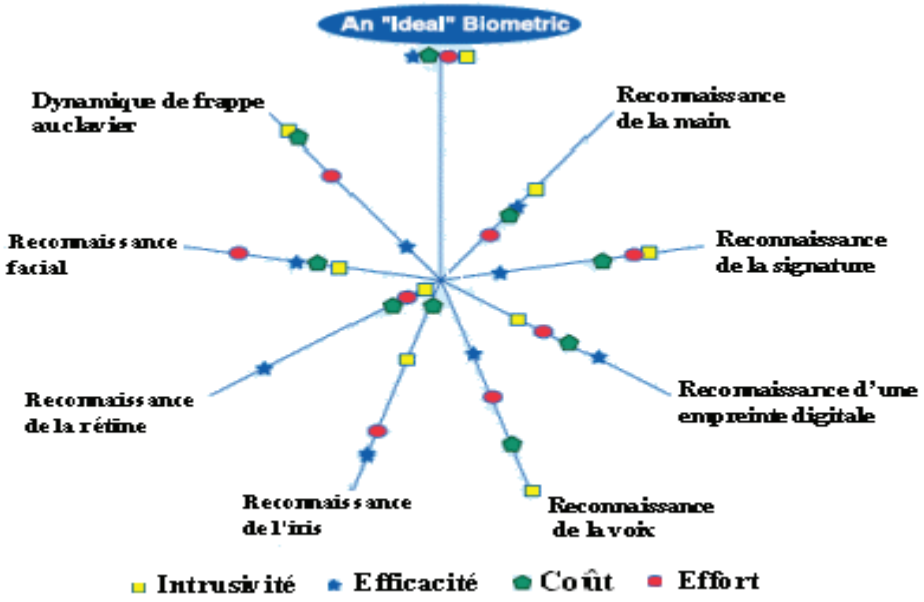


Figure I.11 : Représentation comparative des différentes techniques biométriques utilisées [3]

Cette comparaison permet de choisir une technologie en fonction des contraintes liées à l'application.

Le CLUSIF[1] a également proposé une comparaison (avantages / inconvénients) des principales technologies biométriques.

Techniques	Avantages	Inconvénients
Empreintes digitales	Coût, ergonomie moyenne, facilité de mise en place, taille du capteur	Qualité optimale des appareils de mesure (fiabilité), acceptabilité moyenne, possibilité d'attaques (rémanence de l'empreinte,...)
Forme de la main	Très ergonomique, bonne acceptabilité	Système encombrant, coût, perturbation possible par des blessures et l'authentification des membres d'une même famille
Visage	Coût, peu encombrant, bonne acceptabilité	Jumeaux, psychologie, religion, déguisement, vulnérabilité aux attaques
Rétine	Fiabilité, pérennité	Coût, acceptabilité faible, installation difficile
Iris	Fiabilité	Acceptabilité très faible, contrainte d'éclairage
Voix	Facilité	Vulnérable aux attaques
Signature	Ergonomie	Dépendant de l'état émotionnel de la personne, fiabilité
Frappe au clavier	Ergonomie	Dépendant de l'état physique de la personne

Tableau I.1 : avantages / inconvénients des principales technologies biométriques. [24]

[1] Club de la sécurité de l'information français

I.2.6 Applications de la biométrie

Les applications de la biométrie peuvent être divisées selon quatre groupes : service public, application de la loi, transaction commerciale et bancaire, accès physique et logique.

Figure I.12 : Applications biométriques [26]

I.2.6.1 Service public

-La biométrie est fréquemment utilisée par les services d'immigration pour contrôler automatiquement l'identité des personnes entrantes ou sortantes d'un territoire, ainsi l'iris, le visage et l'empreinte digitale sont à l'essai dans de nombreux aéroports.

-De même en santé publique, la biométrie serait utile pour supprimer les cartes d'assurance sociale, ou du moins vérifier l'identité de leur propriétaire.

I.2.6.2 Applications de la loi

- L'authentification de criminels par reconnaissance automatique de leurs empreintes digitales a montré son efficacité, cette pratique se mondialise.
- Le suivi de prisonniers à domicile est déjà assuré par des systèmes de vérification de la voix dans certains états des Etats-Unis.
- L'identification d'enfants kidnappés ou disparus, dont la véritable identité a été masquée en utilisant l'ADN.
- Protection électronique de documents.

I.2.6.3 Transaction commerciale et bancaire

-Apparition de machines de retrait automatique d'argents disposant d'un système de vérification d'individu.

I.2.6.4 Accès physique et logique

On parle de contrôle d'accès physique lorsqu'on cherche à sécuriser l'accès à un lieu (entrée d'un bâtiment), alors que le contrôle d'accès logique concerne l'accès informatique à un terminal, serveur et réseau informatique ou de télécommunication (ex : ordinateur, téléphone portable, base de données privée).

I.2.7 Le marché mondial de la biométrie

Régulièrement, un rapport sur le marché de la biométrie est édité par **IBG** (International Biometric Group). Cette étude est une analyse complète des chiffres d'affaires, des tendances de croissance, et des développements industriels pour le marché de la biométrie actuel et futur. La lecture de ce rapport est essentielle pour des établissements déployant la technologie biométrique, les investisseurs dans les entreprises biométriques, ou les développeurs de solutions biométriques.

Le chiffre d'affaires de l'industrie biométrique incluant les applications judiciaires et celles du secteur public, se développe rapidement. Une grande partie de la croissance sera attribuable au contrôle d'accès aux systèmes d'information (ordinateur / réseau) et au commerce électronique, bien que les applications du secteur public continuent à être une partie essentielle de l'industrie.

On prévoit que le chiffre d'affaires des marchés émergents (accès aux systèmes d'information, commerce électronique et téléphonie, accès physique, et surveillance) dépasse le chiffre d'affaires des secteurs plus matures (identification criminelle et identification des citoyens).

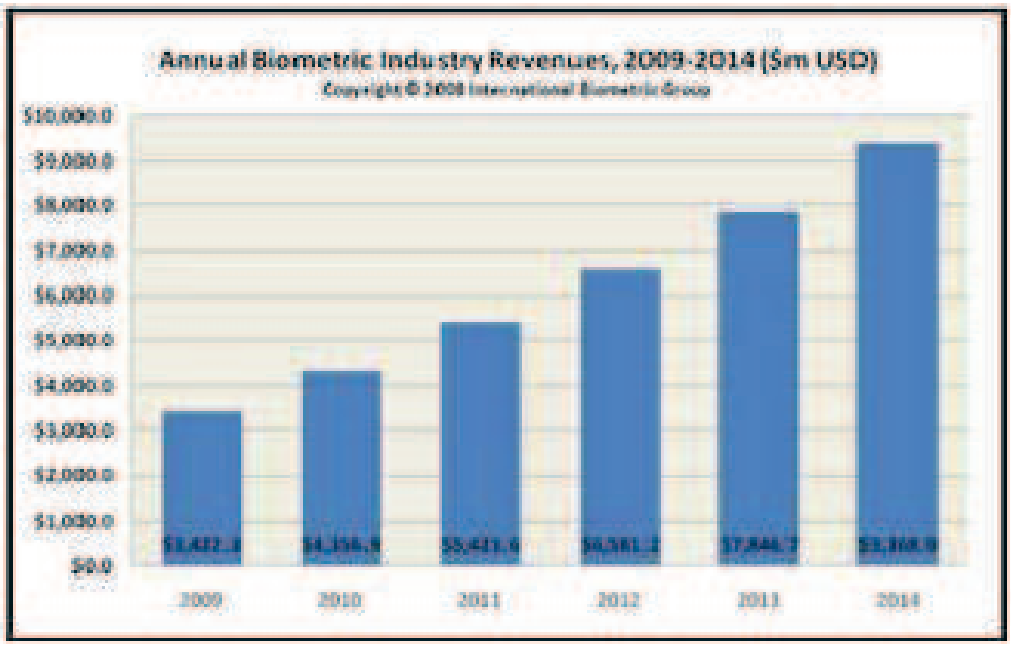


Figure I.13 : Evolution du marché international de la biométrie [3]

I.2.8 Les parts de marché par technologie

Les empreintes digitales continuent à être la principale technologie biométrique en termes de part de marché, près de 50% du chiffre d'affaires total (hors applications judiciaires). La reconnaissance du visage, avec 12% du marché (hors applications judiciaires), dépassant la reconnaissance de la main, qui avait avant la deuxième place en termes de source de revenus après les empreintes digitales.

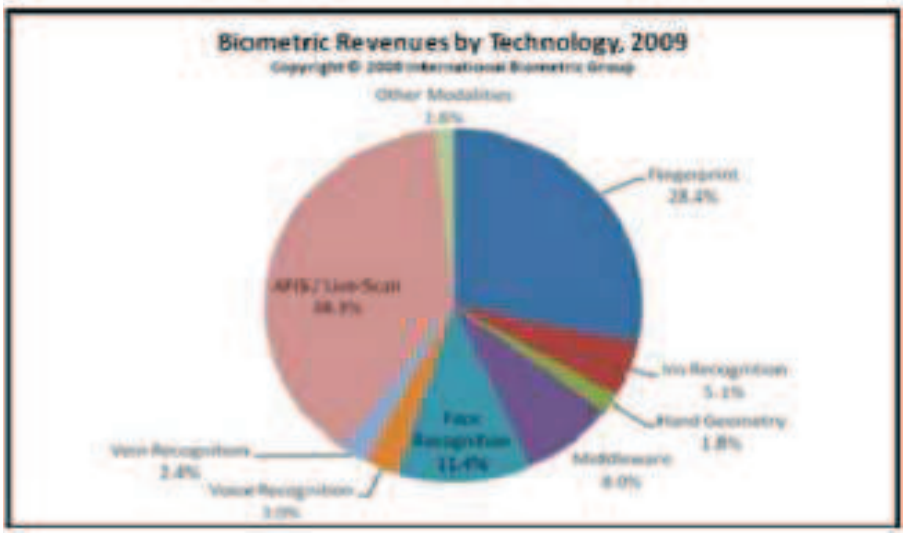


Figure I.14 : Parts de marché des différentes technologies biométriques [3]

I.3 Architecture d'un système biométrique

Il existe toujours au moins deux modules dans un système biométrique: le module d'apprentissage et celui de reconnaissance [18] [19]. Le troisième module (facultatif) est le module d'*adaptation*. Pendant l'apprentissage, le système va acquérir une ou plusieurs mesures biométriques qui serviront à construire un modèle de l'individu. Ce modèle de référence servira de point de comparaison lors de la reconnaissance. Le modèle pourra être réévalué après chaque utilisation grâce au module d'adaptation.

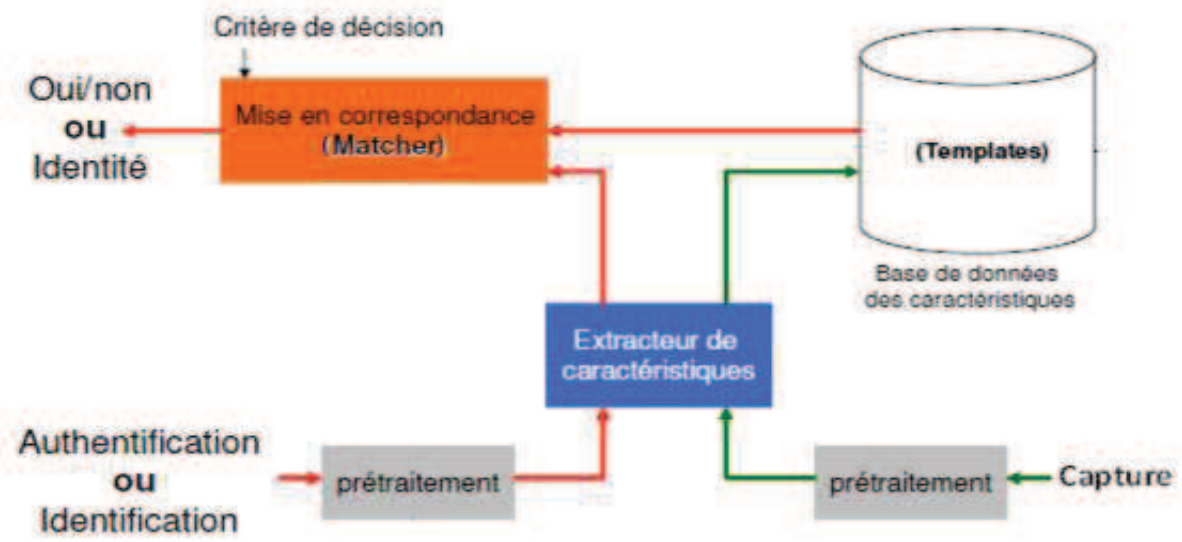


Figure I.15 : Architecture d'un système de reconnaissance biométrique [27]

I.3.1 Module d'apprentissage

Au cours de l'apprentissage, la caractéristique biométrique est tout d'abord mesurée grâce à un capteur; on parle d'acquisition ou de capture. En général, cette capture n'est pas directement stockée et des transformations lui sont appliquées. En effet, le signal contient de l'information inutile à la reconnaissance et seuls les paramètres pertinents sont extraits. Le modèle est une représentation compacte du signal qui permet de faciliter la phase de reconnaissance, mais aussi de diminuer la quantité de données à stocker. Il est à noter que la qualité du capteur peut grandement influencer les performances du système. Meilleure est la qualité du système d'acquisition, moins il y aura de prétraitements à effectuer pour extraire les paramètres du signal.

Cependant, les capteurs de qualité sont en général coûteux et leur utilisation est donc limitée à des applications de haute sécurité pour un public restreint. Le modèle peut être stocké dans une base de données comme représenté sur la figure I.15 ou sur une carte de type carte à puce.

I.3.2 Module de reconnaissance

Au cours de la reconnaissance, la caractéristique biométrique est mesurée et un ensemble de paramètres est extrait comme lors de l'apprentissage. Le capteur utilisé doit avoir des propriétés aussi proches que possibles du capteur utilisé durant la phase d'apprentissage. Si les deux capteurs ont des propriétés trop différentes, il faudra en général appliquer une série de prétraitements supplémentaires pour limiter la dégradation des performances.
La suite de la reconnaissance sera différente suivant le mode opératoire du système: identification ou vérification.

En mode identification, le système doit deviner l'identité de la personne. Il répond donc à une question de type : « Qui suis-je ? ». Dans ce mode, le système compare le signal mesuré avec les différents modèles contenus dans la base de données (problème de type 1 : N). En général, lorsque l'on parle d'identification, on suppose que le problème est fermé, c'est-à-dire que toute personne qui utilise le système possède un modèle dans la base de données.

En mode vérification, le système doit répondre à une question de type : « Suis-je bien la personne que je prétends être ? ». L'utilisateur propose une identité au système et le système doit vérifier que l'identité de l'individu est bien celle proposée. Il suffit donc de comparer le signal avec un seul des modèles présents dans la base de données (problème de type 1 : 1). En mode vérification, on parle de problème ouvert puisque l'on suppose qu'un individu qui n'a pas de modèle dans la base de données (imposteur) peut chercher à être reconnu. Identification et vérification sont donc deux problèmes différents.

L'identification peut-être une tâche redoutable lorsque la base de données contient des milliers, voire des millions d'identités, tout particulièrement lorsqu'il existe des contraintes de type « temps réel » sur le système.

I.3.3 Module d'adaptation

Pendant la phase d'apprentissage, le système biométrique ne capture souvent que quelques instances d'un même attribut afin de limiter la gêne pour l'utilisateur. Il est donc difficile de construire un modèle assez général capable de décrire toutes les variations possibles de cet attribut. De plus, les caractéristiques de cette biométrie ainsi que ses conditions d'acquisition peuvent varier. L'adaptation est donc nécessaire pour maintenir voire améliorer la performance d'un système utilisation après utilisation. [20] [21].

I.4 Evaluation d'un système biométrique

La performance d'un système de reconnaissance peut se mesurer principalement à l'aide de trois critères : sa précision, son efficacité (vitesse d'exécution) et le volume de données qui doit être stocké pour chaque locuteur. Nous nous concentrons au premier aspect qui est la précision.

I.4.1 Evaluation de l'identification

Le taux d'identification est la mesure la plus couramment utilisée mais il n'est pas toujours suffisant. En effet, en cas d'erreur, il peut être utile de savoir si le bon choix se trouve dans les N premiers. On trace alors le score cumulé (cumulative match score) qui représente la probabilité que le bon choix se trouve parmi les N premiers [22].

Le type d'erreurs commises par ce genre de système est d'attribué à l'individu présenté, une identité autre que la sienne. Les performances de ce système sont mesurées à l'aide du taux d'identification [23].

$$\text{Taux d'identification} = \frac{nombre\ de\ tests\ ayant\ conduit\ \text{à}\ une\ identification\ correcte}{\text{nombre total de tests}}$$

Ce paramètre dépend du nombre de personnes contenues dans la base de données. En effet, plus la base est volumineuse (nombre de tests important), plus le taux d'erreurs risque d'être grand.

I.4.2 Evaluation de la vérification

La vérification est un problème qui peut être formulé de la manière suivante : Soient H_0 l'hypothèse : « La capture C provient d'un imposteur » et H_1 l'hypothèse : « La capture C provient de l'utilisateur légitime ».

Il faut donc choisir l'hypothèse la plus probable. On considère que la capture C provient d'un utilisateur légitime si P (H1/C) > P (H0/C). En appliquant la loi de Bayes [4], on obtient :

$$\frac{P(\frac{C}{H1})P(H1)}{P(C)} > \frac{P\left(\frac{C}{H0}\right)P(H0)}{P(C)}$$

Et donc :

$$\frac{P(\frac{C}{H1})}{P(C/H0)} > \frac{P\left(\frac{C}{H0}\right)}{P(H1)}$$

Le taux de vraisemblance (likelihood ratio) $S = \mathrm{P}(\frac{\mathrm{C}}{\mathrm{H1}})/\mathrm{P}(\frac{\mathrm{C}}{\mathrm{H0}})$ est comparé à un seuil appelé seuil de décision. Lors de la vérification, le taux S est comparé au seuil de décision θ, si S est inférieur à θ alors l'individu est rejeté sinon celui-ci est accepté.

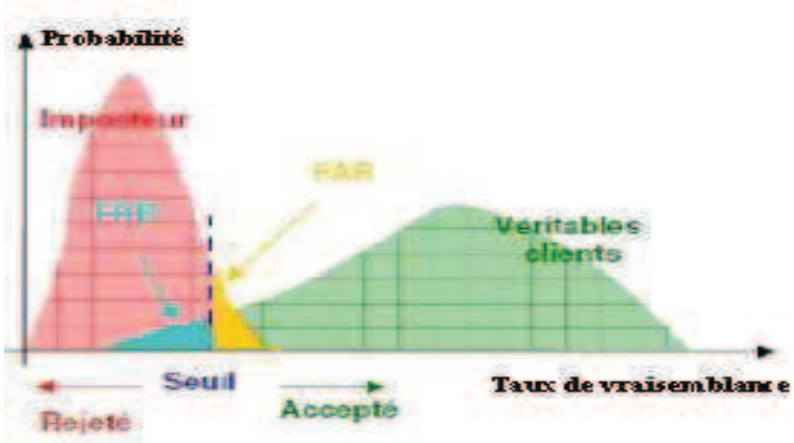


Figure I.16: Distributions des taux de vraisemblance des clients légitimes et des imposteurs d'un système biométrique [27]

Le système peut faire deux types d'erreurs :

- **FA (False Acceptance) :**

La fausse acceptation correspond au cas où le système accepte un individu qui a proclamé une identité qui n'est pas la sienne.

- **FR (False Rejection) :**

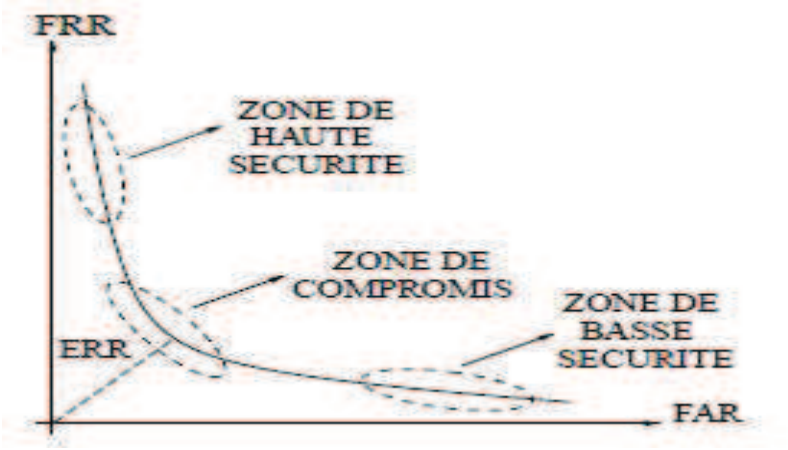
Le faux rejet correspond au cas où le système rejette un client légitime.

Les performances de ce type de système se basent principalement sur le taux de faux rejet et le taux de fausse acceptation.

$$FAR(taux\ de\ fausse\ acceptation) = \frac{nombre\ de\ fausses\ acceptation}{\text{nombre d'imposteur presentés}}$$

$$FRR(taux\ de\ faux\ rejets) = \frac{nombre\ de\ faux\ rejets}{\text{nombre de clients presentés}}$$

On remarque à travers la figure I.16 que plus le seuil de décision est petit, plus le système acceptera de clients légitimes mais aussi d'imposteurs, plus le seuil est grand, plus le système rejettera d'imposteurs mais aussi des utilisateurs légitimes. Le paramétrage d'un système consiste à trouver le bon équilibre entre ces deux taux. Les performances d'un système biométrique peuvent être présentées graphiquement à l'aide de la courbe ROC (Receiver Operating Characteristics) sur laquelle les FRRs sont données en fonction des FARs (voir Figure I.17). Cette courbe est obtenue en calculant un couple (FAR, FRR) pour chaque valeur du seuil de décision, ce dernier varie de la plus petite valeur des taux obtenus en phase de test à la plus grande valeur.



FigureI.17 : Courbe ROC

Il existe d'autres critères fréquemment utilisés pour donner un aperçu des performances des systèmes de vérification :

- **EER (Equal Error Rate) :**

Le taux égal d'erreur correspond à l'intersection de la courbe ROC avec la première bissectrice, en d'autres termes, EER correspond au point de fonctionnement pour lequel le taux de faux rejet est égal au taux de fausse acceptation.

- **HTER (Half Total Error Rate) :** représente la moyenne du FAR et FRR.

$$HTER = \frac{\text{FAR} + \text{FRR}}{2}$$

- **TER (Total Error Rate) :**

Le taux d'erreur global correspond au taux d'erreur total (faux rejet et fausse acceptation).

- Choix du seuil de decision ?

Un système de « *haute sécurité* » laissera difficilement entrer un imposteur, donc il y aura un FAR très faible. Mais les véritables clients auront eux aussi du mal à entrer, et le FRR sera élevé.

Un système « *permissif* », réglé sur une sécurité faible, laissera facilement entrer les véritables clients (le FRR sera faible), mais aussi les imposteurs, et le FAR sera relativement élevé. [27]

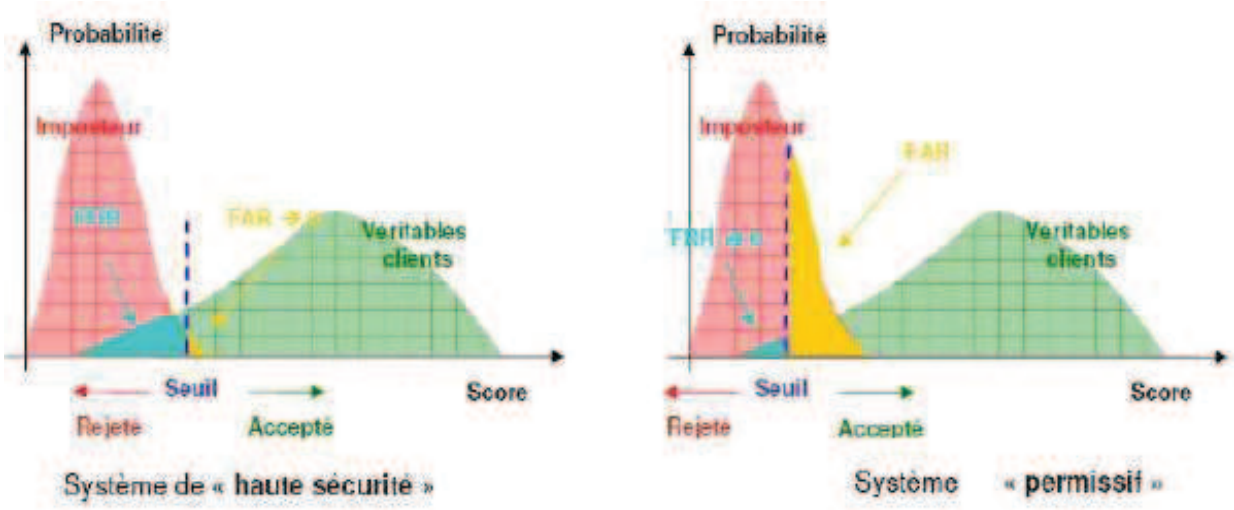


Figure I.18: Courbe représentant les types de systèmes [27]

I.5 Conclusion

Actuellement, la biométrie représente le moyen le plus sûr pour la sécurité dans tous les domaines de la vie. Elle est de plus en plus utilisée dans la vie de tous les jours grâce à ses avantages. Cependant, la biométrie à ses limites malgré les prix des équipements qui sont de plus en plus abordables, elle reste aujourd'hui accessible à un nombre de personnes limité. Aussi, la biométrie n'est pas une science exacte, car elle reste dépendante de la qualité des capteurs, et du traitement de celle-ci.

Dans ce chapitre, nous avons présenté une vue générale de la biométrie, un survol sur quelques techniques ainsi que l'architecture d'un système biométrique. La présentation des méthodes d'extraction des caractéristiques des empreintes digitales fera l'objet du prochain chapitre.

CHAPITRE II

Méthodes de reconnaissance d'empreintes

II.1 Introduction

De nombreuses méthodes de reconnaissance d'empreintes digitales existent, se basant les unes et les autres sur les différentes caractéristiques de celles-ci.

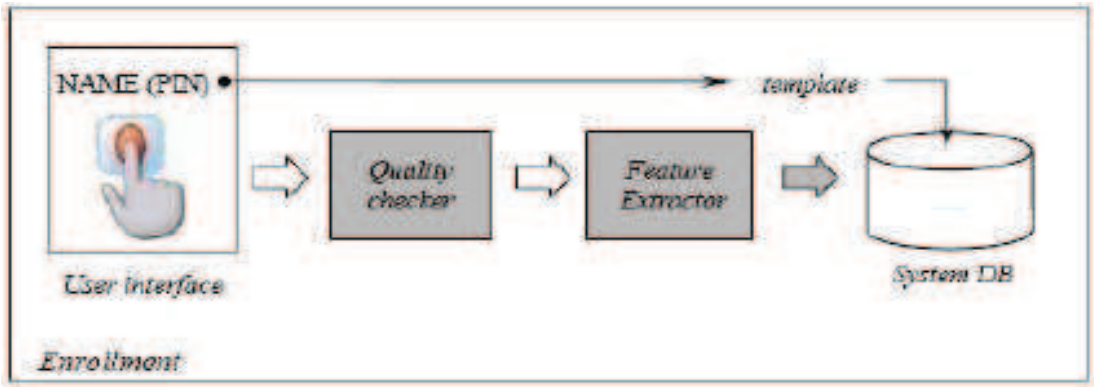
Même si le principe de fonctionnement diffère d'une méthode à une autre, toutes sont cependant basées sur deux principales étapes :

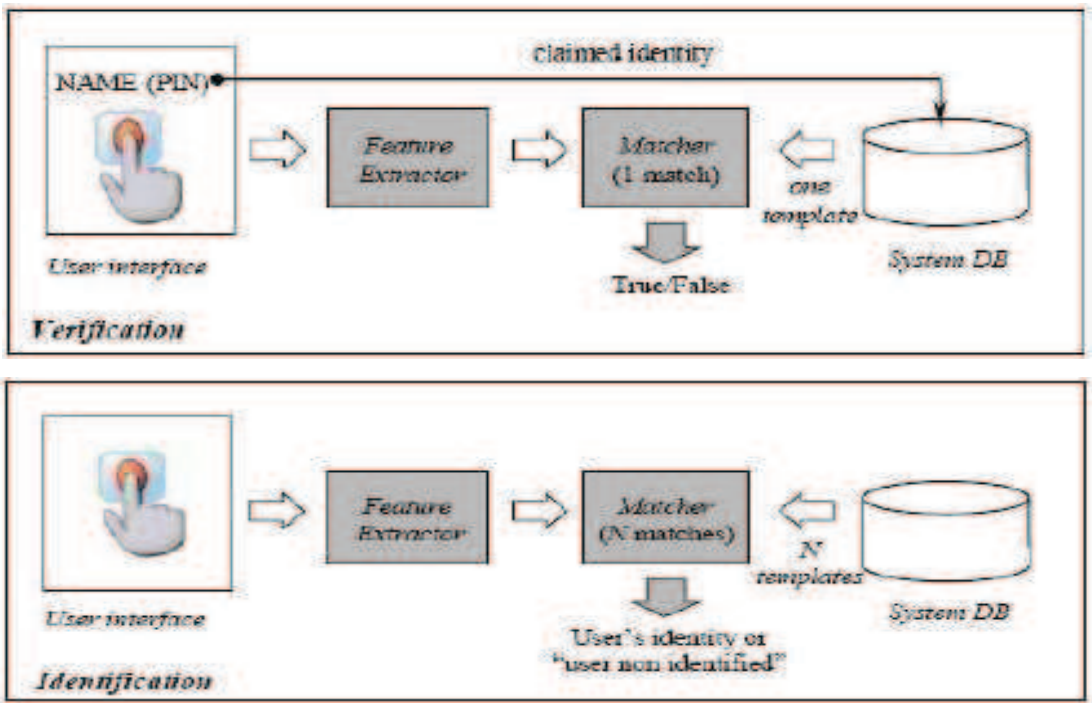
1. Extraction des caractéristiques.
2. Appariement.

II.2 Fonctionnement général d'un système biométrique d'empreintes digitales

Un système biométrique est essentiellement un système de reconnaissance de formes qui fonctionne en acquérant des données biométriques à partir d'un individu, extrayant un ensemble de caractéristiques à partir des données acquises, et comparant ces caractéristiques contre la signature dans la base de données. Selon le contexte d'application, un système biométrique peut fonctionner en mode de vérification ou mode d'identification

En général, tous les systèmes biométriques partagent le même schéma de fonctionnement. Tous se composent des deux processus suivant :





FigureII.1 : Schéma de fonctionnement d'un système biométrique d'empreintes digitales [28]

- **Processus d'enrôlement (enregistrement) :** Ce processus a pour but d'enregistrer les caractéristiques des utilisateurs dans la base de données.
- **Processus d'identification / vérification :** Ce processus est effectué lorsqu'une personne enregistrée dans la base de données biométriques doit s'identifier ou s'authentifier.

II.3 Empreintes digitales

II.3.1 Historique [40]

Les premières traces d'utilisation d'empreintes digitales ont été découvertes en Egypte et datent de l'époque des pyramides il y a plus de 4000 ans. Les Chinois ont aussi utilisé très tôt ce moyen pour signer les documents officiels (le plus vieux document signé date du troisième siècle avant Jésus Christ) mais ils ne savaient surement pas que les empreintes étaient uniques pour chaque personne et permettaient ainsi une identification fiable. C'est en 1856 que l'anglais William Hershel, après avoir utilisé les empreintes en guise de signature sur la population indienne qu'il dirigeait, commença à comprendre que les empreintes étaient uniques et constante dans le temps. En 1888, le britannique Francis Galton publia une étude sur les empreintes digitales où il établit leurs caractéristiques (unicité, invariance, minuties,

classification…) et en 1901 la technique d'identification au moyen des empreintes fut adoptée officiellement en Angleterre dans le système judiciaire. Cette technique fut ensuite largement développée dans les enquêtes criminelles et permit de résoudre un bon nombre d'affaires. De nos jours les empreintes sont toujours largement utilisées et reconnues comme méthode d'identification fiable.

II.3.2 Qu'est-ce qu'une empreinte digitale ?

C'est le dessin formé par les lignes de la peau des doigts, des paumes des mains, des orteils ou de la plante des pieds (les dermatoglyphes), et qui se forme durant la période fœtale. Ces lignes sont appelées *stries* (ou crêtes, ce sont les lignes en contact avec une surface au toucher) et les espaces entre celles-ci *vallées*. Les stries contiennent en leur centre un ensemble de *pores* régulièrement espacés.

Une empreinte digitale est habituellement le dépôt de sébum laissé par le contact du doigt avec une surface. Elle peut toutefois également être causée par des substances présentes sur la surface, comme des pigments ou d'autres substances colorées ou malléables. On l'obtient par exemple en appliquant de l'encre sur les doigts puis en appuyant les doigts ainsi enduits sur un support [29].

II.3.3 Classification des empreintes digitales

En regardant la structure globale d'une empreinte digitale, on remarque que dans certaines régions les dessins de stries adoptent certaines formes distinctes. Ces régions s'appellent des régions caractéristiques ou régions singulières ; elles sont classées en trois grandes catégories : boucle, delta et spires ou tourbillon. Les régions singulières sont utilisées dans la classification des empreintes digitales [26].

La première classification rigoureuse a été introduite en 1982 par Sir Francis Galton [30], qui a défini trois grandes classes d'empreintes digitales : boucle, arche et spires. Cette dernière a été élargie par Edward Henry [31] en subdivisant les trois classes en sous-classes selon la topographie générale de l'empreinte digitale: boucle à gauche, boucle à droite, arche, arche penchée, spires et spires imbriquées ou boucles jumelles (voir Figure II.2). Une autre caractéristique globale de l'empreinte utilisée dans certains algorithmes de comparaison d'empreintes digitales est le cœur de l'empreinte. C'est le point supérieur de la ligne la plus au centre de l'empreinte.

- Une empreinte digitale d'**arch** a les rides qui entrent d'un côté, se lèvent à une petite bosse, et sortent le côté opposé dont elles sont entrées. Les archs n'ont pas de boucles ou de deltas;
- Une empreinte digitale de **tented arch** est semblable à une empreinte digitale d'arch, sauf qu'au moins une ride montre une courbure élevée et il y a une boucle et un delta.
- Une empreinte digitale de **loop** a une ou plusieurs rides qui entrent d'un côté, courbent en arrière, et sortent du même côté qu'elles sont entrées. Une boucle et un delta sont présents. Des loops peuvent être encore subdivisées: les loops qui ont des rides qui entrent et sortent du côté gauche s'appellent le left loop et les loops qui ont des rides qui entrent et sortent du côté droite s'appellent le right loop.
- Une empreinte digitale de **whorl** contient au moins une ride qui fait un chemin 360-dégrée complet autour du centre de l'empreinte digitale. Deux boucles et deux deltas peuvent être trouvés dans des empreintes digitales de whorl;

La figureII.2 montre des exemples d'empreintes digitales de chaque classe :

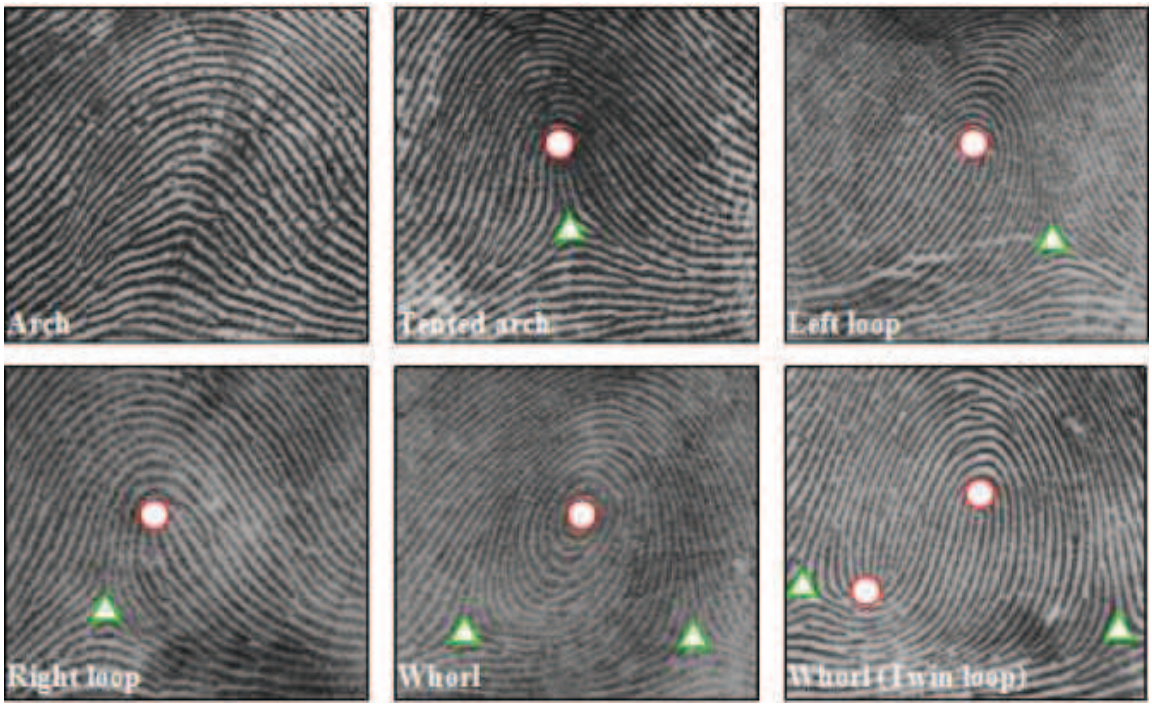


Figure II.2 : Les principales classes d'empreintes digitales selon la classification Galton-Henry [26]

De haut en bas, de gauche à droite nous avons les classes : arche, arche penchée, boucle à gauche, boucle à droite, spires et les boucles jumelles.

On voit aussi les régions singulières delta (triangles verts) et boucle (voisinage des cercles rouges). Les caractéristiques globales d'une empreinte sont donc les régions singulières et le cœur de l'empreinte.

II.3.4 Définition des minuties

Les empreintes digitales possèdent également des traits caractéristiques locaux appelés minuties (littéralement : petits détails). Dans la biométrie des empreintes digitales, ce terme désigne les différentes discontinuités des lignes d'une empreinte.

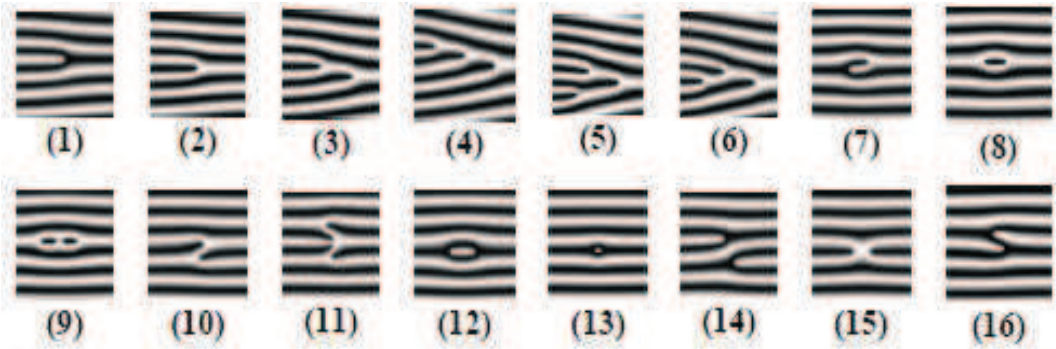
L'analyse des empreintes relève que les crêtes exposent différents types : la crête de bifurcation, le noyau, le lac, le delta, l'île, et la fin de ligne…

FigureII.3: Caractéristiques d'une empreinte digitale

Les centres correspondent à des lieux de convergences des stries tandis que les deltas correspondent à des lieux de divergence. Une étude (FigII.4) a montré l'existence de seize types de minuties différentes mais en général les algorithmes ne s'intéressent qu'aux *bifurcations* et *terminaisons*. L'empreinte est normalement représentée par 70-100 minuties.

En effet, une strie peut bifurquer (la minutie s'appelle alors la bifurcation), s'arrêter soudainement (on parle alors de la terminaison), se déchirer au milieu pour former une sorte de trou (c'est le lac)...

L'institut américain qui supervise le développement de normes ANSI (American National Standard Institute) a proposé un standard de 4 types : les terminaisons, les bifurcations, les trifurcations et les sous-terminaisons [31].



1.	terminaison	9.	boucle double
2.	bifurcation simple	10.	pont simple
3.	bifurcation double	11.	pont jumeau
4.	bifurcation triple I	12.	intervalle
5.	bifurcation triple II	13.	point isolé
6.	bifurcation triple III	14.	traversée
7.	crochet	15.	croisement
8.	boucle simple	16.	tête bêche

FigureII.4: Les différents types de minuties

Une empreinte complète contient en moyenne une centaine de points caractéristiques, mais les contrôles ne sont effectués qu'à partir de 12 points. Il est quasiment impossible de trouver deux individus présentant 12 points caractéristiques identiques, même dans une population de plusieurs millions de personnes.

Le caractère quasi-unique d'une empreinte digitale en fait un outil biométrique très utilisé pour l'identification des individus en médecine légale et pour la police scientifique.

En effet, la probabilité pour que deux personnes aient la même empreinte digitale est de $1/10^{24}$, ce qui est très faible à l'échelle de la population humaine, et donc quasiment impossible. De plus, son caractère aléatoire s'affranchit des risques de ressemblances entre individus partageant un même patrimoine génétique : des individus homozygotes comme des jumeaux ou des triplés par exemple auront chacun un jeu d'empreintes digitales qui leur sera propre et différent de celui des autres individus de la même fratrie. [24]

II.4 Acquisition d'empreintes digitales [32]

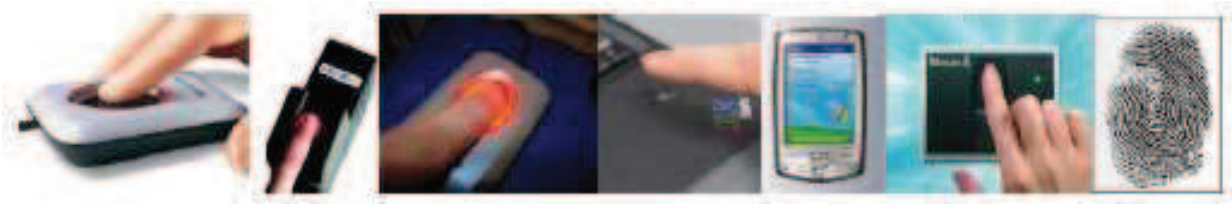
La première phase d'un système de reconnaissance d'empreintes digitales consiste à obtenir une image de l'empreinte du doigt. Longtemps le seul moyen existant a été l'utilisation du papier et de l'encre ce qui a rendu la tâche de reconnaissance très lourde. En effet la qualité de l'image était plutôt mauvaise (plusieurs acquisitions étaient nécessaires) et l'extraction de la signature était effectuée visuellement par un expert (processus très long et pénible). Heureusement avec le développement de l'informatique et de la microélectronique de nouveaux moyens d'acquisition ont fait leur apparition, permettant ainsi d'accélérer la chaine de traitement en l'automatisant (un capteur dédié fournit directement une image numérique).

Il s'agit de capturer les images numériques d'empreintes qui permettront de trouver les lignes tracées par les crêtes (en contact avec le capteur) et les vallées (creux).

Obtenir des images numériques n'est pas une chose simple, les capteurs rencontrent parfois certaines difficultés. Par exemple, les empreintes digitales de certaines ethnies ou classes d'âges sont plus fines (c'est le cas des asiatiques et des enfants).

La qualité d'image de l'empreinte digitale peut varier selon que la peau du doigt est sale, trop humide ou trop sèche, huileuse ou affligée d'une coupure. La pression que l'on exerce sur le lecteur optique de l'appareil est aussi déterminante quant aux détails qui sont recueillis. Un bon système biométrique tiendra compte de ces facteurs.

Les techniques utilisées pour la mesure sont diverses : capteurs optiques, capteurs ultrasoniques, capteurs de champ électrique, de capacité, de température... Ces capteurs sont souvent doublés d'une mesure visant à établir la validité de l'échantillon d'un doigt : les battements du cœur, la pression sanguine...etc.



FigureII.5: Les périphériques d'acquisition d'empreintes digitales

Tous les capteurs possèdent le même fonctionnement de base : ils récupèrent une image d'empreinte digitale, puis la transmettent à une unité de traitement (ordinateur, microprocesseur, etc., parfois intégrée dans le capteur même), par laquelle elle sera analysée. Le point commun à toutes les technologies utilisées pour la prise d'image d'une empreinte, est que l'image est constituée à partir des points de contact du doigt sur le capteur.

II.5 Extraction des caractéristiques

La représentation d'une empreinte est essentielle pour le système automatique. Cette représentation devrait avoir les propriétés ci-dessous :

- Retenir les caractéristiques discriminantes (uniques) de chaque empreinte.
- Facile à calculer.
- Stable et invariant au bruit et à la distorsion.

Plusieurs méthodes d'identification par empreintes digitales existent, se basant les unes et les autres sur différentes caractéristiques des empreintes digitales.
Globalement, ces méthodes peuvent être classifiées en :

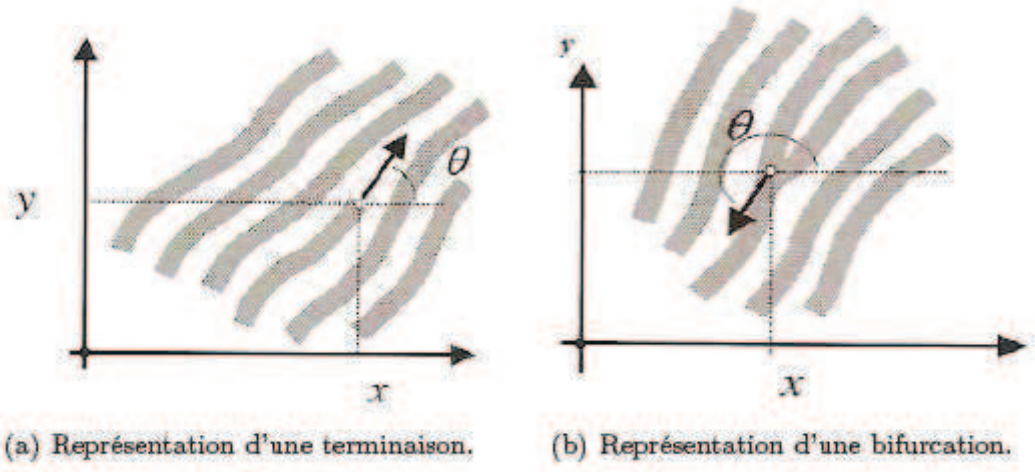
- Méthodes basées minuties.
- Méthodes basées textures.
- Méthodes hybrides.

Nous introduisons dans ce qui suit, quelques algorithmes utilisés dans chaque étape pour chacune des méthodes citées.

II.5.1 Méthodes basée minuties

C'est la méthode qui a suscité le plus d'intérêt par sa simplicité de mise en œuvre. Elle ne retient que l'emplacement des minuties les plus pertinentes. Elle est peu sensible aux déformations des doigts entre plusieurs vérifications (doigts plus ou moins appuyés sur le capteur). Chaque minutie est caractérisée par :

- Son type : bifurcation ou terminaison
- Sa position : coordonnées (x, y)
- La direction du bloc local associé à la strie où est située la minutie : θ



FigureII.6: Les caractéristiques extraites des minuties

L'objectif de cette méthode est de construire le fichier signature, qui correspond à l'information utile contenue dans l'image et qui est nécessaire à l'identification. Il s'agit de la liste des minuties détectées associées de leurs caractéristiques.

Cette méthode nous intéresse dans le cadre de notre travail. Elle sera développée dans ce chapitre.

II.5.2 Méthodes basées texture

Dans le cas idéal, l'image d'une empreinte digitale présente des arêtes dont la direction et la moyenne d'intensité varient continuellement ce qui constitue une texture orientée.

Des techniques utilisant les caractéristiques de texture d'une image d'empreinte ont été développées dans le but de remédier au problème d'introduction de fausse minuties et la négligence des bonnes minuties.

Les caractéristiques discriminatoires utilisées par les développeurs dans cette méthode sont :

1. Nombre, type et position des singularités.
2. Attributs géométriques des lignes d'arêtes.
3. Caractéristiques de formes.
4. Information locale des textures.
5. Pores.

Une technique d'analyse locale et globale de la texture de l'image d'empreintes digitales présentant un sous ensemble des points de l'image globale autour du noyau a été proposée par Jain et al [33]. C'est à partir de cette zone d'intérêt, qu'un vecteur de caractéristique appelé code d'empreinte est formé. Le processus d'appariement est basé sur le calcul des distances euclidiennes entre les codes d'empreintes. Le code d'empreinte de l'image d'entrée est généré et comparé avec les autres codes d'empreintes enregistrés dans la base de données représentant l'empreinte de référence, ce qui donne un résultat de plusieurs distances euclidiennes qui représentent toutes un certain degré de similarité. Le score final d'appariement entre les deux empreintes est la distance minimale entre les distances euclidiennes déjà calculées. Cette distance minimale correspond au meilleur alignement des deux empreintes appariées.

II.5.3 Méthodes hybrides

Nous regroupons ici l'ensemble des classifications combinant deux ou plusieurs techniques d'identification par empreintes digitales. Ces techniques peuvent provenir d'une même ou de différentes méthodes. Le but est de tirer profit des points forts de chacune des techniques combinées afin de maximiser le taux d'exactitude lors du processus d'identification d'empreintes. La fusion au niveau des caractéristiques extraites des empreintes digitales combinent les vecteurs caractéristiques pour constituer un seul vecteur de

dimension plus élevé dans le but d'élever le degré de discrimination de l'identité d'une personne.

La fusion au niveau du score d'appariement combine les scores individuels de plusieurs algorithmes d'appariement. Enfin la fusion au niveau décisionnel combine les décisions individuelles d'acceptation ou de rejet prises par les systèmes à fusionner.

Plusieurs travaux sur l'appariement des empreintes digitales ont mis en évidence le fait que l'hybridation maximise le taux d'identification en bénéficient des avantages de chaque approche.

Ros et al [34] ont suggéré l'utilisation de deux algorithmes l'un basé sur les minuties et l'autre sur la texture des arêtes pour apparier les empreintes, ainsi ils ont montré que la performance de l'algorithme basée minuties peut être améliorée d'une manière significative en utilisant des informations additionnelles fournies par la méthode du code d'empreinte. Ils utilisent une fusion au niveau des scores d'appariement par la méthode de la somme, définie comme suit :
Soient S_{MR} (Score Matching Ridge) et S_{MM} (Score Matching Minutiae) les scores de similarité obtenus par les deux algorithmes, alors le score final S_F de similarité sera calculé comme suit :

$$S_F = \lambda S_M + (1- \lambda) S_T \qquad \lambda \epsilon [0, 1]$$

Pour favoriser l'un des deux algorithmes λ peut être varié, mais dans leurs résultats expérimentaux ils l'ont fixé à 0,5.

II.6 Amélioration des empreintes digitales par les filtres de Gabor

II.6.1 Introduction

Les fonctions de Gabor sont un outil d'analyse de texture très prisé.

- **Définition de la texture**

Dans le domaine du traitement de l'image et de la vision par ordinateur, il n'existe pas de définition satisfaisante de la texture, en plus elle n'est pas nécessaire du fait que la distinction entre les différentes régions texturées repose essentiellement sur les traits caractérisant ces mêmes régions. Dans la plupart des cas et en particulier pour les images d'empreintes ces trais sont la fréquence et l'orientation dominante [35].

- **Définition des filtres de Gabor**[1]

Un filtre de Gabor est un filtre linéaire dont la réponse impulsionnelle est une sinusoïde multiplié par une enveloppe gaussienne.

De nombreux papiers ont été publiés sur leurs applications depuis que Gabor a proposé en 1946 la fonction 1D de Gabor [36]. La famille 2D des filtres de Gabor a été premièrement initiée par Daugman en 1980 comme un avant-projet pour comprendre les propriétés sélectives en orientation et en fréquence de ce genre de filtres.

La fonction 2D de Gabor est une oscillation harmonique, composée d'une onde plane sinusoïdale avec une fréquence particulière et orienté dans une direction donnée modulée par une enveloppe gaussienne.

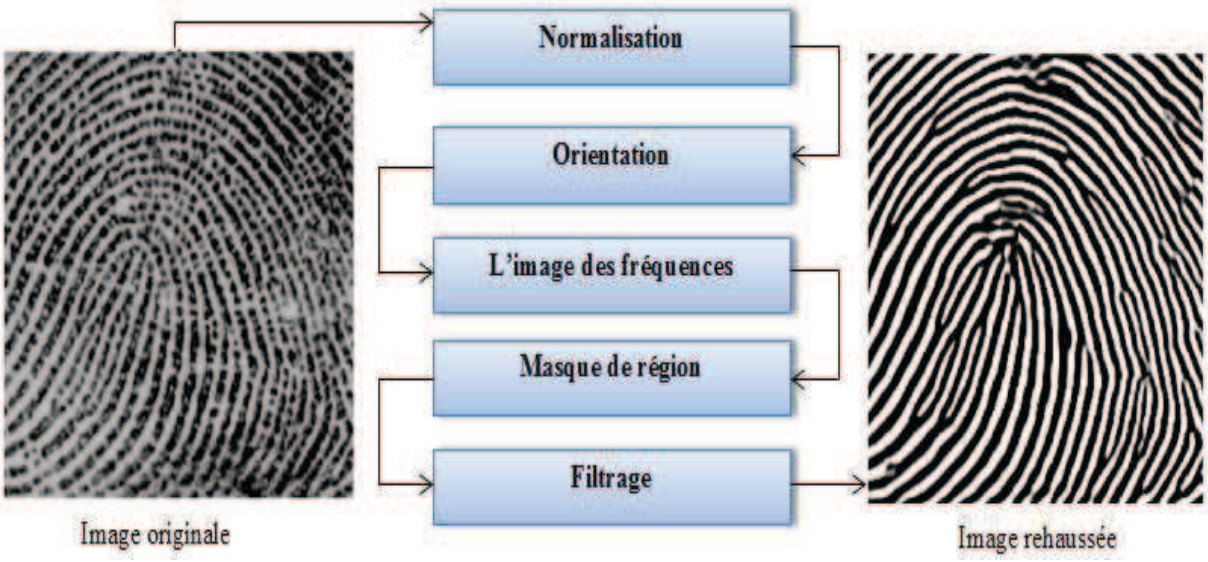
Ce filtre peut être appliqué pour résoudre la problématique d'amélioration des empreintes digitales. Il s'agira de bien fixer les paramètres d'orientation et de fréquence par rapport à chaque pixel de l'image.

II.6.2 Processus de rehaussement d'une image d'empreinte

Le but de cette étape est de supprimer toute ambiguïté en détectant des zones de bruit et en faisant ressortir la plus grande partie possible d'information utile au système pour pouvoir effectuer la reconnaissance.

[1] Il porte le nom du physicien anglais d'origine hongroise Dennis Gabor

L'algorithme utilisé est proposé par le groupe : *Lin Hong, Yifei Wan*, et *Anil Jain* à l'Université de l'Etat du Michigan [37].



FigureII.7 : Processus de rehaussement d'une image d'empreinte [37]

Nous avons repris le même principe en adaptant des modes de calcul presque similaires.

Pour l'image directionnelle, nous allons utiliser l'approche proposée par Ratha, Chen et Jain [39]. Pour l'image des fréquences, nous allons utiliser l'approche décrite dans [37] se basant sur la x-signature :

a. Notation :

Une image d'empreinte en niveau de gris I, est définie comme une matrice N x N.

I (i, j) représente l'intensité du pixel de la i^{ème} ligne et la j^{ème} colonne.

La moyenne et la variance de I sont définies comme suit :

$$M(I) = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} I(i,j)$$

$$Var(I) = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (I(i,j) - M(I))^2$$

Une image d'orientation **o** est définie comme une image N x N tel que o (i, j) est l'orientation locale de la crête au pixel (i, j). L'orientation locale est calculée pour chaque pixel à partir d'un bloc. L'image est divisée en blocs qui ne se chevauchent pas, et une seule valeur d'orientation est définie pour chaque bloc.

Une image de fréquence F est une image N x N tel que *F (i, j)* représente la fréquence locale de la crête. La structure de crêtes et de vallées définie sur le voisinage des minuties ne forment pas une onde sinusoïdale bien définie. Dans ce genre de situation, la fréquence est définie comme étant la moyenne des fréquences de ses voisins.

Les principales étapes de l'algorithme sont :

b. Normalisation :

L'image en entrée est normalisée, ainsi elle aura une moyenne et une variance pré-spécifiées.

Soit ξ l'image normalisée :

$$\xi(i,j) = \begin{cases} M_0 + \sqrt{\frac{Var_0 \ (I(i,j)-M)^2}{Var}} & \text{Si } I(i,j) > M \\ M_0 - \sqrt{\frac{Var_0 (I(i,j)-M)^2}{Var}} & \text{Sinon} \end{cases}$$

M_0 et Var_0 sont les valeurs de moyenne et de variance désirées.

Le but principal de la normalisation est de réduire les variations dans les valeurs des niveaux de gris le long des crêtes et des vallées, ce qui facilitera les étapes suivantes du processus (la normalisation ne change pas la clarté de l'image).

c. Orientation :

Afin de calculer l'image directionnelle, une méthode basée sur le gradient est appliquée. C'est celle proposée par RAO [38]. L'algorithme de base est donné ci-dessous :

1. Diviser l'image d'entrée en blocs de taille W x W ;

2. Calculer les gradients G_x et G_y de chaque pixel (*i, j*), et cela dans chaque bloc ;
3. Estimer l'orientation locale dominante dans chaque bloc en utilisant les équations suivantes (formule des moindres carrés) :

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

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

$$\theta(i,j) = \frac{1}{2}\tan^{-1}\frac{V_x(i,j)}{V_y(i,j)}$$

d. L'image de fréquence :

Sur un voisinage local qui ne contient ni minuties ni points singuliers, les niveaux de gris sur les crêtes et les vallées peuvent être modélisés par une onde sinusoïdale le long de la direction normale à l'orientation locale des crêtes. Ainsi, la fréquence locale est une autre propriété intrinsèque à l'image de l'empreinte.

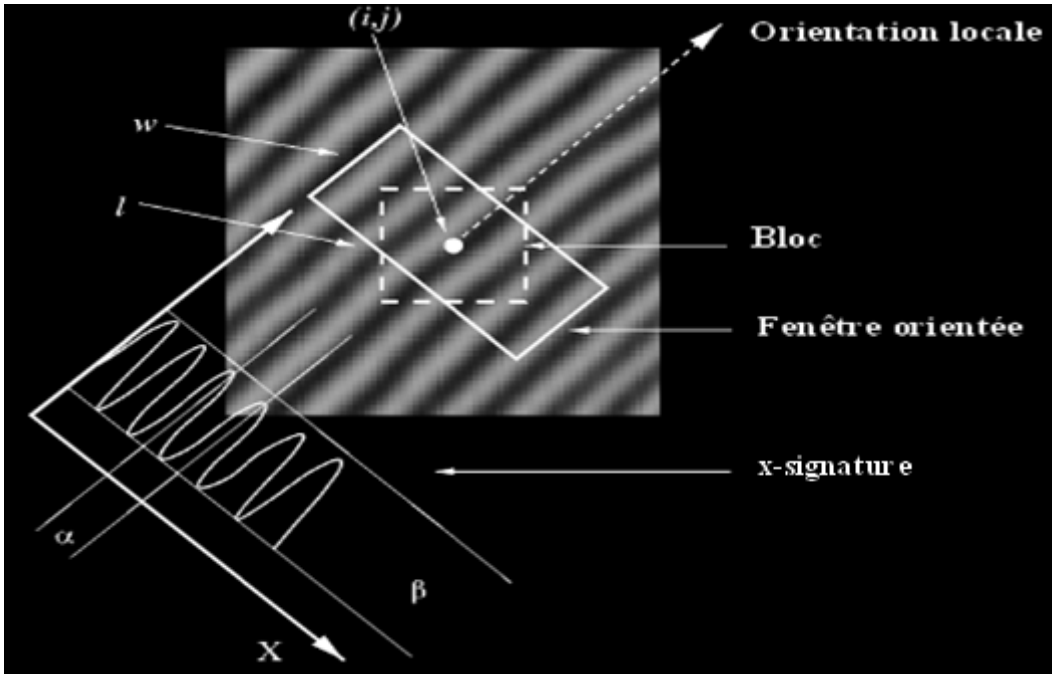
Soient ξ l'image normalisée et **o** l'image d'orientation. Les étapes suivantes permettent de déterminer la fréquence locale:

1. Diviser ξ en blocs de taille w x w (16 x 16)
2. Pour chaque bloc centré en (*i, j*), calculer une fenêtre orienté de taille *l x w* (32 x 16) qui est définie sur le système des coordonnées des crêtes (FigureII.9)
3. Pour chaque bloc centré en (*i, j*), calculer la x-signature, *X [0], X [1], ...X [l-1]* des crêtes et des vallées sur la fenêtre orientée de la façon suivante :

$$X[k] = \frac{1}{w}\sum_{d=0}^{w-1}\xi(u,v), k = 0,1,\dots,l-1$$

$$u = \mathrm{i} + (\mathrm{d} - \frac{w}{2})\cos o(i,j) + (k - \frac{l}{2})\sin o(i,j)$$

$$\mathrm{v} = \mathrm{j} + (\mathrm{d} - \frac{w}{2})\sin o(i,j) + (\frac{l}{2} - k)\cos o(i,j)$$



FigureII.8 : Une fenêtre orientée et la x-signature [35]

Si aucune minutie n'apparait sur la fenêtre orientée, la x-signature forme une onde sinusoïdale discrète qui a la même fréquence que les crêtes et les vallées contenues dans cette dernière.

Par conséquent, la fréquence des crêtes et des vallées peut être estimée à partir de la x-signature.

Soit $\tau\,(i, j)$ le nombre moyen de pixels entre deux pics consécutifs de la x-signature, la fréquence $\Omega\,(i, j)$ est calculée ainsi : $\Omega\,(i, j) = 1/\tau\,(i, j)$.

4. Pour une image scannée avec une certaine résolution, la valeur de la fréquence varie dans un certain intervalle. Une image de 500 dpi, l'intervalle est $[1/3, 1/25]$.

5. Pour les blocs contenant des minuties ou des points singuliers, les crêtes et les vallées sont corrompues, ainsi l'onde sinusoïdale est mal définie. Pour obtenir la fréquence, il faut procéder à une interpolation, qui est réalisé comme ceci :
   - Pour chaque bloc centré sur (i, j) :

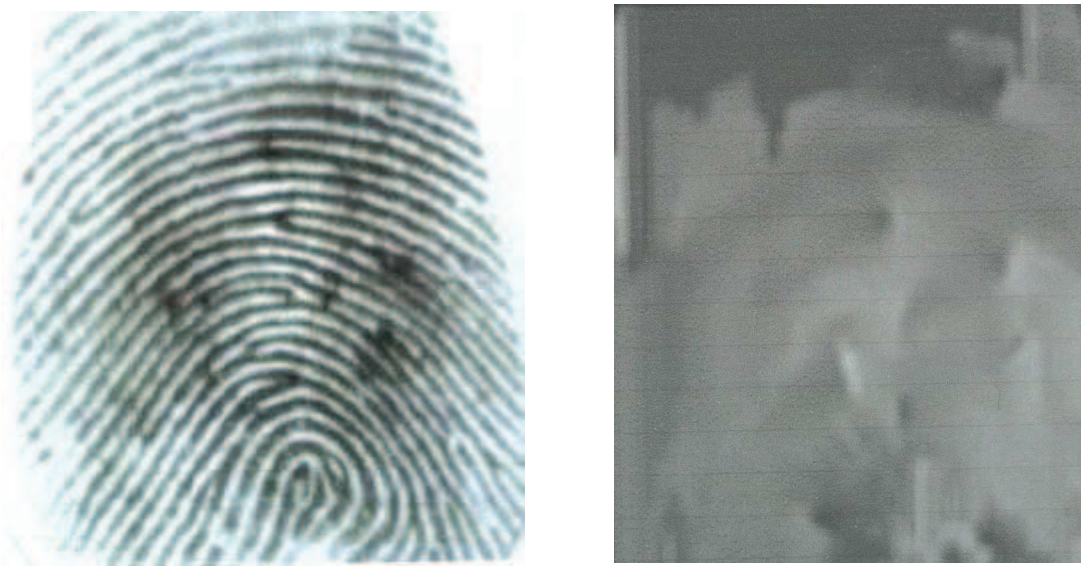
$$\Omega'(i,j) = \begin{cases} \Omega(i,j) & \text{si } \Omega(i,j) \neq -1 \\ \dfrac{\sum_{u=-f/2}^{f/2}\sum_{v=-f/2}^{f/2} W_g(u,v)\ \mu(\Omega(i-uw, j-vw))}{\sum_{u=-f/2}^{f/2}\sum_{v=-f/2}^{f/2} W_g(u,v)\ \delta(\Omega(i-uw, j-vw))} & \text{sinon} \end{cases}$$

Avec: $\mu(x) = \begin{cases} 0 & \text{si } x \leq 0 \\ x & \text{sinon} \end{cases}$

$\delta(x) = \begin{cases} 0 & \text{si } x < 0 \\ x & \text{sinon} \end{cases}$

W_g est un noyau gaussien avec une moyenne de 0 et une variance de 9 ; f=7.

La figure ci-dessous montre une image de fréquence en niveau de gris :



FigureII.9 : Image de fréquence en niveau de gris

e. Masque de région :

La classification du pixel qui consiste à savoir s'il appartient à une région d'intérêt (crêtes et vallées) ou à la région du fond, peut être faite en se basant sur l'évaluation de la forme de l'onde formée par les crêtes et les vallées. Dans cette algorithme, trois caractéristiques sont utilisées pour qualifier l'onde sinusoïdale ; l'amplitude (α), la fréquence (β) et la variance (γ).

Soit *X [0], X [1],...X [l-1]* la x-signature du bloc centré en (*i, j*) avec :

1. α = la hauteur moyenne des pics - la profondeur moyenne des vallées.
2. $\beta = 1/\tau(i,j)$
3. $\gamma = \frac{1}{l}\sum_{i=1}^{l}(X[i] - (\frac{1}{l}\sum_{i=1}^{l}X[i]))^2$

Ces valeurs sont ensuite comparées à des valeurs de référence pour savoir si le bloc w x w est une région valide ou pas.

f. Filtrage :

La fréquence du filtre est déterminée par la fréquence locale des crêtes, et l'orientation du filtre par l'orientation locale des crêtes. Quant à δ_x et δ_y ils sont définis empiriquement. Soit **ξ** l'image normalisée, **O** l'image d'orientation, **F** l'image de fréquence, **R** le masque de région et G(x, y, θ, T) le filtre de Gabor. L'image améliorée A est obtenue par convolution spatiale :

$$A(i,j) = \begin{cases} 255 & \text{Si R (i, j) = 0 (région fond)} \\ \sum_{u=-g/2}^{g/2}\sum_{v=-g/2}^{g/2} G(u,\ v, 1/F(i,j), o(i,j))\ \xi(i-u, j-v) & \text{sinon} \end{cases}$$

Avec g=11, la taille du filtre de Gabor

Afin de calculer l'image directionnelle, une méthode basée sur le gradient est appliquée. C'est celle proposée par Ratha, Chen et Jain [39]. L'algorithme de base est donné ci-dessous :

1. Diviser l'image d'entrée en blocs de taille W x W ;
2. Calculer les gradients Δ_x et Δ_y de chaque pixel (*i, j*), et cela dans chaque bloc ;
3. Estimer l'orientation locale dominante dans chaque bloc en utilisant les équations suivantes (formule des moindres carrés) :

$$\theta = \frac{\pi}{2} + \frac{1}{2}\operatorname{atan} 2(2G_{xy}, G_{xx} - G_{yy})$$

$$G_{xy} = \sum_{h=-8}^{8}\sum_{k=-8}^{8} \Delta_x\left(x_i + h, y_j + k\right) \times \Delta_y\left(x_i + h, y_j + k\right)$$

$$G_{xx} = \sum_{h=-8}^{8}\sum_{k=-8}^{8} \Delta_x\left(x_i + h, y_j + k\right)^2$$

$$G_{yy} = \sum_{h=-8}^{8}\sum_{k=-8}^{8} \Delta_y\left(x_i + h, y_j + k\right)^2$$

II.7 Processus d'extraction des minuties d'une empreinte digitale

II.7.1 La binarisation

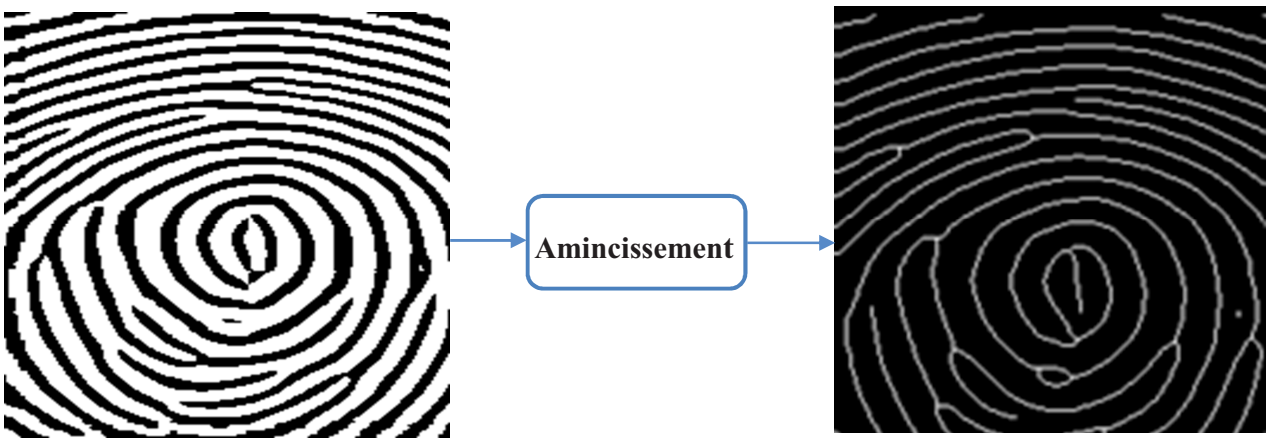
Pour permettre la squelettisation, l'image en niveaux de gris ***P*** doit d'abord être binarisée (les pixels prennent deux valeurs, la valeur 0 qui représente le noir et la valeur 255 qui représente le blanc). Pour effectuer ce traitement, on utilise le seuil ***M*** qu'on a pris comme étant la moyenne des niveaux de gris de l'image et cela comme suit :

$$IT(i,j) = \begin{cases} 255 & \text{Si } P(i,j) > M \\ 0 & \text{sinon} \end{cases}$$

II.7.2 L'amincissement

Dans l'image binarisée (noir et blanc) les lignes se voient clairement mais elles ont des tailles différentes. Pour pouvoir détecter rapidement les minuties (terminaisons, bifurcations), il est nécessaire d'obtenir une image plus schématique de l'empreinte, dans laquelle toutes les lignes ont la même épaisseur (1 pixel).

Notre algorithme d'amincissement est illustré dans la figure suivante :



FigureII.10 : Amincissement de l'image d'empreinte digitale

Il existe deux classes d'algorithmes d'amincissement qui sont basés sur des méthodes morphologiques itératives.

L'algorithme utilisé [41] figure parmi les algorithmes parallèles, qui ont l'avantage d'être performant surtout en termes de rapidité contrairement aux algorithmes séquentiels. Il introduit des critères pour décider si un point p_1 doit être supprimé.

Cet algorithme emploie une image binaire définie par la matrice IT qui possède comme valeurs des **0** et des **1**, ainsi qu'une autre image utilisée pour décider si un pixel de frontière peut être éliminé, nommée carte de drapeau.

La carte de drapeau est employée pour marquer les pixels qui seront par la suite supprimés, la taille est identique à celle de l'image binaire ; initialement, tous les pixels dans la carte de drapeau sont placés à 0, la valeur sera changée en 1 dès que ce pixel sera marqué.

P_9	P_2	P_3
P_8	P_1	P_4
P_7	P_6	P_5

(a)Dans l'image binaire

m_9	m_2	m_3
m_8	m_1	m_4
m_7	m_6	m_5

(b)Dans la carte de drapeau

Les fonctions utilisées dans l'algorithme d'amincissement :

1. La fonction de voisinage précédent B, définie comme suit :

$$B(p_1) = \sum_{i=2}^{9} p_i$$

2. La fonction de transition A, définie comme suit :

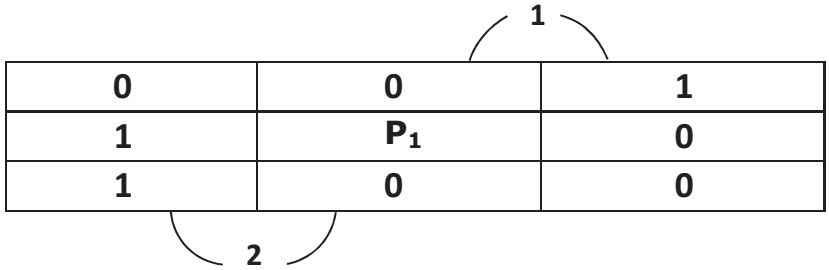
$$A(p_1) = \sum_{i=2}^{9} Count(p_i)$$

Avec:

$$\text{Count}(p_i) = \begin{cases} 1 & \text{si } ((p_i = 0) \;\&\&\; ((p_{i+1} = 1)) \\ 0 & \text{sinon} \end{cases}$$

Et $p_{10} = p_2$, $m_{10} = m_2$

La figure qui suit donne un exemple de calcul de cette fonction :



FigureII.11 : Calcul de transition 0 ⟶ 1

Les symboles p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9 montrés dans la figure suivante représentent les pixels IT[i][j] dans l'image binaire IT et leurs voisins IT[i-1][j-1], IT[i-1][j], IT[i-1][j+1], IT[i][j+1], IT[i+1][j+1], IT[i+1][j], IT[i+1][j-1], IT[i][j-1] respectivement.

P_9 (i-1, j-1)	**P_2 (i-1, j)**	**P_3 (i-1, j+1)**
P_8 (i, j-1)	**P_1 (i, j)**	**P_4 (i, j+1)**
P_7 (i+1, j-1)	**P_6 (i+1, j)**	**P_5 (i+1, j+1)**

FigureII.12 : Désignations des neuf pixels dans une fenêtre 3 x 3

II.7.2.1 Algorithme d'amincissement

Notre méthode pour l'extraction du squelette d'une image d'empreinte consiste à supprimer les points de contour de l'image, sauf les points qui appartiennent au squelette. Pour conserver la connectivité du squelette, on divise chaque itération en deux sous-itérations.

Dans la 1ere sous-itération, le point du bord P_1 sera supprimé s'il satisfait les conditions suivantes :

$2 \leq B(P_1) \leq 6$ (a)

$A(P_1) = 1$ (b)

$P_{2*} P_{4*} P_6 = 0$ (c)

$P_{4*} P_{6*} P_8 = 0$ (d)

$A(P_1)$ est le nombre de 0 1 dans l'ensemble ordonnée : $\{P_2, P_3, P_4, P_5, P_6, P_7, P_8, P_9\}$, et qui sont les voisins de P_1 (FigureII.12), et $B(P_1)$ est le nombre de voisins de P_1 qui ne sont pas à zéro.

$B(P_1) = P_2+P_3+P_4+\ldots+ P_8+ P_9$

Si l'une des conditions: (a), (b), (c), (d) n'est pas satisfaite, P_1 ne sera pas supprimé de l'image.

Dans la 2eme sous-itération, seulement les conditions (c) et (d) sont changées (FigureII.13) comme suit :

(c') $P_{2*} P_{4*} P_8 = 0$

(d') $P_{2*} P_{6*} P_8 = 0$

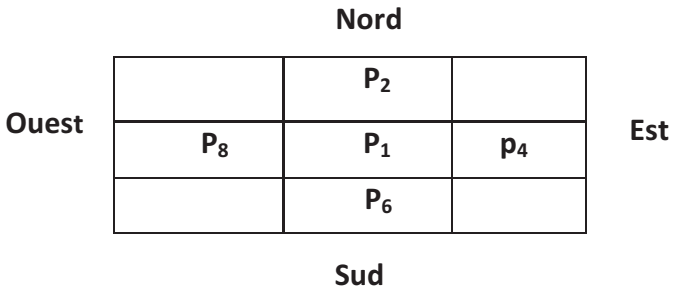
D'après les conditions (c) et (d) de la 1ere sous-itération, on remarque que cette itération supprime seulement les points du bord sud-est et les points corniers du nord-ouest qui n'appartiennent pas à un squelette parfait.

La preuve pour la 1ere sous-itération est donnée, c'est que les points à supprimer satisfont les conditions suivantes :

(c) $P_{2*} P_{4*} P_6=0$ (1)

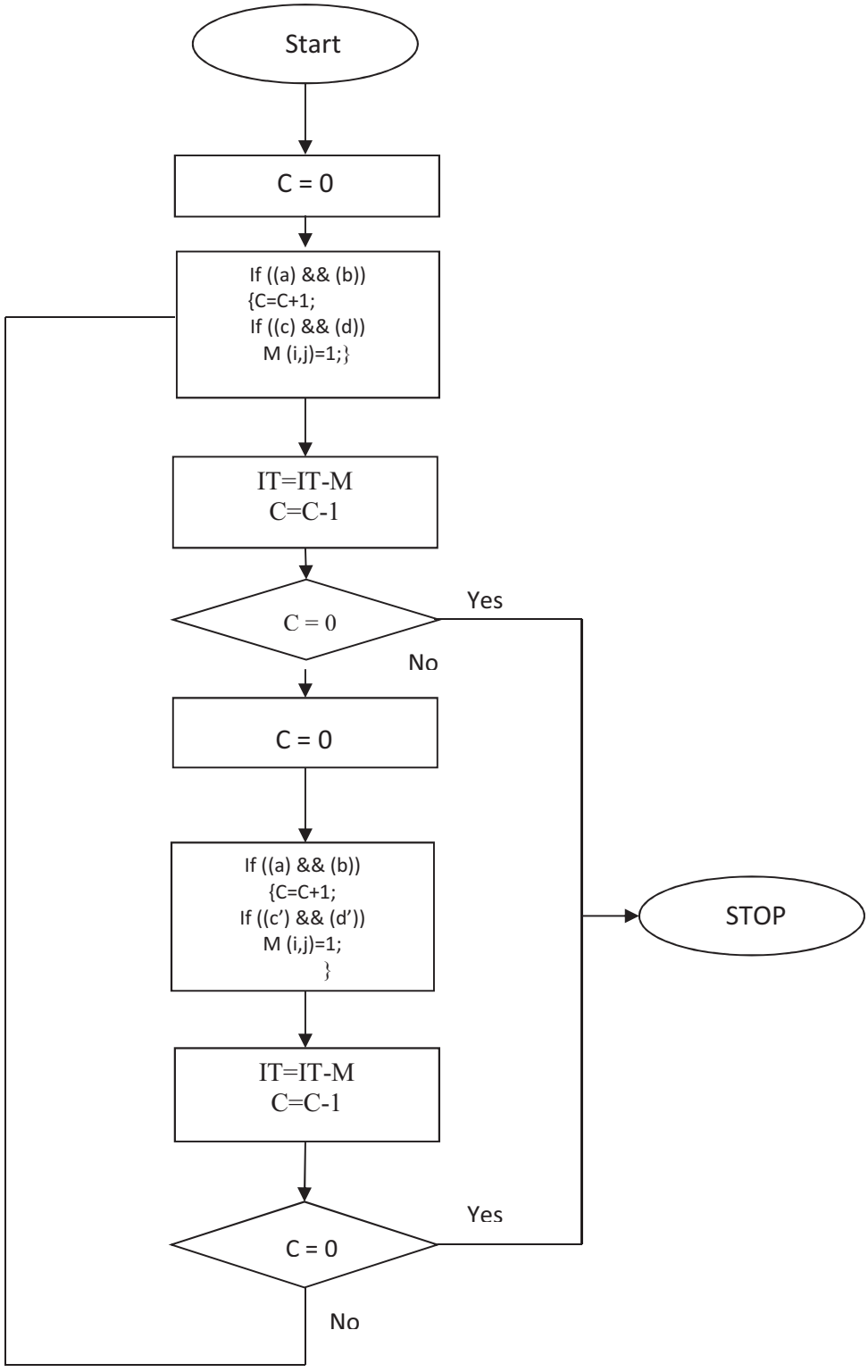
(d) $P_{4*} P_{6*} P_8=0$ (2)

Les solutions de l'ensemble des équations (1) et (2) sont $p_4=0$ ou $p_6=0$ ou ($P_2=2$ et $P_8=0$). Donc le point p_1 qui a été supprimé est un point du bord sud-est ou un point cornier nord-ouest. De même, pour la 2^{eme} sous-itération le point p_1 qui a été supprimé est un point du bord nord-ouest ou un point cornier sud-est.



FigureII.13 : Points sous considération et leurs localisations

Le diagramme de l'algorithme d'amincissement implémenté est montré dans la figure (FigureII.14). Initialement, l'image originale est sauvegardée dans la matrice IT et le compteur C est mis à 0. Le résultat de l'image traitée est sauvegardé dans la matrice IT.



FigureII.14 : Organigramme d'un algorithme d'amincissement

II.7.3 L'extraction des minuties

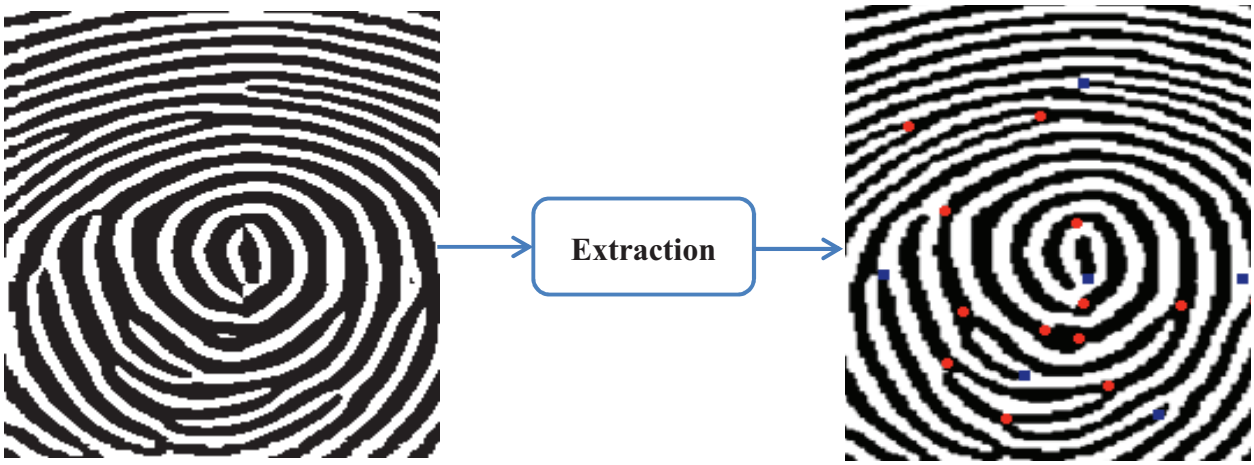
Les *minuties* de l'empreinte digitale sont extraites à partir de son squelette en calculant la fonction de voisinage CN (Crossing Number) initiée par Arcelli [42] en chaque point *P (i, j)* de l'image de la manière suivante :

$$CN = 0.5 \sum_{i=2}^{9} |P_i - P_{i+1}|, \quad P_{10} = P_2$$

(a) Une fin arrête (*CN=1*) (B) Une bifurcation (CN=3)

FigureII.15 : Calcul du CN dans un voisinage de 8 pixels (cadre rouge)

Dans un premier temps on repère l'emplacement de toutes les *minuties* présentes au sein de l'image de l'empreinte digitale, que l'on sauvegarde en associant à chacune d'entre elles la position absolue (x, y) correspondante et le type de *minutie* (*terminaison* ou *bifurcation*).
Le résultat de notre algorithme d'extraction des minuties est illustré dans la figure qui suit :



FigureII.16 : Extraction des minuties d'une empreinte digitale

II.8 L'appariement [43]

Cette étape joue un rôle très important dans un système de reconnaissance, car elle permet la vérification et/ou l'identification d'un individu.

Les différentes approches de comparaison d'empreintes digitales sont classées en trois grandes familles.

1. La comparaison basée sur la corrélation : Elle consiste à comparer tout simplement les matrices de pixels des images de deux empreintes, et calculer la corrélation de pixels. Les techniques basées sur la corrélation de pixels ne sont pas très efficaces car on peut obtenir des images très différentes d'une même empreinte. Si les images d'une même empreinte sont différentes, les valeurs de leurs pixels sont d'office différentes. De plus, deux images prises à des résolutions différentes (par deux capteurs différents par exemple) ne peuvent avoir les mêmes valeurs de pixels. On peut toujours se donner une marge de tolérance (exemple : si la différence des valeurs de deux pixels est inférieure à 3, on considère que ces pixels sont égaux), mais ces techniques ne sont clairement pas rigoureuses.

2. La comparaison basée sur les minuties : C'est la technique la plus utilisée car elle se base sur des traits individuels à chaque personne, les minuties. Ces dernières sont extraites de deux empreintes, et représentées comme un ensemble de points dans un plan à deux dimensions selon le modèle de coordonnées. La comparaison consiste à trouver un bon alignement des minuties de deux empreintes (T et I) qui produit un maximum de paires de minuties semblables. Cette famille sera développée plus tard**.**

3. La comparaison basée sur les singularités de lignes d'empreintes digitales : Lorsqu'on dispose d'une image de mauvaise qualité, il est très difficile (voire impossible si la qualité est très mauvaise) de réaliser l'extraction de minuties; en revanche, d'autres singularités de lignes d'empreintes digitales telles que l'orientation locale, la taille, la position géométrique, la fréquence, la texture locale, les pores de respirations…. peuvent être plus facilement extraites que les minuties. Ces singularités sont cependant très instables et très variables d'une personne à une autre et d'un doigt à un autre (d'une même personne). Ces techniques sont donc très peu utilisées.

II.8.1 La comparaison basée minuties [26]

Comme son nom l'indique, la comparaison de minuties (généralement appelée le "matching de minuties") consiste à comparer les minuties de deux empreintes en les plaçant dans le plan bidimensionnel et à trouver les paires de minuties qui ont le même emplacement et la même orientation ; on dit que ces minuties ont été "matché".

On peut donc voir les deux empreintes à comparer I et T comme des vecteurs de minuties, où chaque minutie m est représentée à son tour par un vecteur (x, y, θ) où x et y sont les coordonnées de l'emplacement de la minutie dans l'image de l'empreinte, et θ l'angle de la minutie (i.e. l'orientation de la minutie,) ; c'est le modèle de coordonnées.

$\boldsymbol{T} = \{\mathbf{m}_1, \boldsymbol{m}_2, \ldots, \boldsymbol{m}_m\}$ $\quad \boldsymbol{m}_i = \{x_i, y_i, \theta_i\}$ $\quad i=1..m,$

$\boldsymbol{I} = \{\boldsymbol{m'}_1, \boldsymbol{m'}_2, \ldots, \boldsymbol{m'}_n\}$ $\quad \boldsymbol{m'}_j = \{x'_j, y'_j, \theta'_j\}$ $\quad j=1..n.$

Où m et n sont respectivement le nombre de minuties de *T* et *I*.

On considère qu'il y a un "match" entre une minutie $\boldsymbol{m'}_j$ de $\boldsymbol{T}$ et une minutie $\boldsymbol{m}_i$ de $\boldsymbol{I}$ lorsque la distance *sd* (pour *spatial distance*) qui les sépare est inférieure ou égale à une tolérance r_o et la différence *dd* (pour *direction difference*) de leurs angles est inférieure ou égale à une certaine tolérance angulaire θ_o :

$$sd(\mathbf{m'_j}, \mathbf{m_i}) = \sqrt{(x'_j - x_i)^2 + (y'_j - y_i)^2} \leq r_0 \qquad (1) \text{ and}$$

$$dd(\boldsymbol{m'_j}, \boldsymbol{m_i}) = min(|\theta'_j - \theta_i|, 360 - |\theta'_j - \theta_i|) \leq \theta_0 \qquad (2)$$

Dans ce qui suit, nous allons définir les fonctions de base qui seront utilisées dans l'algorithme d'appariement.

Soit mm (m''_j, m_i) la fonction qui renvoie 1 si la minutie $\mathbf{m_i}$ est appariée avec la minutie m″j selon les équations (1) et (2) :

$$\text{mm}(m''_j, m_i) = \begin{cases} 1 & \text{si } sd(m''j, m_i) \leq r_0 \text{ et } dd(m''j, m_i) \leq \theta_0 \\ 0 & \text{sinon} \end{cases}$$

Alors le problème d'appariement peut être formulé ainsi :

$$\text{maximize}_{\Delta x,\, \Delta y,\, \theta,\, P} \sum_{i=1}^{m} mm(map_{\Delta x,\Delta y,\theta}(m'_{P(i)}), m_i) \qquad (3)$$

P(i) est la fonction qui détermine l'appariement entre les minuties de *I* et celle de *T*. Chaque minutie a soit exactement une correspondante dans l'autre empreinte digitale, ou elle n'en a pas.

Tout ceci est résumé comme suit :

1- $P(i) = j$ indique que la correspondante de m_i en T est la minutie m'_j en *I*.
2- $P(i)$ = null indique que la minutie m_i en T n'a pas de compagnon dans *I*.
3- Une minutie m'_j en *I,* n'a pas de correspondante dans *T* si $P(i) \neq j$ $\forall i=1\ldots m$.
4- $\forall i = 1\ldots m,\ k = 1\ldots m,\ i \neq k \Rightarrow P(i) \neq P(k)$ ou $P(i) = P(k)$ = null (cela signifie que chaque minutie dans *I* est associée à un maximum d'une minutie dans *T*, ce qui veut dire aussi que *P* est une fonction bijective).

II.8.2 Notre méthode d'appariement

La méthode qu'on a implémentée est basée sur la transformation de Hough proposée par Chang et al (1997).

Elle comprend les étapes suivantes

1. Détecter la paire de minutie (appelée la paire principale) qui reçoit le maximum Matching Pair Support (MPS) et les paramètres d'alignement (θ, s) qui peuvent faire correspondre un maximum de minuties entre *T* et *I*. La paire principale qui a le MPS maximale est déterminée à travers un processus de vote basé sur la transformée de Hough.
2. La fonction P est Alors déterminé une fois les deux empreintes digitales ont été enregistrées pour superposer les minuties constituant la paire principale.
3. L'alignement exact est calculé au sens des moindres carrés une fois la fonction de correspondance est connue.
4. Le score de l'appariement est ainsi calculer selon cette formule:

Score= K / ((n+m)/2) avec K le nombre de minuties appariées

Pour accomplir l'étape 1, qui est le cœur de cette approche, l'algorithme considère les segments formés par les paires de minuties m_{i2} m_{i1} dans T et m'_{j2} m'_{j1} dans I et dérivée de chaque paire de segments, les paramètres θ et s selon les formules suivantes:

$$\theta = \text{angle}(\overline{m_{i2}m_{i1}}) - \text{angle}(\overline{m'_{j2}m'_{j1}}) \qquad (11)$$

$$s = \frac{length(\overline{m_{i2}m_{i1}})}{length(\overline{m'_{j2}m'_{j1}})} \qquad (12)$$

La paire principale et les paramètres (θ*, s*) sont déterminés selon l'algorithme suivant :

maxMPS =0 //maximum Matching Pair Support

for each $\mathbf{m}_{i1}$, $i1$= 1…m

for each $\mathbf{m}'_{j1}$, $j1$= 1…n // $\mathbf{m}_{i1}$, $\mathbf{m}'_{j1}$ is the current pair for which MPS has be to estimated

{Reset **A** // the accumulator array

for each $\mathbf{m}_{i2}$, $i2$= 1…m, , $i2 \neq i1$

for each $\mathbf{m}'_{j2}$, j2 = 1…n, $j2 \neq$ j1

{θ, s are computed from $\overline{\boldsymbol{m}_{i2}\boldsymbol{m}_{i1}}$, $\overline{\mathbf{m}'_{j2}\,\mathbf{m}'_{j1}}$ according to Equations (11) and (12)

θ^+, s^+= quantization of θ, s to the nearest bins

$\mathbf{A}[\theta^+, s^+] = \mathbf{A}[\theta^+, s^+] + 1$

}

MPS = max $_{\theta+, s+}$ $\mathbf{A}$ $[\theta^+, s^+]$

if MPS ≥ maxMPS

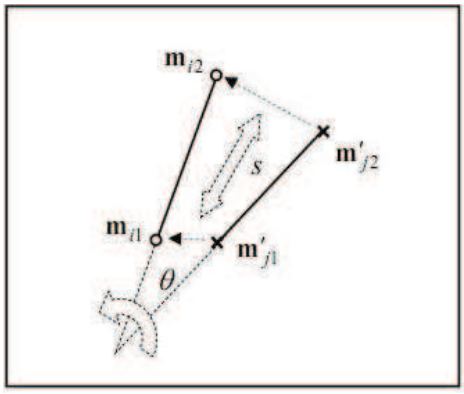
{ maxMPS = MPS

$(\theta^*, s^*) = \arg \max \mathbf{A}[\theta^+, s^+]$

Principal pair = ($\mathbf{m}_{i1}$, $\mathbf{m}'_{j1}$)

}

}



FigureII.17: L'alignement des deux segments provoqué par la rotation et le changement

II.9 Conclusion

A travers ce chapitre, nous avons décrit les caractéristiques des empreintes digitales ainsi que le fonctionnement global d'un système de reconnaissance d'empreintes digitales. Nous avons aussi présenté les différentes étapes du processus de rehaussement d'images d'empreintes.

Dans le prochain chapitre, nous nous intéresserons à la conception UML de notre projet.

CHAPITRE III

Conception

III.1 Introduction

Dans le but d'une meilleure organisation et une bonne maîtrise du travail, tout processus de développement d'application doit suivre une démarche méthodologique et rigoureuse, ainsi nous avons opté pour le langage UML « Unified Modeling Language »en raison de sa qualité de modélisation orienté objet, et de son formalisme relativement simple et compréhensible.

III.2 Conception du système

III.2.1 Identification des acteurs

1) **L'administrateur**

C'est la personne qui gère le système et qui choisit la configuration la plus efficace dans le but de le rendre robuste et performant.

- Il prend en charge la gestion de la base de données, il enregistre les nouvelles personnes.
- Il met à jour les informations des personnes et supprime ceux qui quittent le système.
- Il teste les paramètres du système et choisit la meilleure configuration.

2) **L'utilisateur**

Toute personne interagissant avec le système, en effectuent une authentification (vérification) ou une identification.

III.2.2 Le diagramme de cas d'utilisation

Le cas d'utilisation (ou use case) correspond à un objectif du système, motivé par un besoin d'un ou plusieurs acteurs.

L'ensemble des use cases décrit les objectifs (le but) du système.

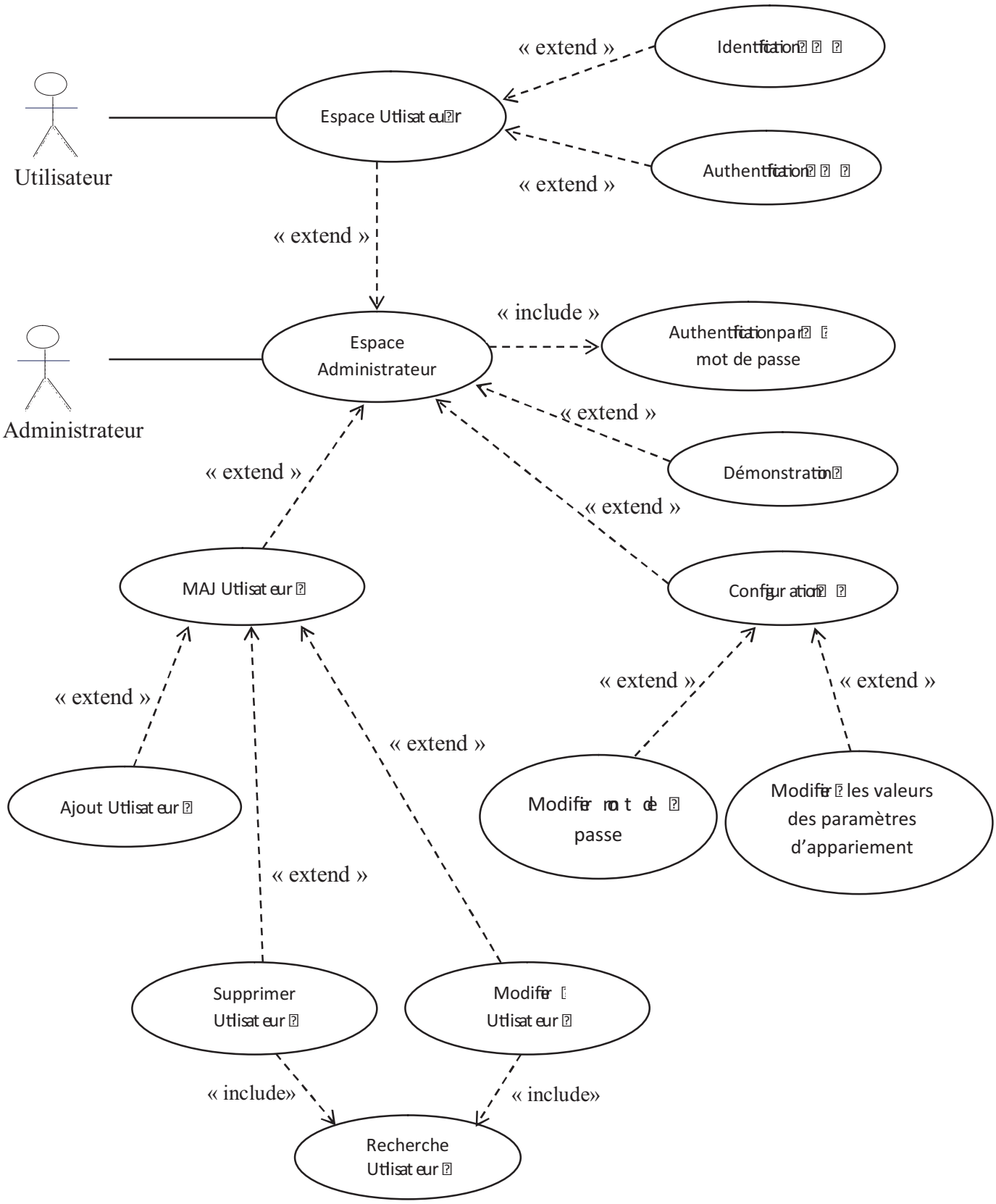


Figure III.1 : Diagramme de cas d'utilisation

Description textuelle de quelques cas d'utilisation

- ***Accéder à l'espace administrateur*:**

Apres avoir lancé l'application, l'administrateur clique sur le bouton *Espace Administrateur* une boite de dialogue lui demandant de saisir le mot de passe s'affiche. Si les informations sont correctes il accède à son espace, sinon un message d'erreur s'affiche.

- ***Ajout d'un utilisateur*** :

L'administrateur clique sur le bouton *MAJ Utilisateur* qui se trouve dans son espace. Une interface contenant un formulaire d'ajout s'affiche, il saisit les informations personnelles de la personne (N° CIN[1], Nom, Prénom, date de naissance, Profession) et joint à ce même formulaire sa photo d'identité ainsi que l'image de son empreinte digitale qui sera utilisée dans la reconnaissance. Ensuite, il clique sur le bouton *Ajouter* pour sauvegarder les informations dans la base de données.

- ***Configuration des paramètres de reconnaissance :***

L'administrateur clique sur le bouton *Configuration* se trouvant dans son espace, ensuite sur l'onglet paramètres de reconnaissance, une interface contenant résumé des paramètres appliqués s'affiche, ainsi qu'un formulaire de modification lui permettant de les redéfinir s'il le souhaite.

- ***Identification :***

Le but de ce cas d'utilisation est de trouver l'identité d'une personne grâce à son empreinte digitale. Une fois l'identification terminée, la fiche d'information de la personne dont le score d'appariement est supérieur ou égal au seuil de décision sera affichée.

- ***Démonstration* :**

Une fois dans son espace, l'administrateur clique sur le bouton *Démonstration,* une interface s'affiche lui permettant de suivre l'étape de rehaussement d'une image d'empreinte digitale ainsi que le processus d'extraction des minuties.

[1] Numéro de la carte d'identté natonal e

III.2.3 Les diagrammes de séquences :

Le diagramme de séquences montre les interactions entre objets selon un point de vue temporel. La représentation du contexte des objets se concentre sur l'expression des interactions.

Un objet est matérialisé par une barre verticale appelée ligne de vie des objets. Les objets, communiquent en échangeant des messages représentés au moyen de flèches orientées, de l'émetteur du message vers le destinataire. L'ordre des messages est donné par leur position sur l'axe vertical.

a. Accéder à l'espace administrateur

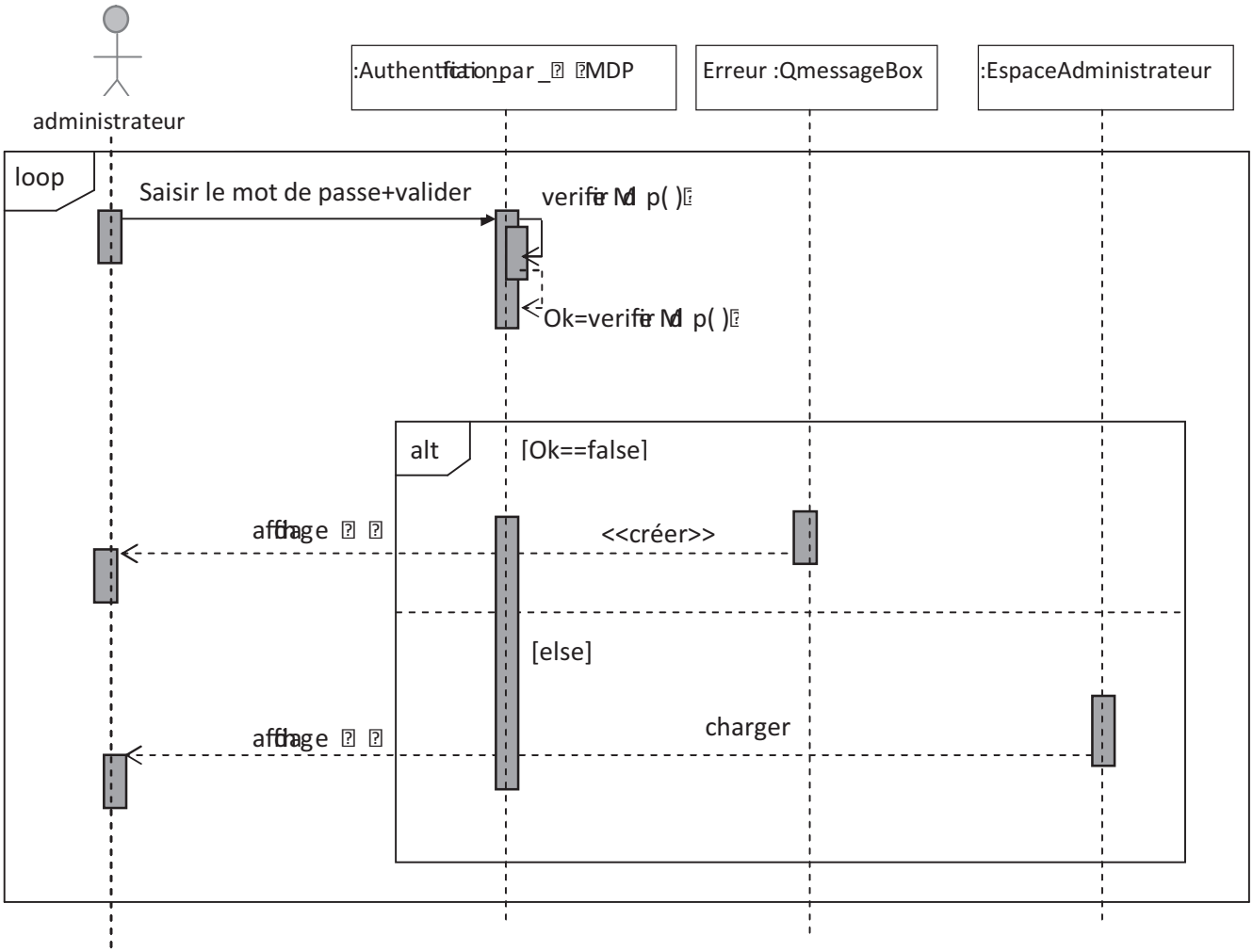


Figure III.2 : Diagramme de séquence du cas d'utilisation « Accéder à l'espace administrateur »

b. Lancer la démonstration

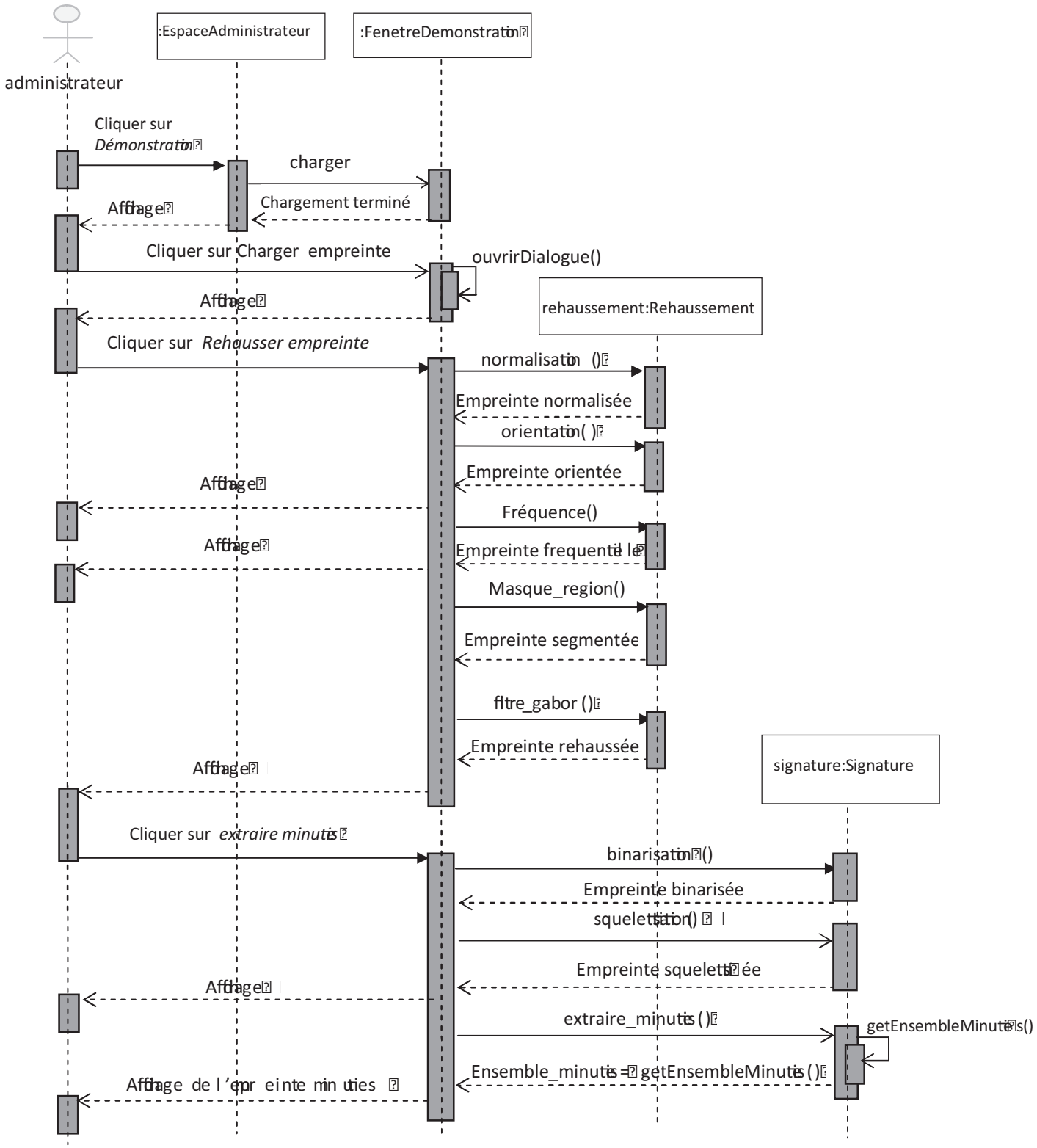


Figure III.3 : Diagramme de séquence du cas d'utilisation « Démonstration »

c. L'ajout (Enrôlement)

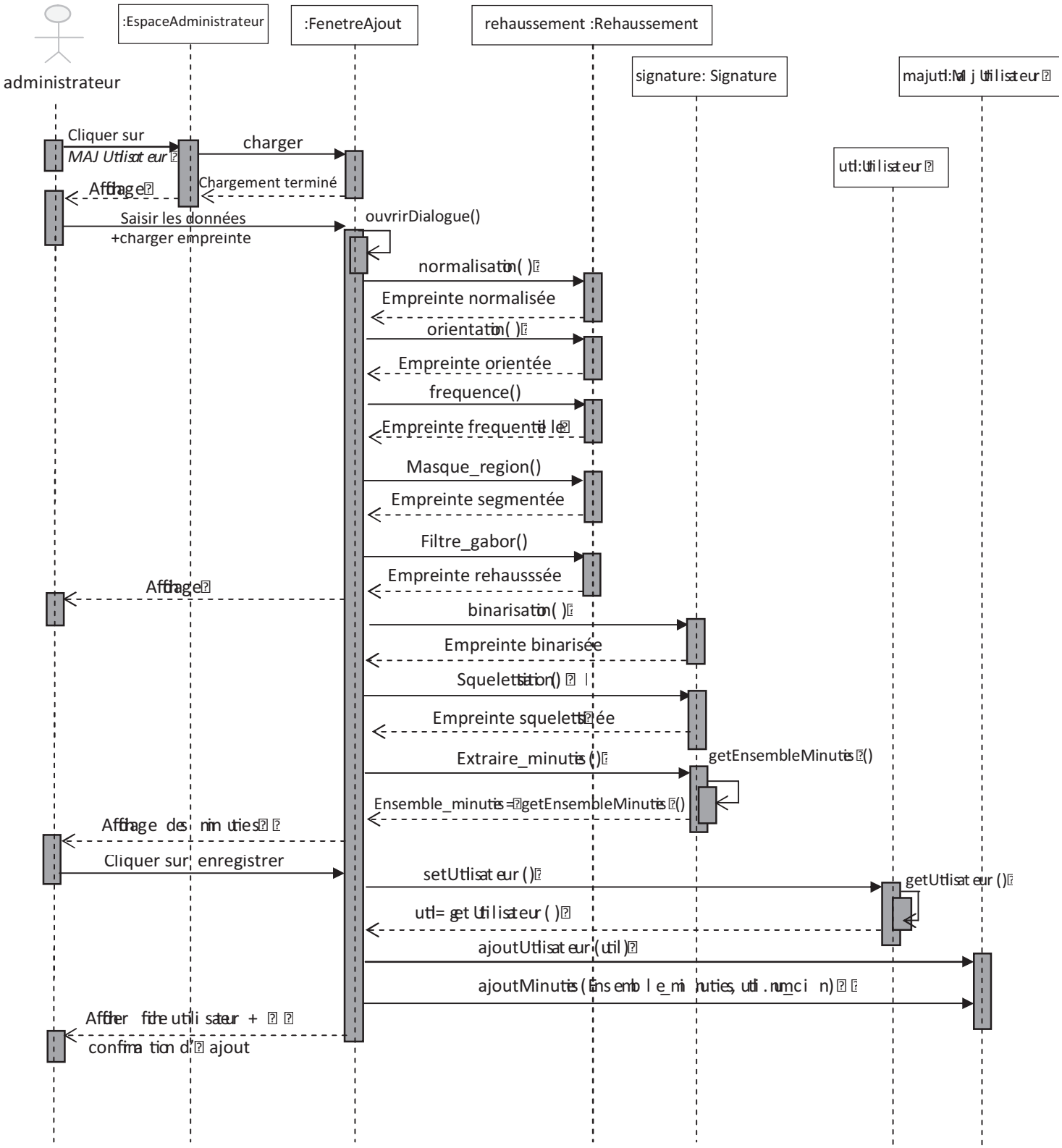


Figure III.4 : Diagramme de séquence du cas d'utilisation « Ajout d'un utilisateur »

d. S'identifier

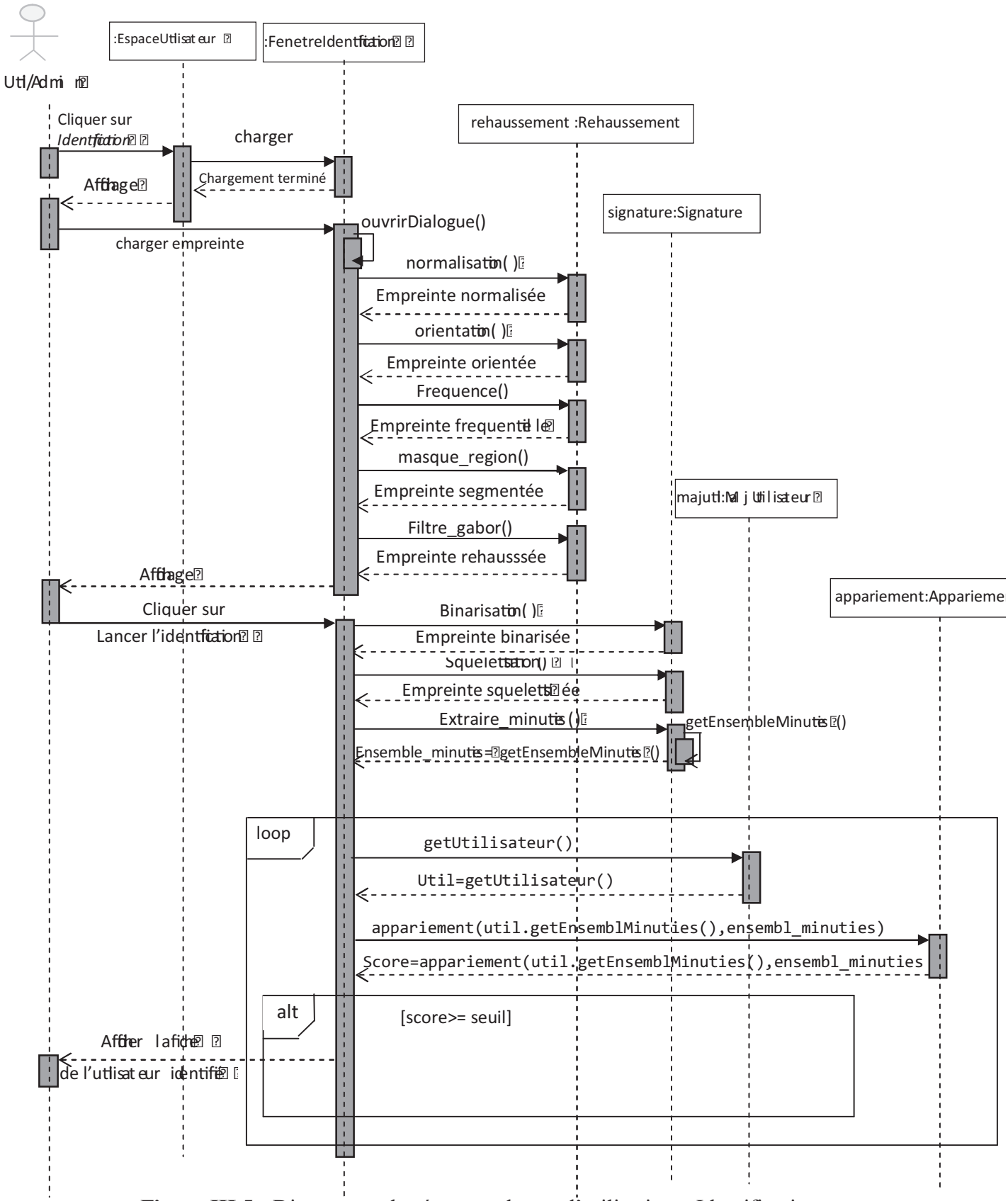


Figure III.5 : Diagramme de séquence du cas d'utilisation « Identification »

e. Configuration des paramètres de reconnaissance

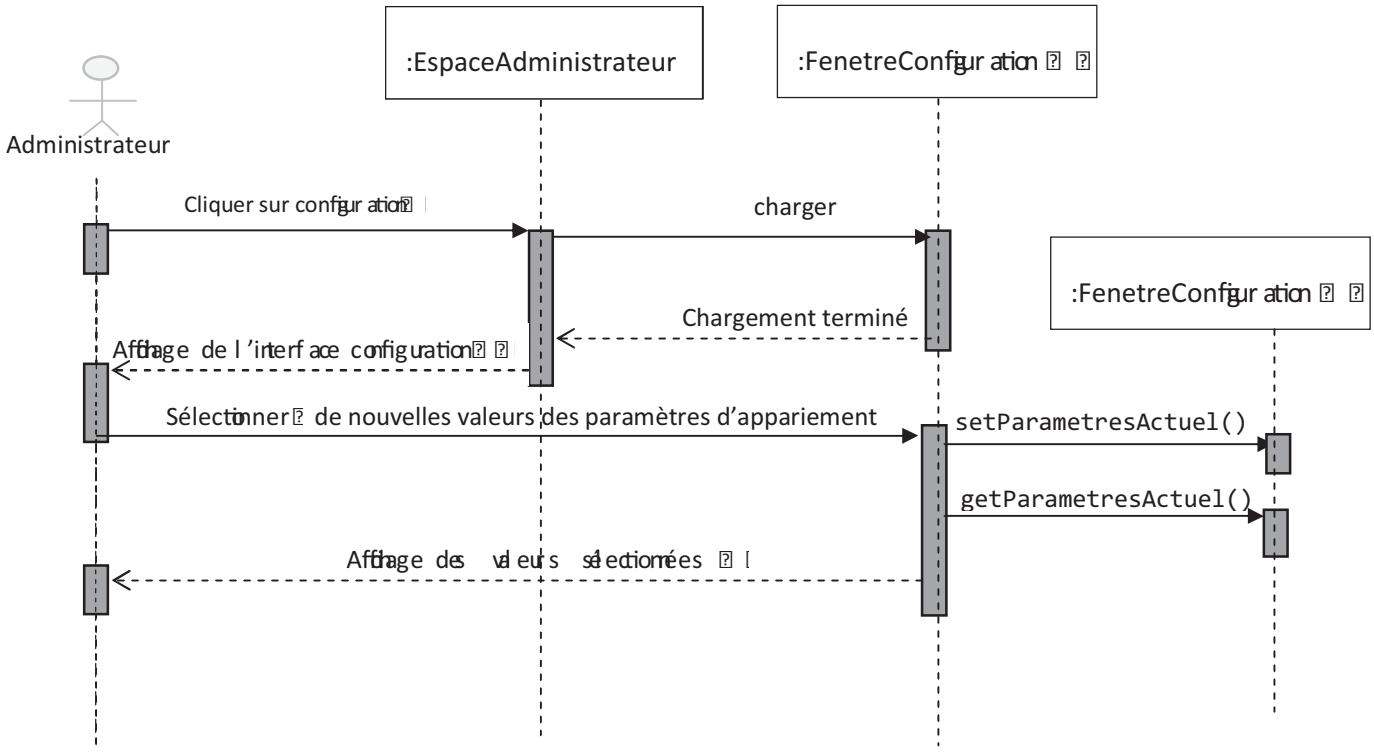


Figure III.6 : Diagramme de séquence du cas d'utilisation « modifier les valeurs des paramètres d`appariement »

III.2.4 Les diagrammes de classes

Il représente l'architecture conceptuelle du système : il décrit les classes que le système utilise, ainsi que leurs liens.

III.2.4.1 Diagramme de classes de l'application Administrateur

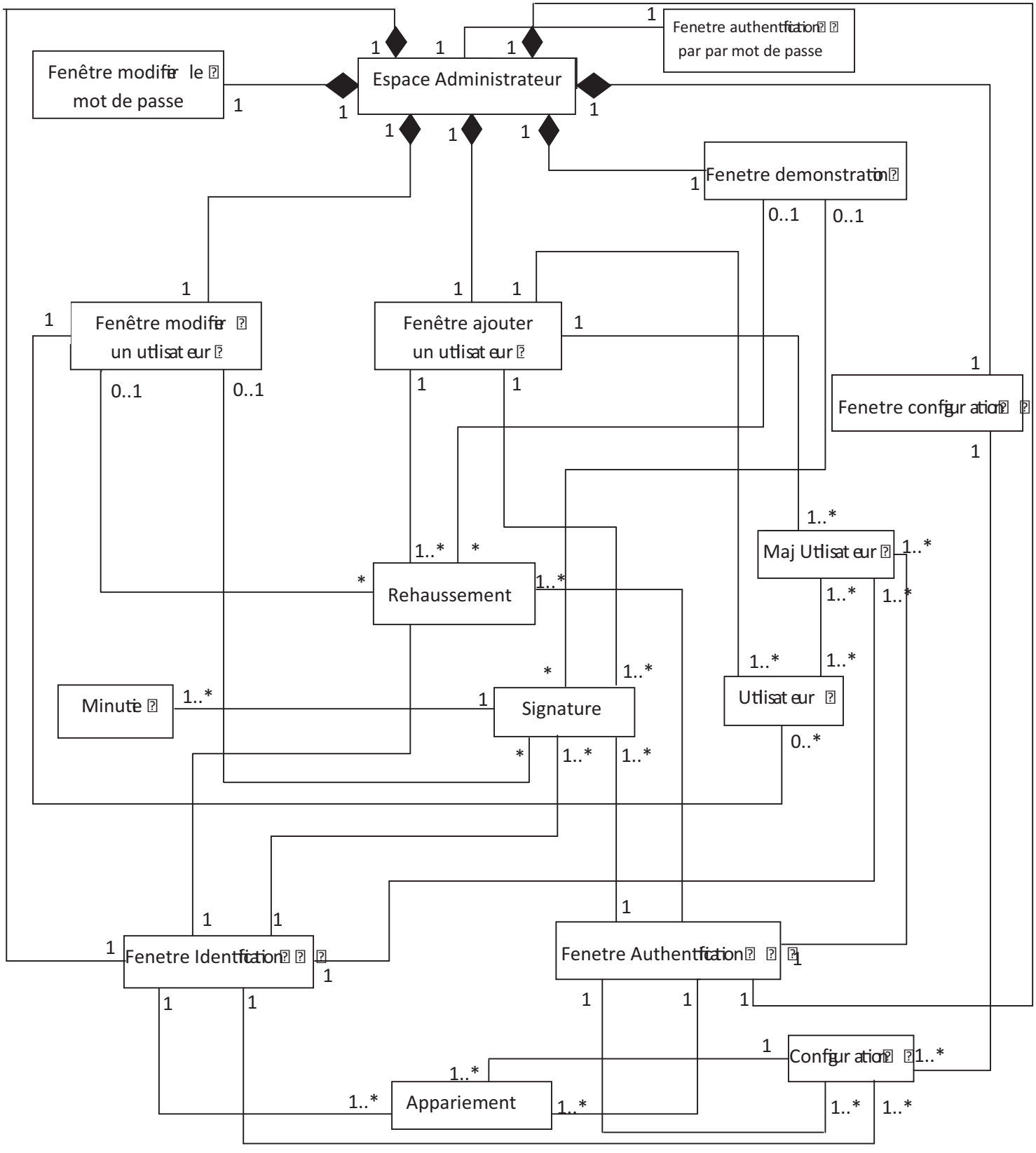


Figure III.7 : Diagramme de classes de l'application Administrateur

III.2.4.2 Diagramme de classes de l'application Utilisateur

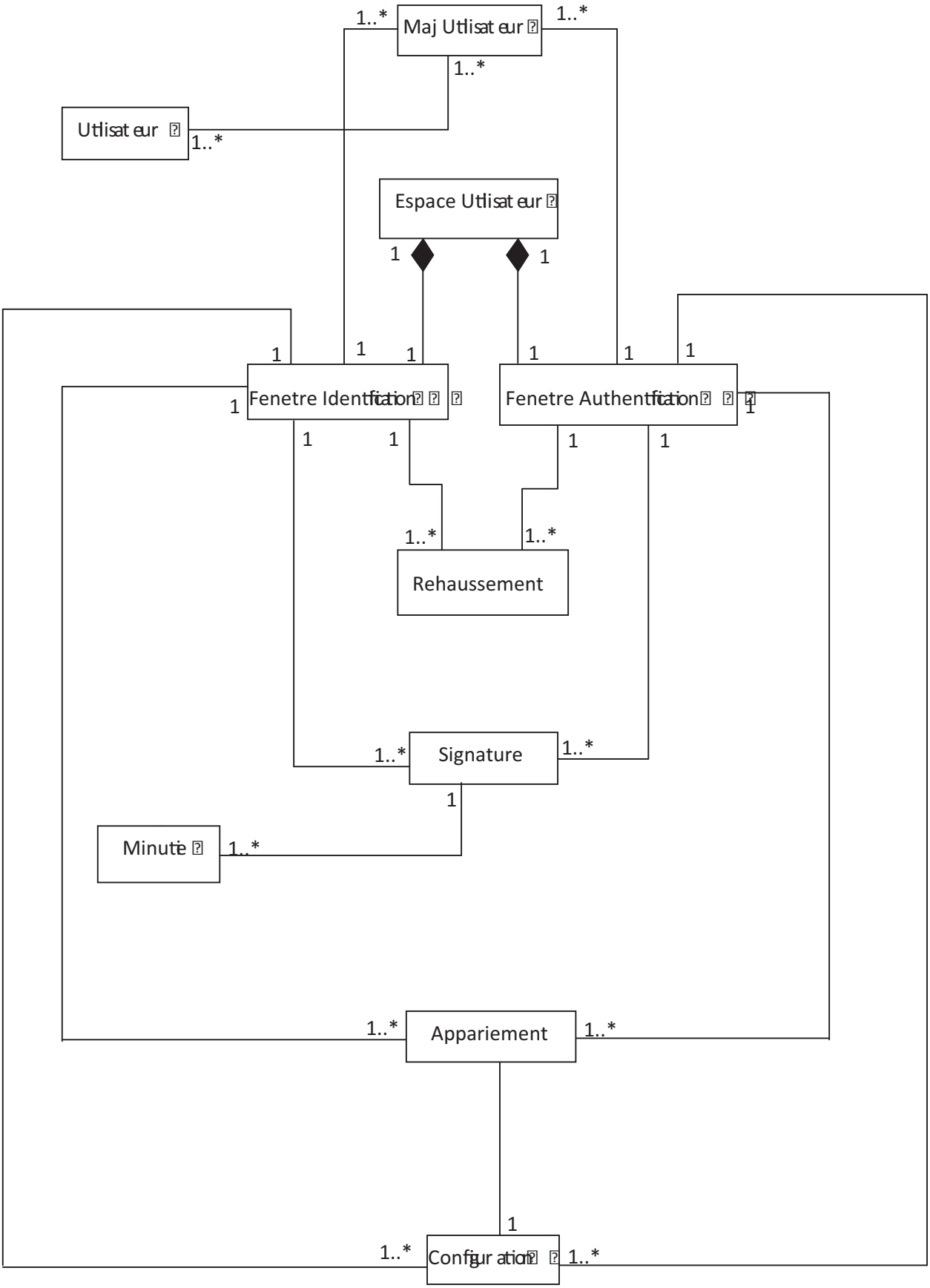


Figure III.8 : Diagramme de classe de l'application Utilisateur

III.3 Conclusion

Cette partie nous a permis de proposé une démarche de modélisation pour développer notre application. Cette démarche est basée sur l'UML, nous avons commencé par la spécification des cas d'utilisations dans un premier temps, suivi d'une élaboration des diagrammes de séquences ensuite les diagrammes de classes, ce qui a permis de définir les classes de notre application que nous allons utiliser pour la réalisation de cette dernière. Dans le prochain chapitre nous allons nous consacrer à présenter l'environnement et les outils de développement utilisés, nous présenterons quelques interfaces de notre application.

CHAPITRE IV

Réalisation

IV.1 Introduction

Après avoir finalisé l'étape de conception, nous passons à la réalisation de notre application « FingerPrint Recognition System ». Nous commençons tout d'abord par la description du langage et des outils utilisés, ensuite nous présenterons la façon dont l'application fonctionne à travers quelques interfaces, ainsi que quelques tests.

IV.2 Langage de programmation

IV.2.1 La puissance du langage C++

- Il est **très répandu**. Il fait partie des langages de programmation les plus utilisés sur la planète. On trouve donc beaucoup de documentation sur Internet et on peut facilement avoir de l'aide sur les forums.
- Il est **rapide**, très rapide même, ce qui en fait un langage de choix pour les applications critiques qui ont besoin de performances. C'est en particulier le cas des jeux vidéo, mais aussi des outils financiers ou de certains programmes militaires qui doivent fonctionner en temps réel.
- Il est **portable** : un même code source peut théoriquement être transformé sans problème en exécutable sous Windows, Mac OS et Linux. Vous n'aurez pas besoin de réécrire votre programme pour d'autres plates-formes !
- Il existe de **nombreuses bibliothèques** pour le C++. Les bibliothèques sont des extensions pour le langage, un peu comme des plug-ins. De base, le C++ ne sait pas faire grand-chose mais, en le combinant avec de bonnes bibliothèques, on peut créer des programmes 3D, réseaux, audio, fenêtrés, etc.
- Il est **multi-paradigmes**. Cela signifie qu'on peut programmer de différentes façons en C++. L'une des plus célèbres est la *Programmation Orientée Objet* (POO). C'est une technique qui permet de simplifier l'organisation du code dans nos programmes et de rendre facilement certains morceaux de codes réutilisables.

IV.3 Environnement et outils de développement

IV.3.1 Le Système d'exploitation Windows 7

Malgré sa récente sortie sur le marché des systèmes d'exploitation, cette dernière version éditer par le géant Microsoft est très apprécié du grand public pour sa rapidité son efficacité et surtout sa maniabilité.

IV.3.2 Wampserver 2.0

C'est un utilitaire qui installe et configure automatiquement un environnement de travail complet. Il regroupe un serveur web Apache, un serveur de bases de données MySQL et le PHP. Il joint également PhpMyAdmin à MySQL permettant ainsi la gestion de la base de données que ce soit par interface graphique ou par exécution des instructions SQL.

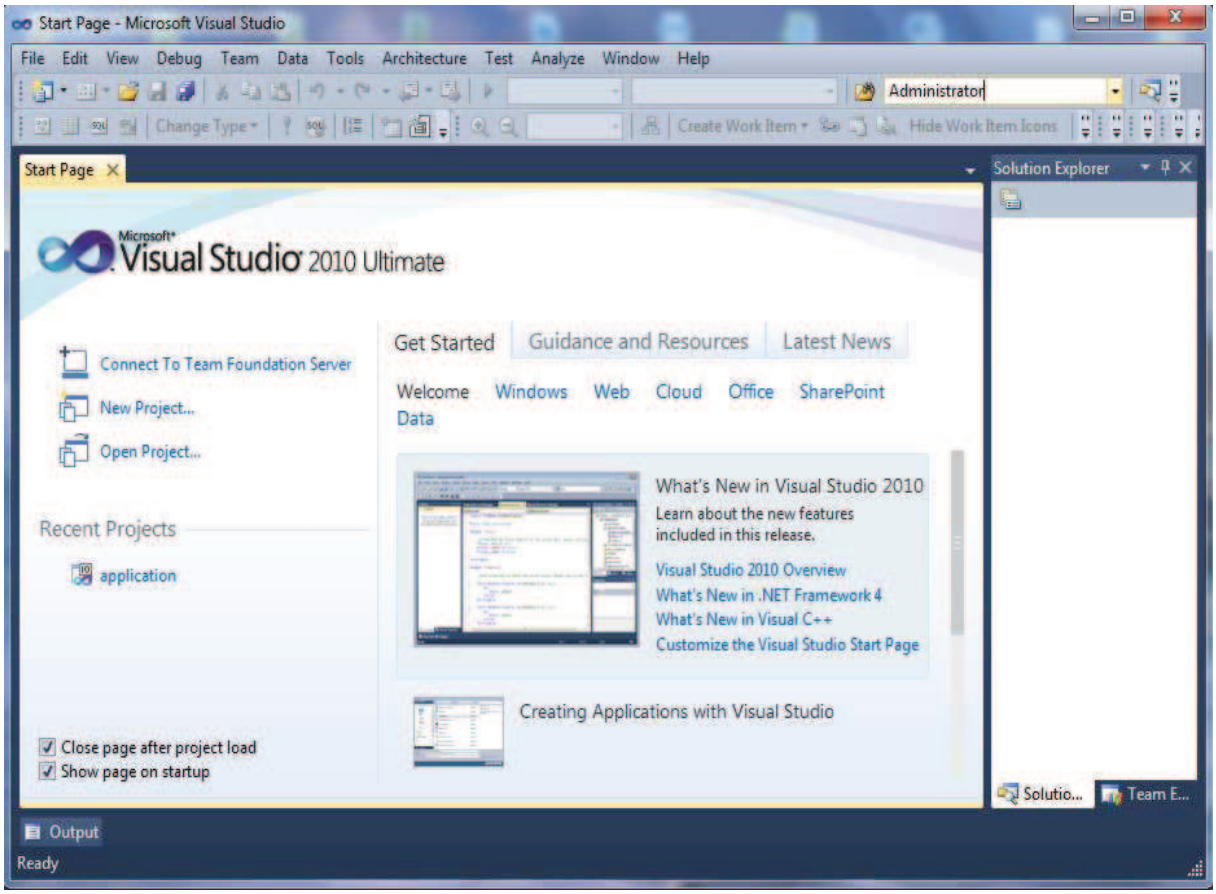
FigureIV.1 : Interface principale de Wampserver

IV.3.3 L'environnement Microsoft Visual Studio

Microsoft Visual Studio est un ensemble complet d'outils de développement, conçu par Microsoft. Nous avons utilisé la version « *Visual Studio 2010 Ultimate* ».

Il fournit un environnement intégré qui facilite les tâches courantes de création et de débogage pour détecter et résoudre les bogues rapidement et facilement afin de créer des solutions de haute qualité, tout en réduisant les coûts de développement.

Visual Studio 2010 Ultimate permet de laisser libre cours à l'imagination et de donner vie aux idées facilement.



FigureIV.2 : Interface de l'environnement Microsoft Visual Studio 2010 Ultimate

IV.3.4 Le Framework Qt

Qt est une ***bibliothèque*** multiplateforme pour créer des GUI (programme sous forme de fenêtre).
Elle est écrite en C++ et est faite pour être utilisée à la base en C++, mais il est aujourd'hui possible de l'utiliser dans d'autres langages comme Java, Python, etc.

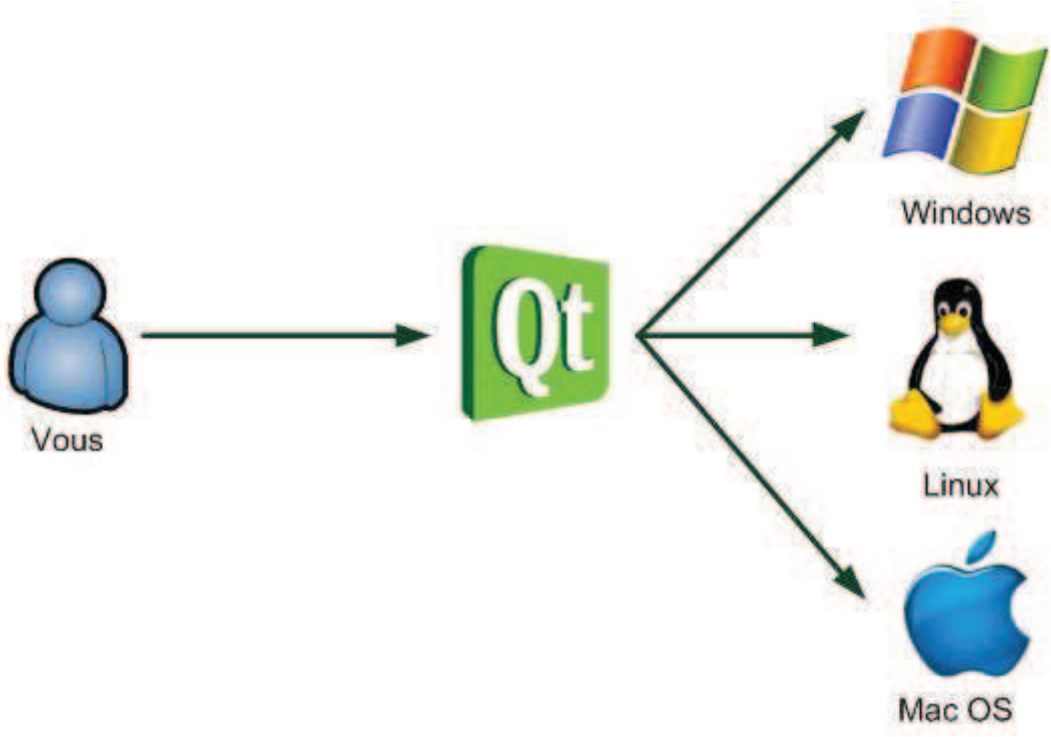
Elle est en fait... bien plus qu'une bibliothèque. C'est un ensemble de bibliothèque. Le tout est tellement énorme qu'on parle d'ailleurs plutôt de ***Framework*** : cela signifie qu'on a à notre disposition un ensemble d'outils pour développer nos programmes plus efficacement.

Qu'on ne s'y trompe pas : Qt est à la base faite pour créer des fenêtres, c'est en quelque sorte sa fonction centrale. Mais ce serait dommage de limiter Qt à ça.
Qt est donc constituée d'un ensemble de bibliothèques, appelées "modules". On peut y trouver entre autres ces fonctionnalités :

- **Module GUI** : c'est toute la partie création de fenêtres.
- **Module OpenGL** : Qt peut ouvrir une fenêtre contenant de la 3D gérée par OpenGL.
- **Module de dessin** : pour tous ceux qui voudraient dessiner dans leur fenêtre (en 2D), le module de dessin est très complet !
- **Module réseau** : Qt fournit une batterie d'outils pour accéder au réseau, que ce soit pour créer un logiciel de Chat, un client FTP, un client Bittorent, un lecteur de flux RSS...
- **Module SVG** : possibilité de créer des images et animations vectorielles, à la manière de Flash.
- **Module de script** : Qt supporte le JavaScript (ou ECMAScript), que vous pouvez réutiliser dans vos applications pour ajouter des fonctionnalités, sous forme de plugins par exemple.
- **Module XML** : pour ceux qui connaissent le XML, c'est un moyen très pratique d'échanger des données avec des fichiers formés à l'aide de balises, un peu comme le XHTML.
- **Module SQL** : permet un accès aux bases de données (MySQL, Oracle, PostgreSQL...).

Qt est un ***Framework multiplateforme***, développé initialement par la société Trolltech, qui fut racheté par Nokia. Elle est distribuée sous deux licences, au choix : LGPL ou propriétaire. Celle qui nous intéresse est la licence LGPL car elle nous permet d'utiliser gratuitement Qt (et même d'avoir accès à son code source si on veut !).

Voici un schema illustrant le fonctionnement de Qt :



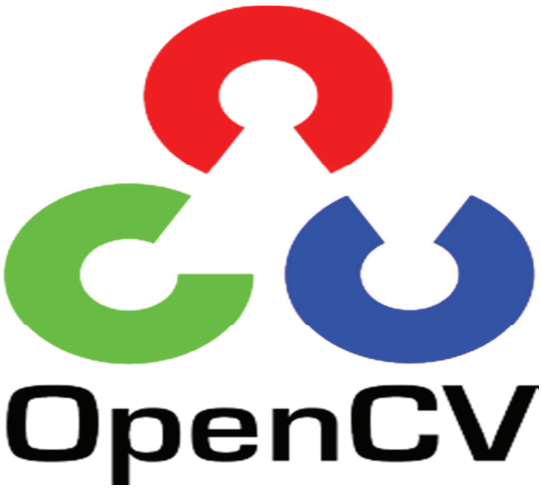
FigureIV.3 : Fonctionnement de Qt

IV.3.5 La Bibliothèque OpenCV

OpenCV (**O**pen Source [1]**C**omputer **V**ision) est une bibliothèque graphique libre, initialement développée par Intel, spécialisée dans le *traitement d'images* en temps réel.

Officiellement lancée en 1999, elle a été écrite en C et C++, et tire profit des processeurs multi cœurs.

C'est la bibliothèque de référence pour la vision par ordinateur, aussi bien dans le monde de la recherche que celui de l'industrie.



FigureIV.4 : Logo d'OpenCV

[1] Correspond à une licence de logiciel obéissant à une définition très précise établie par l'open source initiative, dont les principaux critères sont : 1\La libre redistribution, 2\Un code source disponible, 3\Les travaux dérivés disponibles.

IV.4 Les interfaces de « FingerPrint Recognition System »

Nous avons implémenté deux applications :

- L'application Administrateur
- L'application Utilisateur

1. Interface de démarrage de l'application Administrateur :

FigureIV.5 : Fenêtre de démarrage de l'application Administrateur

Elle permet d'accéder à l'espace Administrateur après s'être authentifié avec un mot de passe.

FigureIV.6 : Interface Espace Administrateur

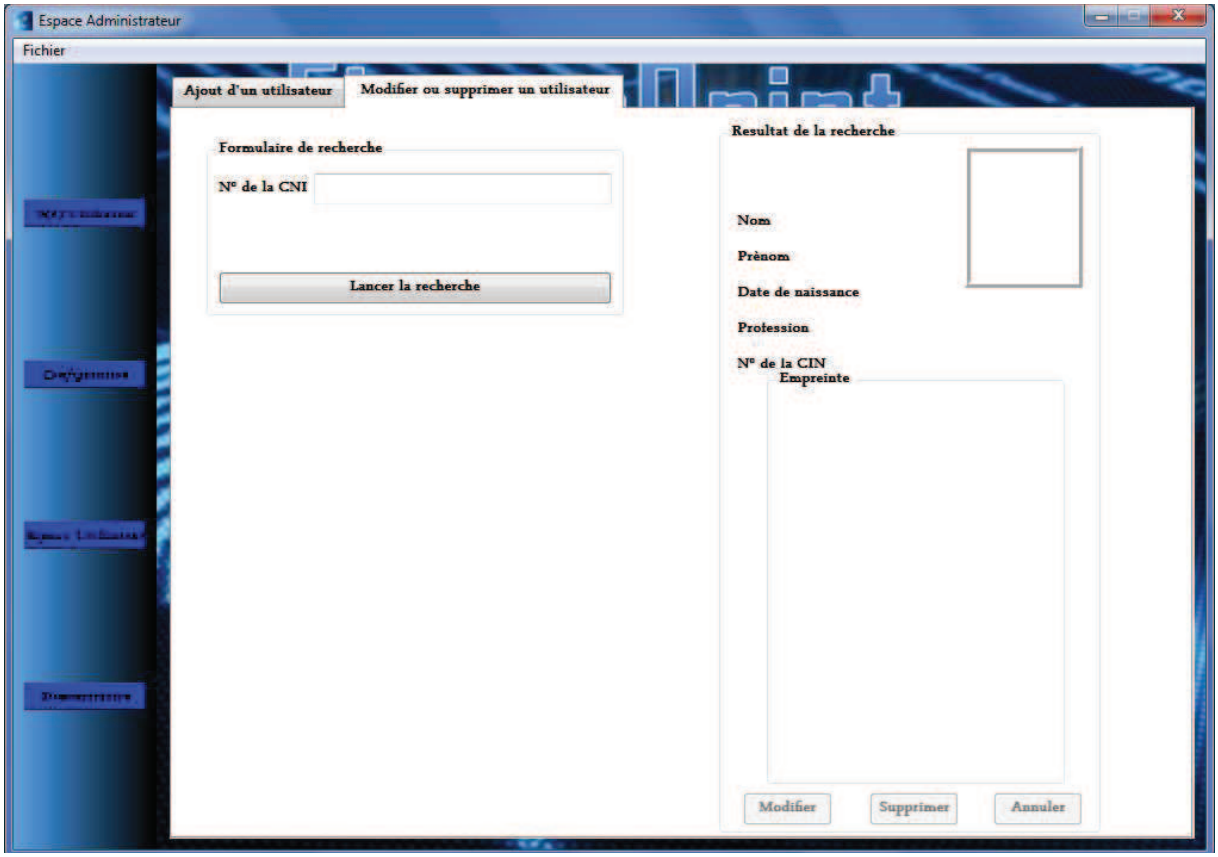
Une fois dans cet espace, l'administrateur peut accéder à plusieurs interfaces en cliquant sur les boutons du menu figé de gauche. Il peut soit :

- Afficher l'interface d'enrôlement en cliquant sur le bouton *MAJ Utilisateur* :

FigureIV.7 : Interface d'ajout

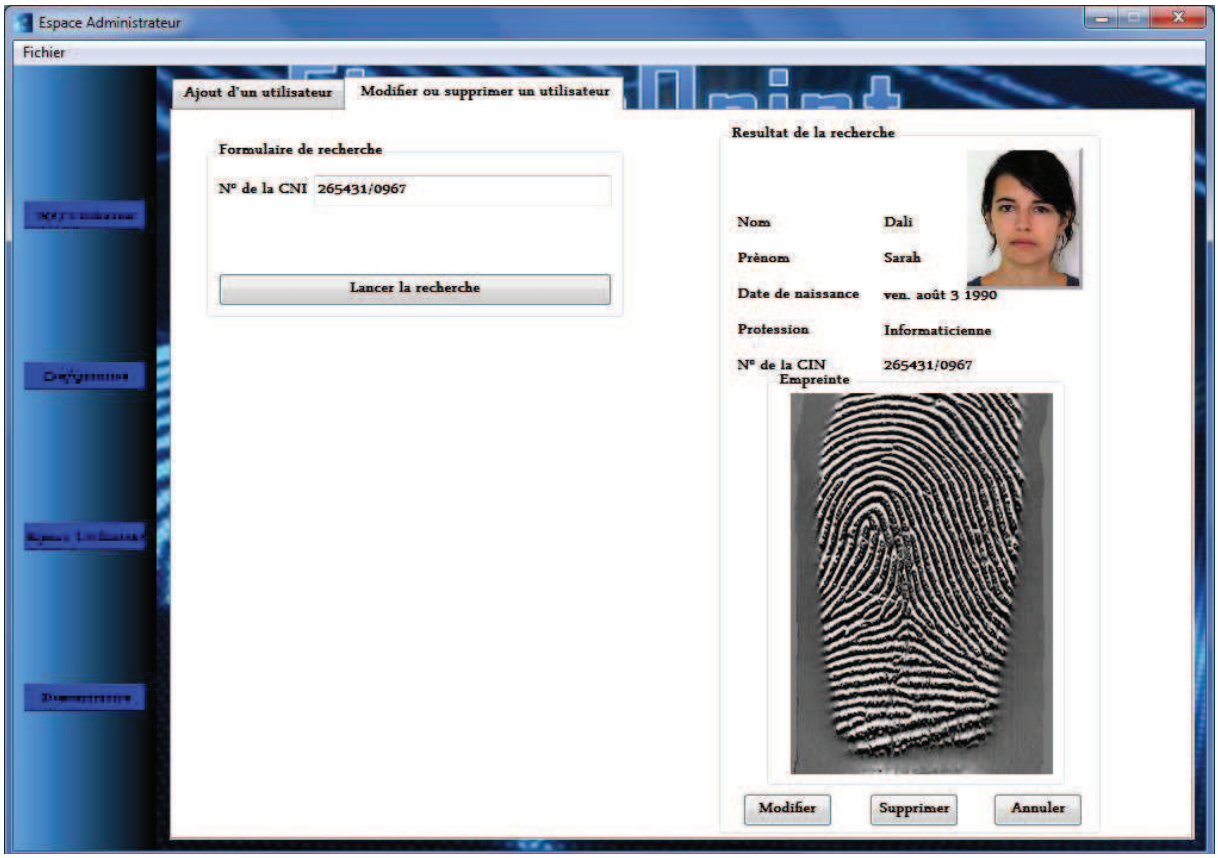
Après avoir rempli le formulaire d'ajout et charger l'empreinte digitale, l'administrateur clique sur le bouton *Enregistrer* et le résumé des informations saisies s'affiche.

- Afficher l'interface de modification / suppression, en cliquant sur le bouton *MAJ Utilisateur*, ensuite sur l'onglet *modification / suppression :*



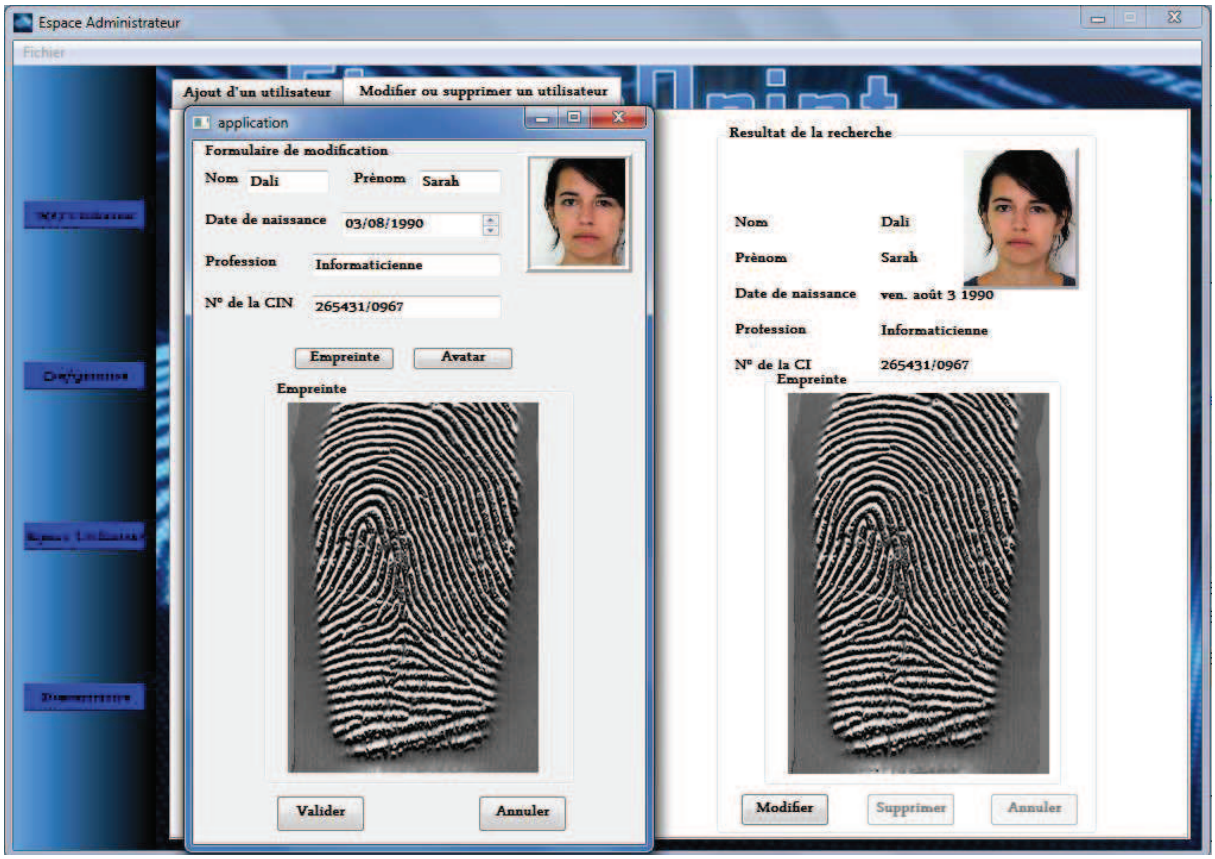
FigureIV.8 : Interface de Modification / Suppression

Une fois dans cette interface, l'administrateur peut effectuer une recherche par N° CIN de l'utilisateur à modifier ou à supprimer.



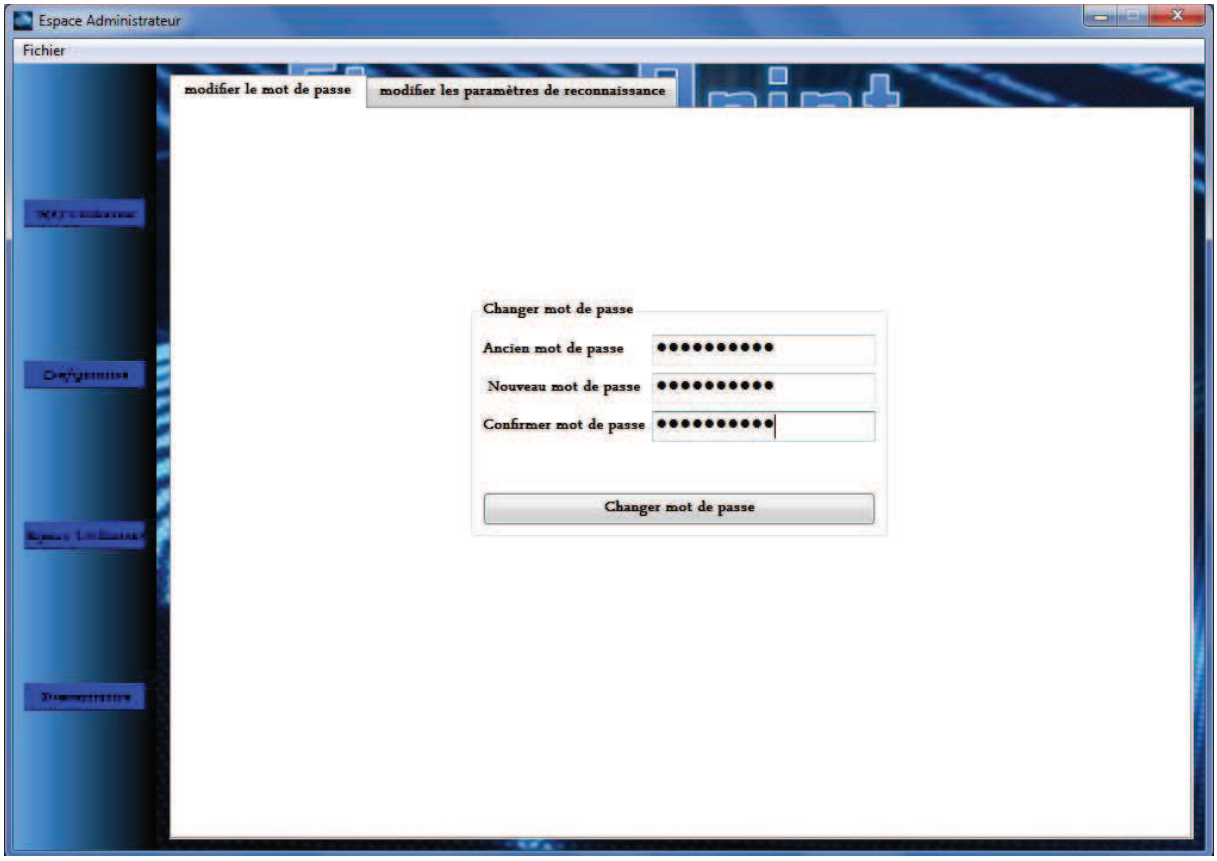
FigureIV.9 : Interface de recherche

En cliquant sur le bouton *Modifier,* une petite fenêtre contenant un formulaire s'affiche pour qu'il puisse saisir les nouvelles informations de l'utilisateur choisi.



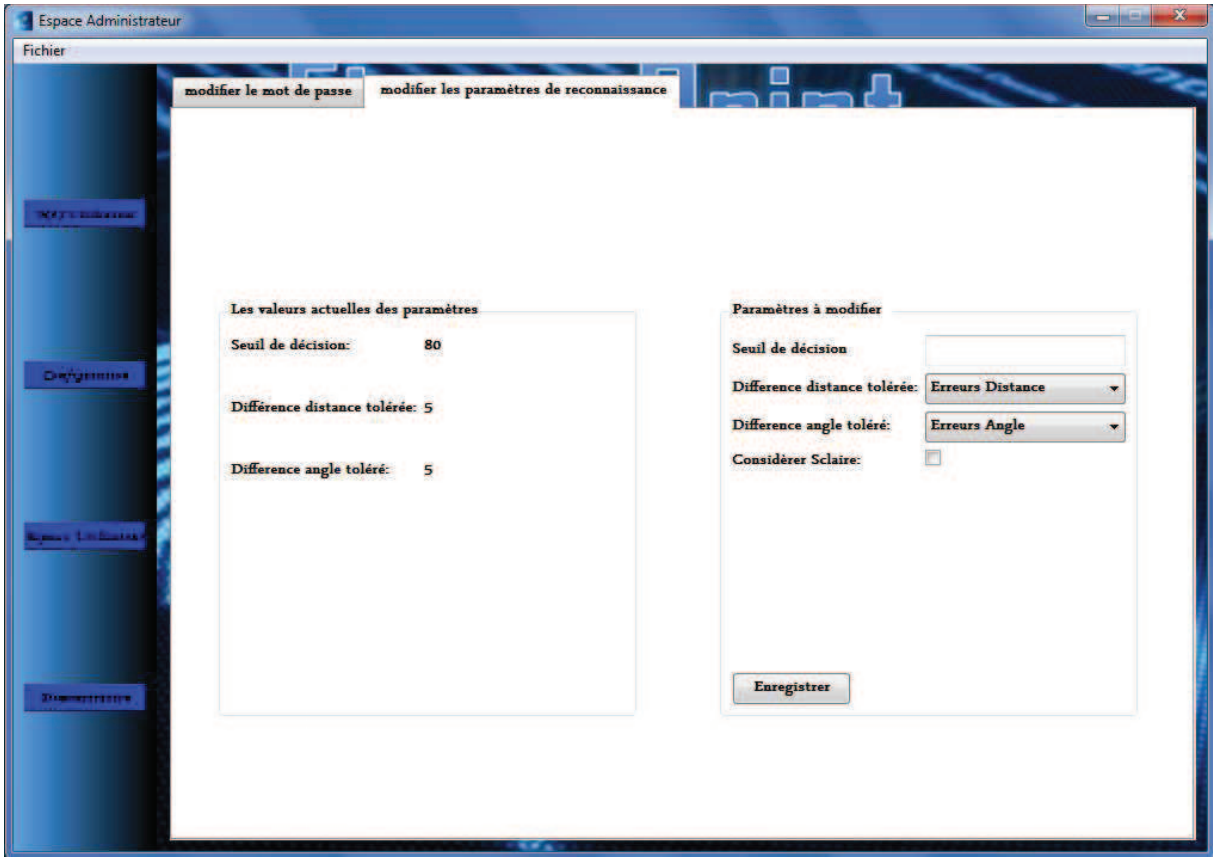
FigureIV.10 : Formulaire de modification

- Afficher l'interface de modification du mot de passe en cliquant sur le bouton *Configuration*



FigureIV.11 : Interface de modification du mot de passe

- Afficher l'interface de configuration des paramètres de reconnaissance, en cliquant sur le bouton *Configuration* ensuite sur l'onglet *Paramètres de reconnaissances* :



FigureIV.12 : Interface de configuration des paramètres de reconnaissance

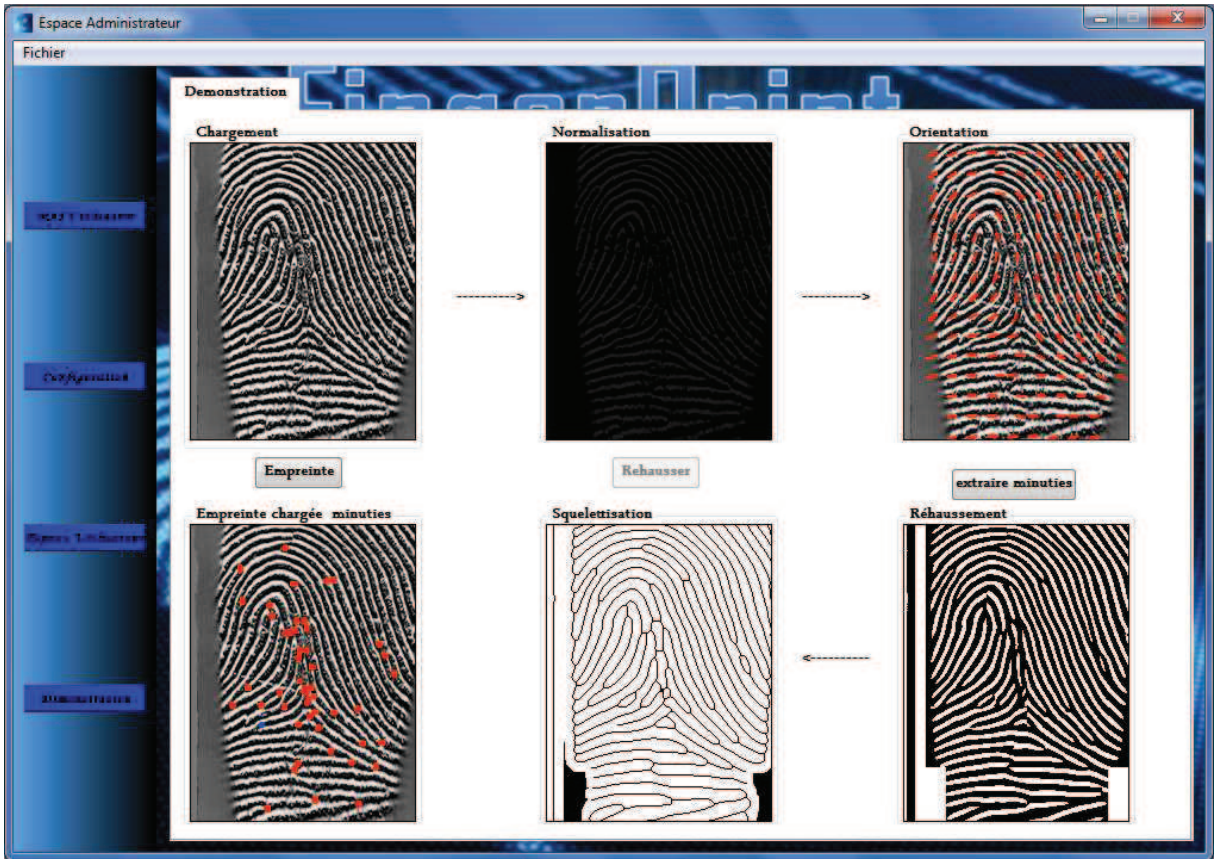
Elle permet de consulter et / ou modifier les paramètres du système de reconnaissance :

- Seuil de décision
- La rotation
- La translation
- Le scalaire

Les trois derniers paramètres servent à compenser les erreurs engendrées par le changement de position du doigt lors de la capture, et cela concerne la même empreinte.

- Afficher l'interface de l'espace utilisateur, en cliquant sur le bouton *Espace Utilisateur* (voir dans ce qui suit).

- Afficher l'interface de démonstration en cliquant sur le bouton *démonstration* :



FigureIV.13 : Interface Démonstration

Cette interface permet de suivre et de mieux comprendre l'étape de rehaussement d'une image d'empreinte digitale ainsi que le processus d'extraction des minuties et cela en illustrant chaque étape.

2. Interface de démarrage de l'application Utilisateur :

FigureIV.14 : Interface Espace Utilisateur

Le menu figé de gauche contient deux boutons qui permettent d'accéder aux interfaces suivantes :

- Interface d'identification en cliquant sur le bouton *Identification* :

FigureIV.15 : Interface d'Identification

L'utilisateur doit charger son empreinte digitale pour lancer l'identification.
Une fois l'identification terminée, la fiche d'information de la personne qui a obtenu le plus grand score d'appariement s'affiche.

- Interface d'authentification en cliquant sur le bouton *Authentification* :

FigureIV.16 : Interface d'Authentification

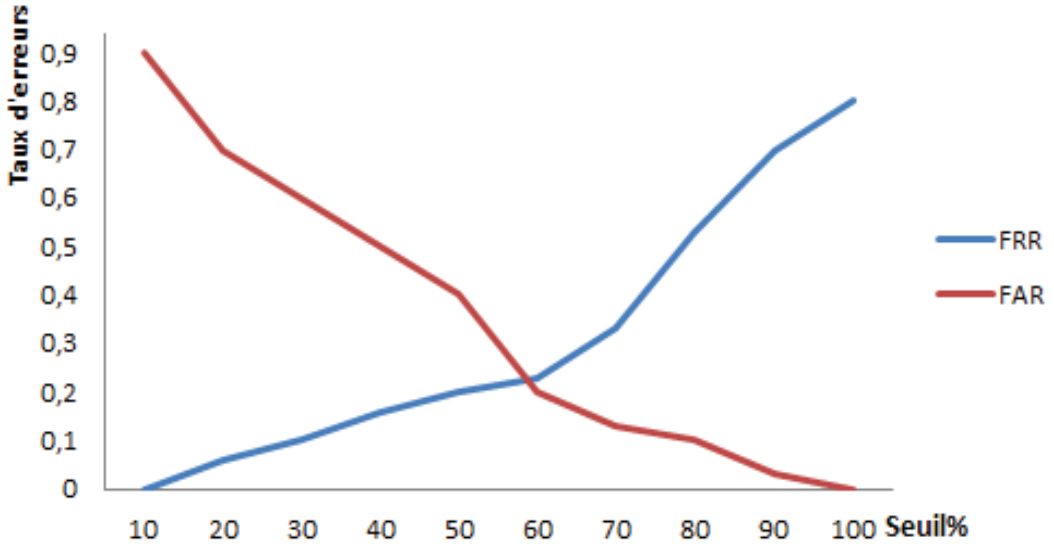
Pour s'authentifier, l'utilisateur doit entrer son N° CIN et charger son empreinte. Si le système trouve un résultat, une fiche contenant les informations de la personne s'affiche ainsi que le score d'appariement.

IV.5 Evaluation de notre système de reconnaissance

Nous avons effectué sur notre application Utilisateur, une série de tests permettant de tracer la courbe ROC de notre système biométrique.
Les images d'empreintes utilisées proviennent de la base de données FVC[2]2004 [44] (30 personnes possédant plusieurs empreintes du même doigt).

Cette courbe permet, d'évaluer les performances du système de reconnaissance. Elle est réalisée en calculant le couple (FAR, FRR).

La figure suivante, montre le résultat de nos tests :



FigureIV.17 : Variation des taux d'erreurs en fonction du seuil de décision

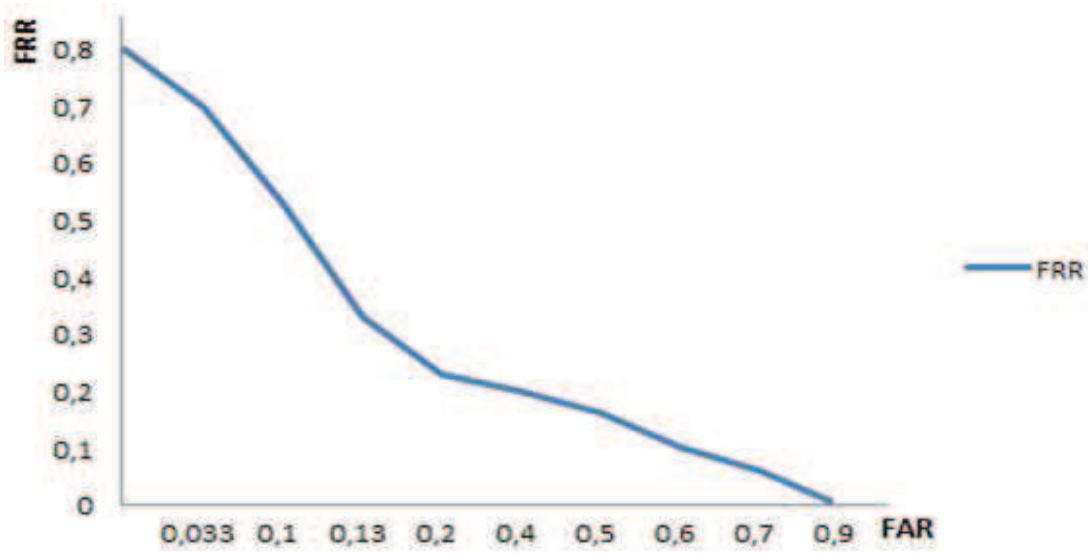
On remarque l'existence d'une relation inverse entre le couple (FAR, FRR) :

- A chaque fois que le seuil de décision est bas la valeur du FAR augmente, et celle du FRR diminue.
- A chaque fois que le seuil de décision est haut la valeur du FAR diminue, et celle du FRR augmente.

Lorsque le FRR=FAR, on dit que le système est en équilibre.

[2] First International Competition for Fingerprint Verification Algorithms

La courbe ROC de notre système de reconnaissance d'empreintes digitales est comme suit:



FigureIV.18 : Courbe ROC de notre système de reconnaissance d'empreintes digitales

Les faux rejets, sont dus à la variation des minuties entre les différentes empreintes d'une même personne, lors de la capture. Par contre, les fausses acceptations sont dues à la ressemblance de quelques minuties qui appartiennent aux empreintes des différentes personnes.

IV.6 Conclusion

Dans ce chapitre nous avons en premier lieu présenté les langages et outils utilisés pour implémenter nos deux applications, puis nous avons décrit une partie de leur fonctionnement en l'illustrant avec quelques interfaces, et conclu en effectuant une série de tests.

CONCLUSION GENERALE

CONCLUSION GÉNÉRALE

Ce mémoire de fin d'études, nous a conduit à concevoir et à réaliser un système de vérification et d'identification par empreintes digitales, en implémentant l'approche basée minuties qui est très utilisées pour ses différents avantages, principalement ses bons résultats et sa simplicité de mise en œuvre.

L'exécution de notre système de reconnaissance, a montré des résultats corrects sur une base de tests effectués en utilisant des images d'empreintes digitales provenant de la base de données FVC2004.

Du point de vue pédagogique, ce travail nous a permis d'acquérir de nouvelles connaissances et de nous familiariser avec les concepts du domaine de traitement d'image, d'approfondir et de mieux maitriser la conception orientée objet, avec la modélisation UML, ainsi que l'environnement de développement d'application *Visual Studio 2010 Ultimate*.

Ce thème reste ouvert pour d'éventuelles améliorations par d'autres étudiants intéressés par les techniques biométriques notamment l'amélioration de la performance du système de reconnaissance en le rendant multimodale, soit :

- En combinant plusieurs algorithmes d'appariement d'empreintes digitales
- En utilisant plusieurs images d'empreintes (ex : pouce, index) pour reconnaitre une personne
- En le combinant avec d'autres modalités biométriques (ex : visage)

Nous espérons que ce présent travail puisse leurs servir de base.

ANNEXE 1

1. Définition de notions liées à la numérisation d'images :

- ***Bruit :*** Le bruit (parasite) sur une photo numérique correspond aux pixels parasites. Ce phénomène de bruit est la cause de multiples facteurs :
  - une luminosité trop forte ou trop faible
  - un matériel (objectif ou appareil photo) de mauvaise qualité ou abîmé

  Les pixels sur la photo se superposent ou s'interposent.
- ***Définition d'une image*** : elle est définie par le nombre de points la composant. En image numérique, cela correspond au nombre de pixels qui compose l'image en hauteur (axe vertical) et en largeur (axe horizontal) : *200 pixels par 450 pixels* par exemple, abrégé en « 200 × 450 ».
- ***Image en couleur :*** Il existe plusieurs modes de représentation de la couleur, le plus utilisé pour le maniement des images numériques est celui qui représente les couleurs à l'aide de leurs composantes primaires (RVB).

  Aussi il existe différentes façons de sauvegarder les images, qui sont spécifiées par le format, comme le format BMP, JPEG, etc.

  Chacun des pixels de l'image est codé sur :
  - -Un bit dans le cas des images monochromes
  - 4 bits dans le cas des images de 16 couleurs
  - 8 bits dans le cas des images de 256 couleurs
  - 24 bits dans le cas des images de 16 millions de couleurs
- ***Les images en teintes de gris***

  En général, les images en niveaux de gris renferment 256 teintes de gris. Image `a 256 couleurs, simplement chacune de ces 256 couleurs est définie dans la gamme des gris.

  Par convention la valeur zéro représente le noir (intensité lumineuse nulle) et la valeur 255 le blanc (intensité lumineuse maximale).
- ***Image numérique :*** toute image (dessin, icone, photographie…) acquise, créée, traité et stockée sous forme binaire. Elle est composée d'unités élémentaires (pixels).
- ***Voisinage :*** c'est l'ensemble de pixels entourant un pixel donné, généralement en nombre de huit, cette notion est très importante dans le traitement de l'image numérique comme le filtrage.

- ***Résolution d'une image*** *:* elle se définie par le nombre de point par pouce ou le nombre de pixel par unité de longueur. La résolution de l'image est définie en PPI (Pixel Per Inch) ou PPP (Pixel Par Pouce).
- ***Pixel*** *:* souvent abrégé px, est l'unité de base permettant de mesurer la définition d'une image numérique matricielle. Son nom provient de la locution anglaise « ***pict****ure* ***el****ement* », qui signifie « élément d'image ».

  À chaque pixel est associée une couleur, usuellement décomposée en trois composantes primaires (voir RVB).

- ***RVB*** (Rouge Vert Bleu) ou RGB (Red Green Blue) sont des couleurs primaires additives. Ces trois couleurs permettent 256 niveaux d'intensité de couleurs.

2. Définition de notions liées à la biométrie :

- **1 contre 1** : Voir authentification
- **1 contre n** : Voir identification
- **Algorithme de Matching** : Procédé mathématique permettant d'effectuer la comparaison de deux échantillons biométriques
- **Authentification** : Procédé permettant de vérifier l'identité d'une personne. Il comprend deux étapes :

  1. L'utilisateur fournit un identifiant « id » au système de reconnaissance (par exemple un numéro d'utilisateur)
  2. L'utilisateur fournit ensuite un échantillon biométrique qui va être comparé à l'échantillon biométrique correspondant à l'utilisateur « id » contenu dans la base de données biométrique du système. Si la comparaison correspond, l'utilisateur est authentifié

- **Biométrie comportementale** : Il s'agit d'un type de biométrie caractérisée par un trait d'attitude qui est appris et acquis au fil du temps (par exemple sa façon de signer un document, de marcher, d'utiliser un clavier...) plutôt que par une caractéristique physiologique

- **Capteur biométrique** : Dispositif d'acquisition permettant d'obtenir une représentation numérique d'un élément du corps humain
- **Capture** : méthode de collecte d'un échantillon biométrique d'un utilisateur
- **Caractéristique Biométrique** : La plupart des systèmes biométriques ne compare pas directement les données acquises (image, son, etc.). On utilise plutôt différentes méthodes mathématiques pour extraire une quantité de données moins importante, mais contenant l'essentiel de l'information permettant de différencier deux individus (par exemple les minuties dans le cas de l'empreinte digitale). Ces données sont des éléments caractéristiques.
- **CIP/NIP** : Code/Numéro d'Identification Personnel. En anglais PIN « Personnal Identification Number ». Combinaison de chiffres et/ou de lettres destinés à l'identification d'une personne. Le CIP est unique ; il ne peut qualifier qu'un seul individu. Les codes confidentiels de cartes bancaires ne sont pas des CIP puisqu'un même code peut être attribué à plusieurs personnes
- **Comparaison** : Processus d'évaluation de correspondance d'un échantillon biométrique avec un ou plusieurs modèle(s) de référence précédemment stocké(s)
- **Correspondance** : Processus de comparaison d'un échantillon biométrique avec une référence déjà stockée et évaluation du degré de similarité. Une décision d'acceptation ou de rejet est fondée sur le dépassement ou non du seuil par le score
- **Donnée biométrique** : Information extraite d'un échantillon biométrique et utilisé soit pour construire un modèle de référence ou pour comparer à des modèles existants
- **Échantillon biométrique** : Représentation sous forme numérique d'un élément du corps humain. On obtient un échantillon biométrique à l'aide d'un capteur biométrique
- **Empreinte digitale** : Motif formé par les crêtes et les vallées du relief cutané
- **Empreinte latente** : Trace d'empreinte laissée sur un objet après contact entre celui-ci et un doigt. C'est ce type d'empreintes qui est relevé sur les scènes de crime
- **Enrôlement** : Étape initiale au cours de laquelle sont capturées les données biométriques qui serviront de références lors des authentifications ou identifications futures. C'est aussi lors de cette étape qu'un identifiant est associé aux données biométriques de chaque personne. Un soin tout particulier doit être apporté à cette première capture, car c'est sa qualité qui déterminera les performances futures du système

- **Extraction** : Processus de conversion d'un échantillon biométrique capturé en donnée biométrique pouvant être comparée au modèle de référence
- **Fausse Acceptation** : Événement ayant lieu lorsqu'un système biométrique accepte une personne alors qu'elle n'est pas dans sa base d'utilisateurs. Cet événement doit être le plus rare possible pour assurer la sécurité d'un système biométrique
- **Faux Rejet** : Événement ayant lieu lorsqu'un système biométrique refuse une personne alors qu'elle est dans sa base d'utilisateurs. Cet événement est souvent dû à une mauvaise acquisition des données biométriques et est perçu comme une gêne par l'utilisateur
- **Identification** : Procédé permettant de déterminer l'identité d'une personne. Il ne comprend qu'une étape :
    - L'utilisateur fournit un échantillon biométrique qui va être comparé à tous les échantillons biométriques contenus dans la base de données biométriques du système. Si l'échantillon correspond à celui d'une personne de la base, on renvoie son numéro d'utilisateur. Sinon l'identification échoue
- **Gabarit** : En anglais : **Template**, Modèle initial créé au cours de l'enrôlement. Modèle mathématique décrivant certaines caractéristiques physiques ou comportementales d'un individu. On comparera par la suite les demandes de reconnaissance à ce modèle
- **Minuties** : Petites imperfections dans le flot des lignes cutanées d'une empreinte digitale. Il en existe différents types (îlot, lacs, etc.) mais seules deux sont utilisés dans les applications informatiques de reconnaissance d'empreintes : les fins de lignes et les bifurcations.
- **Modèle de référence** : Donnée représentant une caractéristique biométrique d'un individu utilisée par un système biométrique pour permettre la comparaison avec des échantillons soumis a posteriori
- **Moteur biométrique** : Ensemble d'algorithmes permettant l'enrôlement, le matching, ainsi que toutes les étapes intermédiaires du procédé de reconnaissance d'un élément biométrique (amélioration des images, détermination de la qualité, extraction des caractéristiques discriminantes, etc.)
- **ROC Curve** (Receiver Operating Caracteristics curve) : Dans le cadre biométrique, cette courbe représente l'évolution du FRR en fonction du FAR. L'étude de cette courbe permet de déterminer les performances d'un système biométrique

- **Seuil de décision** : L'acceptation ou rejet d'une donnée biométrique dépend du passage du score de correspondance au-dessus ou au-dessous du seuil. Ce dernier est ajustable pour rendre le système biométrique plus ou moins strict, cela dépend des éléments requis par tout système application biométrique
- **Seuil de rejet** : Score minimum en dessous duquel un algorithme biométrique rejettera une authentification/identification
- **Seuil d'acceptation** : Score au-dessus duquel un algorithme biométrique acceptera une authentification/identification

- **Système Biométrique** : Dispositif automatisé permettant de :

    1. Acquérir des données biométriques
    2. Extraire des informations discriminantes à partir de données
    3. Comparer ces informations avec celles contenues dans un ou plusieurs gabarits servant de référence.
    4. Décider s'ils correspondent
    5. Indiquer à l'utilisateur si l'authentification ou l'identification a réussi ou échoué

- **TFA - Taux de fausse acceptation** : Indique la probabilité qu'un utilisateur inconnu soit identifié comme étant un utilisateur connu. Ce taux définit la sécurité du système biométrique
- **TFR - Taux de faux rejet** : Indique la probabilité qu'un utilisateur connu soit rejeté par le système biométrique. Ce taux définit en partie le confort d'utilisation du système biométrique
- **TEE - Taux d'égale erreur** : Donne un point pour lequel le TFA est égal au TFR

ANNEXE 2

UML « Unified Modeling Language »

Introduction

Pour faire face à la complexité croissante des systèmes d'information, de nouvelles méthodes et outils ont été créées. La principale avancée des quinze dernières années réside dans la programmation orientée objet (P.O.O.).

Face à ce nouveau mode de programmation, les méthodes de modélisation classique (telle que MERISE) ont rapidement montré certaines limites et ont dû s'adapter.

De très nombreuses méthodes ont également vu le jour comme Booch, OMT …

Dans ce contexte et devant le foisonnement de nouvelles méthodes de conception « orientée objet », l'Object Management Group (OMG) a eu comme objectif de définir une notation standard utilisable dans les développements informatiques basés sur l'objet. C'est ainsi qu'est apparu UML (Unified Modeling Language « langage de modélisation objet unifié »), qui est issu de la fusion des méthodes Booch, OMT (Object Modeling Technique) et OOSE (Object Oriented Software Engineering).

Issu du « terrain » et fruit d'un travail d'experts reconnus, UML est le résultat d'un large consensus. De très nombreux acteurs industriels de renom ont adopté UML et participent à son développement.

Comme UML n'impose pas de méthode de travail particulière, il peut être intégré à n'importe quel processus de développement logiciel de manière transparente. UML est une sorte de boite à outils, qui permet d'améliorer progressivement vos méthodes de travail, tout en présentant vos modes de fonctionnement.

En l'espace d'une poignée d'années seulement, UML est devenu un standard incontournable, et nous donnerons dans ce qui suit une brève présentation de ce langage.

1. L'approche orientée objet [45]

L'approche orientée objet considère le logiciel comme une collection d'objets dissociés, et identifiés, définis par des propriétés (attributs i.e. une donnée caractérisant l'état de l'objet). La fonctionnalité du logiciel émerge alors de l'interaction entre les différents objets qui le constituent. L'une des particularités de cette approche est qu'elle rapproche les données et leurs traitements associés au sein d'un unique objet.

Comme nous venons de le dire, un objet est caractérisé par plusieurs notions :

- **L'identité –** L'objet possède une identité, qui permet de le distinguer des autres objets, indépendamment de son état. On construit généralement cette identité grâce à un identifiant découlant naturellement du problème (par exemple un produit pourra être repéré par un code, une voiture par un numéro de série, etc.)
- **Les attributs –** Il s'agit des données caractérisant l'objet. Ce sont des variables stockant des informations sur l'état de l'objet.
- **Les méthodes –** Les méthodes d'un objet caractérisent son comportement, c'est-à-dire l'ensemble des actions (appelées opérations) que l'objet est à même de réaliser. Ces opérations permettent de faire réagir l'objet aux sollicitations extérieures (ou d'agir sur les autres objets). De plus, les opérations sont étroitement liées aux attributs, car leurs actions peuvent dépendre des valeurs des attributs, ou bien les modifier.

La difficulté de cette modélisation consiste à créer une représentation abstraite, sous forme d'objets, d'entités ayant une existence matérielle (voiture, ampoule, personne, . . .) ou bien virtuelle (client, temps, . . .).
La Conception Orientée Objet (COO) est la méthode qui conduit à des architectures logicielles fondées sur les objets du système, plutôt que sur la fonction qu'il est censé réaliser.

1.1 Concepts importants de l'approche objet

L'approche objet rapproche les données et leurs traitements. Mais cette approche ne fait pas que ça, d'autres concepts importants sont spécifiques à cette approche et participent à la qualité du logiciel.

- **Notion de classe**

Tout d'abord, introduisons la notion de classe. Une classe est un type de données abstrait, caractérisé par des propriétés (attributs et méthodes) communes à toute une famille d'objets et permettant de créer (instancier) des objets possédant ces propriétés. Les autres concepts importants qu'il nous faut maintenant introduire sont l'encapsulation, l'héritage et l'agrégation.

- **Encapsulation**

L'encapsulation consiste à masquer les détails d'implémentation d'un objet, en définissant une interface.
L'interface est la vue externe d'un objet, elle définit les services accessibles (offerts) aux utilisateurs de l'objet.

L'encapsulation facilite l'évolution d'une application car elle stabilise l'utilisation des objets : on peut modifier l'implémentation des attributs d'un objet sans modifier son interface, et donc la façon dont l'objet est utilisé.

L'encapsulation garantit l'intégrité des données, car elle permet d'interdire, ou de restreindre, l'accès direct aux attributs des objets.

- **Héritage, Spécialisation, Généralisation et polymorphisme**

L'héritage est un mécanisme de transmission des propriétés d'une classe (ses attributs et méthodes) vers une sous-classe. Une classe peut être spécialisée en d'autres classes, afin d'y ajouter des caractéristiques spécifiques ou d'en adapter certaines. Plusieurs classes peuvent être généralisées en une classe qui les factorise, afin de regrouper les caractéristiques communes d'un ensemble de classes.

Ainsi, la spécialisation et la généralisation permettent de construire des hiérarchies de classes. L'héritage peut être simple ou multiple. L'héritage évite la duplication et encourage la réutilisation.

Le polymorphisme représente la faculté d'une méthode à pouvoir s'appliquer à des objets de classes différentes. Le polymorphisme augmente la généricité, et donc la qualité, du code.

- **Agrégation**

Il s'agit d'une relation entre deux classes, spécifiant que les objets d'une classe sont des composants de l'autre classe. Une relation d'agrégation permet donc de définir des objets composés d'autres objets.

L'agrégation permet donc d'assembler des objets de base, afin de construire des objets plus complexes.

2. Présentation générale d'UML [45] [46]

2.1 Historique

UML est une norme du langage de modélisation objet qui a été publiée, dans sa première version, en novembre 1997 par l'OMG (Object Management Group), instance de normalisation internationale du domaine de l'objet.

En quelques années, UML s'est imposée comme standard à utiliser en tant que langage de modélisation objet.

Les grandes étapes de la diffusion d'UML peuvent se résumer comme suit :

1994-1996 : rapprochement des méthodes OMT, BOOCH et OOSE et naissance de la première version d'UML.

23 novembre 1997 : version 1.1 d'UML adoptée par l'OMG.

1998-1999 : sortie des versions 1.2 à 1.3 d'UML.

2000-2001 : sortie des dernières versions suivantes 1.x.

2002-2003 : préparation de la version 2.

10 octobre 2004 : sortie de la version 2.1.

5 février 2007 : sortie de la version 2.11

2.2 Définition d'UML

UML n'est pas une méthode (i.e. une description normative des étapes de la modélisation) : ses auteurs ont en effet estimé qu'il n'était pas opportun de définir une méthode en raison de la diversité des cas particuliers. Ils ont préféré se borner à définir un ensemble de notations graphiques (modèles) qui s'appuient sur une syntaxe (méta modèle). L'objectif initial était de permettre aux informaticiens de représenter un système logiciel et son utilisation prévue dans l'entreprise, afin d'améliorer la qualité des applications informatiques qu'ils développaient. Le méta modèle permet notamment d'assurer que tous les outils de génie logiciel orientés UML présentent une cohérence et autorisent l'interopérabilité des modèles.

2.3 Présentation générales des diagrammes d'UML

UML s'articule autour de treize types de diagrammes, chacun d'eux étant dédié à la représentation des concepts particuliers d'un système logiciel. Ces types de diagrammes sont répartis en deux grands groupes : les diagrammes structurels et les diagrammes comportementaux

- **Les diagrammes structurels sont** :

Diagramme de classes :

Il est généralement considéré comme le plus important dans un développement orienté objet. Il représente l'architecture conceptuelle du système : il décrit les classes que le système utilise, ainsi que leurs liens.

Diagramme d'objets :

Il permet de mettre en évidence des liens entre les objets. Les objets, instances de classe, sont reliés par des liens, instance d'associations.

A l'exception de la multiplicité, qui est explicitement indiquer, le diagramme d'objets utilise les mêmes concepts que le diagramme de classes. Ils sont essentiellement utilisés pour comprendre ou illustrer des parties complexes d'un diagramme de classes.

Diagramme de packages (nouveau dans UML 2) :

Il montre l'organisation logique du modèle et les relations entre packages.

Diagramme de structure composite (nouveau dans UML 2) :

Il montre l'organisation interne d'un élément statique complexe.

Diagramme de composants :

Ils permettent de décrire l'architecture physique et statique d'une application en termes de modules : fichiers sources, librairies, exécutables, etc. Ils montrent la mise en œuvre physique des modèles de la vue logique avec l'environnement de développement.

Diagramme de déploiement :

Il montre la disposition physique des différents matériels qui entrent dans la composition d'un système et la répartition des instances de composants, processus et objet qui « vivent » sur ces matériels. Les diagrammes de déploiement sont donc très utiles pour modéliser l'architecture physique d'un système.

- **Les diagrammes comportementaux sont**:

Diagramme de cas d'utilisation :

Il représente la structure des grandes fonctionnalités nécessaires aux utilisateurs du système. C'est le premier diagramme du modèle UML, celui où s'assure la relation entre l'utilisateur et les objets que le système met en œuvre.

Diagramme de séquence et de communication (anciennement appelé collaboration) :

Le diagramme de séquence représente la chronologique des opérations réalisées par un acteur. Il indique les objets que l'acteur va manipuler et les opérations qui font passer d'un objet à l'autre. On peut représenter les mêmes opérations par un diagramme de communication, qui est un graphe dont les nœuds sont des objets et les arcs (numérotés selon la chronologie) les échanges entre objets. En fait, le diagramme de séquence et le diagramme de communication sont deux vues différentes mais logiquement équivalentes (on peut construire l'une à partir de l'autre) d'une même chronologie. Ce sont des diagrammes d'interaction.

Diagramme de temps (nouveau dans UML 2):

Il fusionne les diagrammes d'états et de séquence pour montrer l'évolution de l'état d'un objet au cours du temps.

Diagramme d'activité :

Le Diagramme d'activité permet de représenter la dynamique du système d'information. Il représente les règles d'enchaînement des actions et décisions au sein d'une activité.

Diagramme de vue d'ensemble des interactions (nouveau dans UML 2) :

Il fusionne les diagrammes d'activité et de séquence pour combiner des fragments d'interaction avec des décisions et des flots.

Diagramme d'états-transitions :

Il représente la façon dont évoluent les objets appartenant à une même classe et font donc apparaitre l'ordonnancement des travaux.

Conclusion

Il faut retenir qu'UML fournit un moyen visuel standard pour spécifier, concevoir et documenter les applications orientées objets, en collectant ce qui se faisait de mieux dans les démarches méthodologiques préexistantes.

En fin de compte, l'intérêt de la normalisation d'un langage de modélisation tel que UML réside dans sa stabilité et son indépendance vis-à-vis de tout fournisseur d'outil logiciel.

Quant au langage lui-même, il a été conçu pour couvrir tous les aspects de l'analyse et de la conception d'un logiciel, en favorisant une démarche souple fondée sur les interactions entre les différentes vues que l'on peut avoir d'une application.

BIBLIOGRAPHIE

Bibliographie

[1] A.K Jain1.hong.S.Pankanti « Biometrie: Prosing Frontiers for Emerging Identification Market». Communication of the ACMP. pp 91-98. February 2000.

[2] Jean-Luc Dugelay AL. «Recent Advances In Biometric Person Authentification » IEEE Int. Conf. on Acoustics Speed and Signal Processing (ICASSP), Orlando, Florida, May 2002.

[3] http://www.biometricgroup.com

[4] Florent Perronnin, Jean-Luc Dugelay. «Introduction a la biometrie : Authentification des individus par traitement audio-Video », Institut Eurocom, Multimedia Communication Departement, Revue Traitement du signal, Vol. 19, N°4,2002.

[5] N. Galy «Etude d'un système complet de reconnaissance d'empreinte digitales par un capteur microsystéme a balayage », Thèse de doctorat, Institut National Polytechnique de Grenoble, Avril 2005

[6] G.O. Williams « Iris Recognition Technology », IEEE Aerospace and Electronics Systems Magazine, Vol.12 Issue 4, pp. 23-29, April 1997.

[7] R.P.Wildes « Iris recognition: an emerging biometric technology ». Proceeding of the IEEE, Vol. 85, Issue 9, pp. 1348-1363, September 1997.

[8] A.K. Jain, A.Ross,and S.Pankanti, « A prototype hand geometry-based verification system », Proc of 2nd International Conf on Audio- and Video-based Biometric Person Authentification, pp.166 171, March 1999

[9] W.Zhao R Chellappa PJ.Phillips A.Rosenfeld. « Face recognition: A literature survey » ACM Computing Surveys (CSUR). Vol. 35, Issue 4, December 2000.

[10] W.A Banett. « A survey of face recognition algorithms and testing results » Conference Record of the Thirty-First Asilomar Conference on Signals. Systems & Computers, pp. 301-305, 1997.

[11] http://www.retina-scan.com/retina_scan_technology.htm

[12] http://www.veinid.com

[13] S.J Vaughan-Nichols, «Voice authentification speaks to the marketplace », Computer, Vol. 37, Issue 3, pp. 13-15, March 2004.

[14] B.H.Juang, T. Chen, « The past, present, and future of speech processing », IEEE Signal Processing Magazine, Vol. 15, Issue 3, pp. 24-48, May 1998.

[15] R.Gaines W.Lisowski, S. Press, N. Shapiro, « Authentification by Keystroke Timing: Some Preliminary Results », Rand Report R-256NSF, Rand Corp 1980.

[16] L. L. Lee, T. Berger, E. Aviezer, «Reliable On-Line Human Signature Verification », IEEE Transaction on PAMI, Vol. 18, N°6, pp. 643-647, June 1996.

[17] Lily Lee, « Gait analysis for recognition and classification », Recherch Abstract of the Artificial Intelligence Laboratory at MTT, Cambridge-Massachusetts, 2002.

[18] S. Liu, M. Silveman, « A pratical Guide to Biometric Security Technology », IEEE Computer Society, IT Pro-Security, Janvier-Février 2001.

[19] A. K. Jain, L. Hong, S. Pankanti, «Biometrics: Promising Frontiers for Emerging Identification Market», Communications of the ACM, pp. 91-98, February 2000.

[20] C. Fredouille, J, Mariethoz, C. Jaboulet, J. Hennebert, J. F. Bonastre, C. Mokbel, F. Bimbot, « Behavior of a Bayesian Adaptation Method for Incremental Enrollment in speaker Verification », International Conference on Acoustics, Speech, and Signal Processing pp. 1197-1200, Istanbul, Turquie, 5-9 Juin 2000.

[21] L. Heck, N. Mirghaton, « On line Unsupervised Adaptation in Speaker Verification », International Conference on Spoken Language Processing, Vol. 2, pp. 454-457, Pékin, Chine, 16-20 Octobre 2000.

[22] P.Phillips, H.Hheonjoon, S.Rizvi, P.Rauss, « The FERET Evaluation Methodologie for face recognition Algorithms », IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.22, N°10, Octobre 2000

[23] Jamal Kharroubi, « Etude de Techniques de Classement; Machines à vecteurs Supports pour la Vérification Automatique du Locuteur », Ecole Nationale Supérieure des Télécommunications, 2002.

[24] www.clusif.asso.fr

[25] http://www.biometrie.online.fr

[26] Davide Maltoni, Dario Maio, Anil K. Salil Prabhakar, "Handbook of fingerprint recognition", Springer, New York, 2003

[27] Cours « Introduction à la biométrie », Boulbaba Ben Amor, Institut Telecom/Telecom Lille1, Département Info & Réseaux

[28] Lin Hong and al, An identification system using fingerprints, 1997

[29] http://fr.wikipedia.org/wiki/Biom%C3%A9trie#Technologies

[30] Francis Galton, Fingerprint, McMillan, London, 1892

[31] International Biometric Group, The Henry Classification, www.biometricgroup.com

[32] X. Xia and L. O'Gorman, "Innovations in fingerprint capture devices", Pattern Recognition, Vol.36, pp.361-369, 2003

[33] A.K. Jain, S. Prabhakar, L. Hong and S. Pankanti, Filter bank-Based Fingerprint Matching, IEEE Trans, on image processing, Vol. 9, N° 5, pp. 846-859, Mai 2000

[34] A. Ross, A. K. Jain ET J. Reisman. A Hybrid Fingerprint matcher, Pattern Recognition, 36(7): 1661-1673, 2003

[35] R. BELGUECHI, "Contribution à la reconnaissance d'empreintes digitales par une approche hybride", Institut National de formation en informatique (I.N.I), Année universitaire: 2005/2006

[36] D. Gabor, « Theory of communication » IEE93

[37] Lin Hong, Yifei Wan, Anil Jain: Fingerprint Image Enhancement: *Algorithm and Performence Evaluation,* Pattern Recognition and Image Processing Laboratory, Michigan State University, 1998

[38] A. Ravi Shankar Rao, « A Taxonomy For Texture Description and Identification », Springer Verlag, New York, 1990

[39] N. Ratha, S. Chen, A. Jain, "Adaptive Flow Orientation Based Texture Extraction in Fingerprint Images", Pattern Recognition, Vol.28, No; 11, pp.165751672, Nov, 1995

[40] Nicolas Galy, « Etude d'un système de reconnaissance d'empreintes digitales pour un capteur microsystème à balayage », 2005

[41] T. Y. Zhang and C. Y. Suen "A Fast Parallel Algorithm for Thinning Digital Patterns" Vol 27, N°3, March 1984

[42] Arcelli and Baja G.S.D., "A Width Independent Fast Thinning Algorithm," IEEE Transaction on pattern Analysis and Machine Intelligence, vol. 4, no. 7, pp. 463-474, 1984.

[43] Dusenge Tony, « La Reconnaissance des Empreintes Digitales », BA3-INFO Université Libre de Bruxelles, 25 mai 2009

[44] http://bias.csr.unibo.it/fvc2004

[45] Laurent Piechocki Frédéric Di Gallo, cours UML, Ecole Nationale des Ingénieurs des Travaux Agricole de Bordeaux Département Entreprise et Système Unité de formation Informatique et Génie des Equipement, Mars 2005

[46] Laurent AUDIBERT, Cours UML 2.0 (IUT, département informatique) Institut Universitaire de Technologie de Villetaneuse – Département Informatique