جامعة أبو بكر بلقايد – تلمسان

Université Abou Bakr Belkaïd de Tlemcen

Faculté de Technologie

Département de Génie Biomédical

Laboratoire de Recherche de Génie Biomédical

MEMOIRE DE PROJET DE FIN D'ETUDES

Pour l'obtention du Diplôme de

MASTER en GENIE BIOMEDICAL

Spécialité : **Informatique Biomédicale**

Présenté par : Hasnaoui Nassim Aboubakr

LA RECONNAISSANCE AUTOMATIQUE DES EMPREINTES DIGITALES

Soutenu le 25 mai 2016 devant le Jury

Mr	Boudefla Amine	*MAA*	Université de Tlemcen	Président
Mr	Abderrahim Mohammed El Amine	*MCA*	Université de Tlemcen	Encadreur
Mr	Benabdella Ali	*MAA*	Université de Tlemcen	Examinateur

Année universitaire 2015-2016

Je dédie ce mémoire

A mes parents pour leur confiance, leur soutien, leurs sacrifices et toutes les valeurs qu'ils ont su m'inculquer.
À la mémoire de mes grands-parents paternel.
Mes tantes et mes oncles bien aimés ainsi que mes cousins et cousines, avec toute mon affection et mon estime indéfinis et pour leurs précieux encouragements.
A tous mes amis et collègues en souvenir des bons moments passés ensemble.

Remerciement

Je remercie tout d'abord « Allah » le tout puissant, de m'avoir donnée le courage et la patience afin de mener à bien mon projet de master.
Je remercie chaleureusement ma mère qui est toujours à côté de moi dans les moments difficiles.

Je remercie considérablement mon encadreur M.ABDERRAHIM Mohamed Amine pour la qualité de son encadrement, sa patience, ses compétences et ses hauts caractères personnels que j'ai beaucoup appréciés.

Je tiens à remercier sincèrement les membres du jury qui me font le grand honneur d'évaluer ce travail.
Enfin, J'aimerais également souligner le support de tous les membres de ma famille. Grandes remerciements à mes meilleurs amis Mohamed et Abdenour et à tous ceux qui m'aiment et à tous ceux qui m'aident dans les moments difficiles.

Résumé:

La reconnaissance d'empreintes digitales est une technique biométrique mature pour toute application d'authentification ou de vérification d'individus. Dans ce projet de fin d'études, nous décrivons la conception et le développement d'un système automatique d'authentification d'identité par empreintes digitales qui consiste à implémenter le meilleur algorithme permettant de faire la comparaison entre plusieurs empreintes.

Abstract:

Fingerprint recognition is an important biometric technique for personal authentication or verification. In this project graduation, we describe the design and implementation of an automatic identity authentication system that uses fingerprints to authenticate the identity of an individual.

ملخص :

إنَ أُستخدام بصمة الأصبع للتعرف تعدُ من أهم التقنياتْ البيومترية لتعريفْ الأشخاصْ أو التأكدْ من الهوياتْ. في هذا البَحْث قُمنا بِدراسة و تَصْميمْ وتَطوير نظام المصادقةْ الآلية على بصماتْ الأصابع.

Table des matières

1- Introduction:

De nos jours, l'authentification devient un des points essentiels au niveau de la sécurité des contrôles d'accès dans les sociétés ou systèmes informatiques . La reconnaissance biométrique est utilisée dans bon nombre d'applications telles que la protection de l'accès à un ordinateur, un téléphone portable, une clé USB, un établissement, des cartes bancaires... De nombreuses technologies biométriques ont été développées, toutes basées sur les identificateurs biométriques (iris, voix, empreintes digitales, face, signature...). En effet, la biométrie est l'usage de différentes caractéristiques physiologiques et comportementales afin de réaliser une reconnaissance automatique d'un individu.

Ce sont ces caractéristiques qu'on appelle Identificateurs Biométriques. Ces derniers sont plus fiables que les systèmes classiques (clé, mot de passe. . .) dans la reconnaissance d'une personne car ils sont difficilement falsifiables. C'est la raison pour laquelle les systèmes biométriques sont actuellement de plus en plus sollicités.[1]
Les identificateurs biométriques peuvent être comparés selon certains facteurs : l'universalité (tous les êtres humains en possèdent), l'unicité ou individualité (sont uniques à chaque personne), la persistance ou permanence, la collectabilité ... Les identificateurs biométriques les plus utilisés sont les empreintes digitales grâce à leur individualité et persistance. En effet, les empreintes digitales sont uniques à chaque personne et ce dès sa naissance. De plus, elles demeurent inchangées pendant toute la vie de la personne [5]. Lorsqu'elles sont légèrement endommagées (par une blessure par exemple.
La figure 1 illustre la domination des systèmes biométriques des empreintes digitales sur le marché de la biométrie.

L'authentification est basée sur deux composantes:

- L'identification dont le rôle est de définir les identités d'un utilisateur.
- L'authentification permettant de vérifier les identités présumées des utilisateurs.

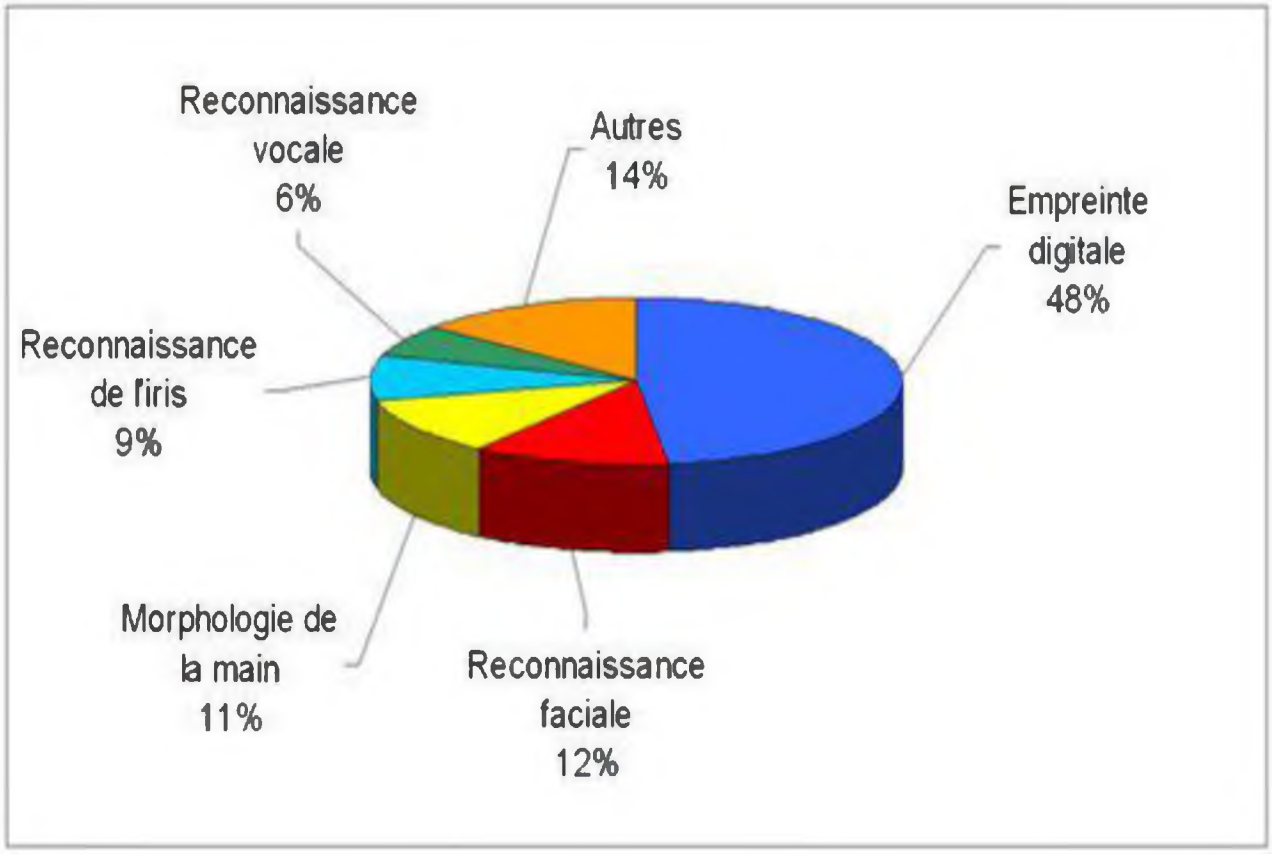


FIGURE 1:PARTAGE DU MARCHE DE LA BIOMETRIE DANS LE MONDE, EN 2008 (SOURCE : [2]).

1.1. Concepts de base:

En 1888, l'anthropologue anglais Francis Galton introduit la notion de minuties4 pour réaliser la comparaison d'empreintes digitales (fingerprintsmatching en anglais). Quatre ans plus tard, Galton publie son ouvrage (voir [3]) où il propose une classification rigoureuse des empreintes digitales et démontre qu'il y a seulement une chance sur 64 milliards que deux individus aient une même empreinte.[4]

Cet arrangement particulier des lignes papillaires forme des points caractéristiques, nommés minuties qui sont à l'origine de l'individualité des dessins digitaux .

A ce jour, on considère qu'il faut 8 à 17 de ces points sans discordance pour qu'on estime établie l'identification. Un chiffre inférieur au seuil minimum aboutit à l'exclusion de l'empreinte digitale comme élément de preuve.

Il existe 13 types de minuties qui nous permettent de classifier les empreintesdigitales.

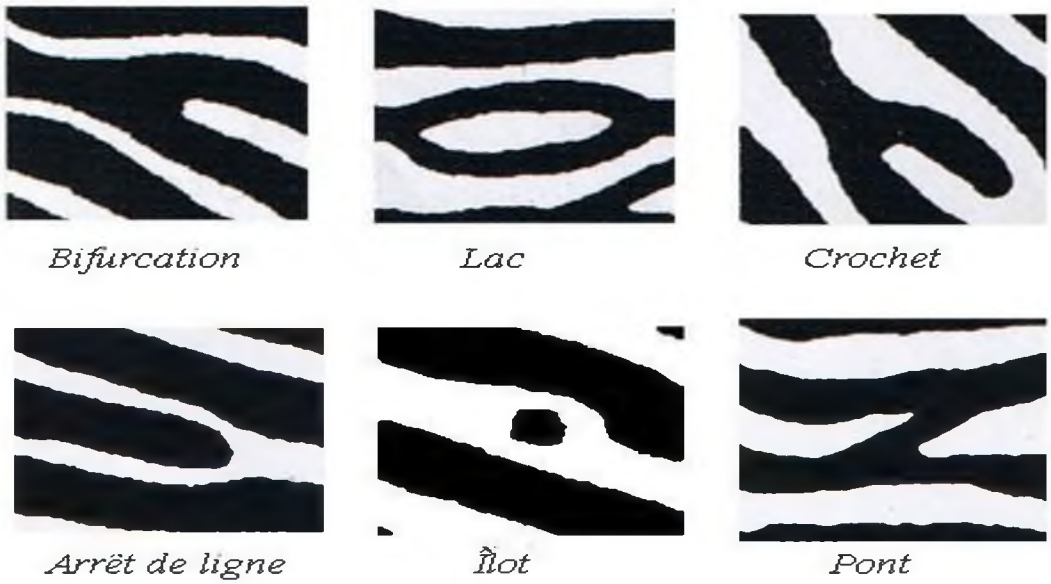


FIGURE 2: LES 6 PRINCIPAUX TYPES DE MINUTIES [5]

1.2. Vue globale du mécanisme de reconnaissance des empreintes digitales:

Le principe de la reconnaissance des empreintes digitales consiste à comparer une empreinte fournie au système, à une ou plusieurs autres empreintes (les modèles) dont le système dispose préalablement dans sa base de données biométrique. Le système biométrique renvoie un résultat positif au cas où l'empreinte fournie à l'entrée correspond à l'un des modèles, et un résultat négatif dans le cas contraire. La figure 3 illustre l'enregistrement préalable de modèles .

Lors de l'enregistrement, l'image scannée de l'empreinte est recueillie par le système, puis un contrôle de la qualité de l'image est effectué. En effet, une empreinte sérieusement endommagée (par une brûlure grave par exemple) est intraitable par le système [7]. Ensuite, une extraction de traits caractéristiques (généralement appelés minuties) est effectuée pour donner lieu au modèle final que le système sauvegarde dans la base de données. La méthode généralement utilisée pour détecter les minuties consiste à mettre l'image de l'empreinte en noir et blanc, c'est la binarisation de l'image, et à donner une même taille aux lignes de l'empreinte, c'est la squelettisation (voir figure 4) [7]. Une fois que l'on dispose de l'image binaire squellettisée, les minuties (singularités) sont mieux visibles ; on procède alors à leur détection. Nous verrons la détection de minuties en détail le chapitre 3 partie2.2 (page 37).

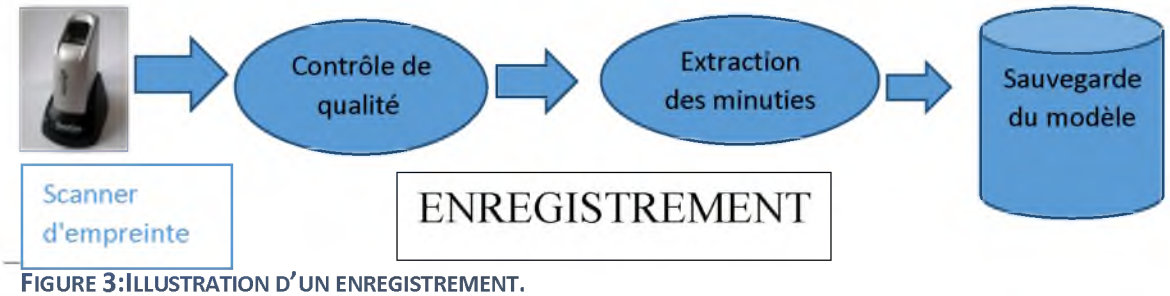


FIGURE 3:ILLUSTRATION D'UN ENREGISTREMENT.

1.3. *Approche générale*

Dans ce travail, nous étudions dans le chapitre 1 les étapes principales de la reconnaissance des empreintes, et les opérations nécessaires du traitement d'image, puis le processus de la reconnaissance.

Le chapitre 2 présentera quelques techniques biométriques qui existent dans la littérature, sesapplications et les détails sur la technique biométrique basée sur la reconnaissance des empreintes digitale

Dans le chapitre 3 on a appliqué les méthodes proposé dans le chapitre 2 et présenté des améliorations aux résultats obtenus par les algorithmes décrits en chapitre 2.

Nous terminons enfin par une conclusion générale.

CHAPITRE 1 : LA RECONNAISSANCE DES EMPREINTES

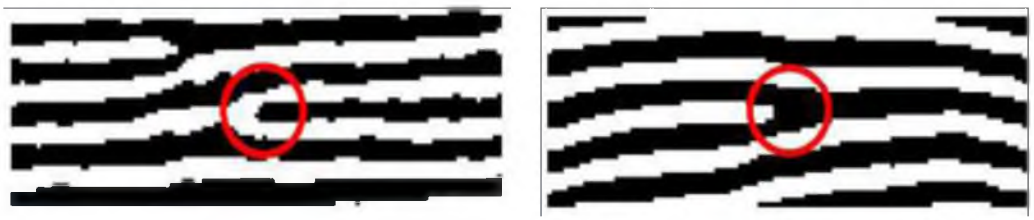
1- Concepts:

Une empreinte digitale est le dessin formé par les lignes de la peau des doigts, des paumes des mains, des orteils ou de la plante des pieds. Ce dessin se forme durant la période fœtale. Il existe deux types d'empreintes : l'empreinte directe (qui laisse une marque visible) et l'empreinte latente (saleté, sueur ou autre résidu déposé sur un objet). Elles sont uniques et immuables, elles ne se modifient donc pas au cours du temps (sauf par accident comme une brûlure par exemple) .[6]

Les empreintes sont composées, de terminaisons en crêtes, soit le point où la crête s'arrête, et de bifurcations, soit le point où la crête se divise en deux. Le noyau est le point intérieur, situé en général au milieu de l'empreinte. Il sert souvent de point de repère pour situer les autres minuties. D'autres termes sont également rencontrés : le lac, l'île, le delta, la vallée, la fin de ligne… Ces caractéristiques peuvent être numérisées. Une empreinte complète contient en moyenne une centaine de points caractéristiques mais les contrôles ne sont effectués qu'à partir de 12 points. Statistiquement, il est impossible de trouver 2 individus présentant 12 points caractéristiques identiques, même dans une population de plusieurs millions de personnes.

Donc éléments qui différencient les empreintes:[6]

Les minuties:



B): EXEMPLE D'ARRET DE RIDE TERMINAISON A):EXEMPLE DE BIFURCATION

FIGURE 4:EXEMPLE DES DEUX TYPE LES PLUS UTILISER [6]

D'après [7], La probabilité de trouver deux empreintes digitales similaires est de 1 sur 10 puissances 24. Les jumeaux, par exemple, venant de la même cellule, auront des empreintes très proches mais pas semblables.

2- L'algorithme de la reconnaissance d'empreintes digitales :

Le principe de la reconnaissance des empreintes digitales consiste à comparer une empreinte fournie au système, à une ou plusieurs autres empreintes dont le système dispose préalablement dans sa base de données biométrique. Le système biométrique renvoie un résultat positif au cas où l'empreinte fournie à l'entrée correspond à l'un des modèles, et un résultat négatif dans le cas contraire.

A ce point, le but global est donc d'avoir un système qui fait la différence entre une image en entrée et plusieurs images situées dans une base de données. Pour cela, il faut utiliser une approche rapide et précise, c'est la raison pour laquelle on va éliminer l'approche par comparaison des images pixel par pixel parce qu'elle est assez lente.

La comparaison entre les empreintes est basée sur la recherche de la différence entre les minuties d'image d'entrée et les autres dans la base de données.

La méthode généralement utilisée pour détecter les minuties consiste à mettre l'image de l'empreinte en noir et blanc, c'est la binarisation de l'image, et à donner une même taille aux lignes de l'empreinte c'est la squelettisation. Une fois que l'on dispose de l'image binaire squellettisée, les minuties (singularités) sont mieux visibles, on procède alors à leur détection.

2.1- Prétraitement des images d'empreinte:

Basé sur la nature des bases de données proposées par [8] qui contient les différentes empreintes on observe que tous les images nécessitent un traitement.
Le premier objective est de chercher à regrouper les images est les transformer en se basant sur la même dimension afin de faciliter la comparaison

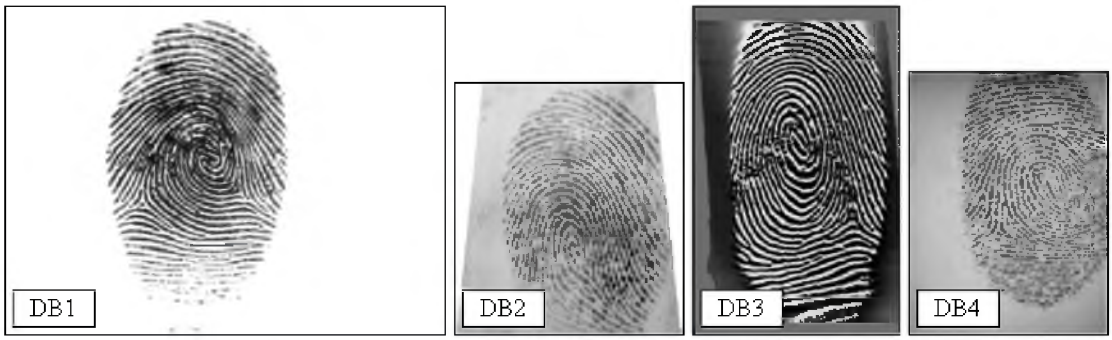


FIGURE 5:ECHANTILLANT DES BASES DE DONNEES PROPOSENT PAR [8]

La 2ème étape est de faire un prétraitement au niveau d'image, ce traitement a pour objectif d'améliorer la qualité de l'image contre le bruit lié à la mesure de perturbation.

La figure 4 nous donne une idée sur le processus suivi dans phase du prétraitement.

(a) Image d'origine. (b) Image binarisée. (c) Image squellettisée, minuties détectées (points en couleur).

FIGURE 6:TRAITEMENT D'UNE EMPREINTE DIGITALE [8]

A) La binarisation:

La binarisation consiste à transformer une image à plusieurs niveaux en une image en noir et blanc (deux niveaux seulement). C'est le moyen privilégié pour isoler des objets.

Par suite, une image binaire peut être représentée par une matrice booléenne dont chaque élément signifie Vrai (1 = blanc) ou Faux (0 = noir).[10]

La binarisation d'empreintes digitales est une technique pour produire une image de type 1-bit, avec 0 comme crêtes qui sont teintées de noir et de 1 les vallées qui sont teintées de blanc [9] (voir figure 7).

Pour arriver à une image binarisée correctement il faut bien choisie une méthode de binarisation qui nous donne la forme d'empreinte sans malformation, nous avons testé plusieurs algorithme de binarisation dans le chapitre 3 la partie 2.1.1 page 29, pour la préparer à la 2ème étape de prétraitement " la squelettisation"

a) Image originale	b) Image binarisée

FIGURE 7:EXEMPLE D'OPERATION DE BINARISATION [8]

B) La squelettisation:

Un algorithme d'amincissement (ou shrinkingalgorithm) consiste en la suppression jusqu'à stabilité de points simples, le résultat obtenu s'appelle un noyau homotypique. Si la suppression est réalisée de façon séquentielle alors la topologie est préservée ; cela par la définition même d'un point simple. Si le processus est modifié de façon à ce que certains points simples soient préservés durant le processus de suppression, il est alors possible de conserver des caractéristiques géométriques. Un tel processus s'appelle algorithme de squelettisation (ou thinningalgorithm), et le résultat est appelé squelette. Les points à préserver sont appelés points terminaux ou points extrémités.[11]

L'objectif est ici de diminuer l'information redondante contenue dans une image, donc la quantité de données à analyser. La méthode est l'isolement des lignes principales de l'image avec des amincissements successifs jusqu'à ce que l'image résultante ne contienne que des lignes d'épaisseur 1 pixel. La méthode nécessite l'emploi successif de 8 masques. On effectue sur l'image une succession de passes; on arrête lorsque le résultat entre deux passes successives est inchangé. Une passe consiste en l'application successive, sur toute l'image de chacun des 8 masques (le point central sur le point courant à traiter). Les 8 masques correspondent aux transformations suivantes : si la situation de gauche est rencontrée, alors on remplace le pixel traité par 0.

a) Image binaire	b) Image squelette

FIGURE 8:EXEMPLE D'OPERATION DE SQUELETTISATION [8]

Répéter
Pour tout point de l'image déterminé selon un balayage séquentiel **faire**
Sile point est simple **alors** il est supprimé
(**Sinon**examiner le point suivant, déterminé par le balayage)
Jusqu'à ce qu'il n'y ait plus de suppression durant un balayage complet de l'image.

FIGURE 9:SCHEMA SEQUENTIEL DE SQUELETTISATION [11].

Nous avons par ailleurs évalué les performances de deux procédés de squelettisation, « Zhang » [12] et «Shapiro» [13] dans le chapitre 3 la partie 2.1.2 page 34.

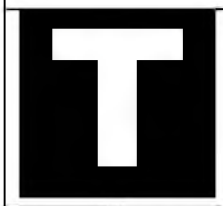
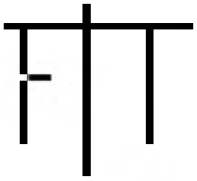
	«Shapiro»	« **Zhang** »
		
		
Temps d'exécution (image 256*360, P133 MHz)	1.6 s	1.3 s
Mémoire nécessaire (en taille image K octets)	2	1

FIGURE 10:COMPARAISON EXPERIMENTALE DES ALGORITHMES DE SQUELETTISATION DE « ZHANG » ET DE «SHAPIRO.» [9].

2.2- Extraction des minuties

Après avoir obtenu l'image traitée, on doit trouver dans cette dernière les minuties les plus intéressantes de l'image.

La signature retenue pour caractériser l'empreinte est basée sur un ensemble suffisant et fiable de minuties. On entend par suffisant, le nombre minimum de minuties nécessaires pour pouvoir établir des comparaisons fiables entre empreintes. Ce minimum se situe à 12 minuties vis-à-vis de la loi, voire moins pour beaucoup d'entre eux (jusqu'à 8 minimum). Le nombre 12 provient de la règle des 12 points selon laquelle il est statistiquement impossible de trouver 2 individus présentant les mêmes 12 points caractéristiques, même en considérant une population de plusieurs dizaines de millions de personnes.

On entend par fiable, les minuties qui ne sont pas influencées par des défauts lors de l'acquisition de l'image ou par l'altération temporaire de l'empreinte digitale (blessure, érosion, etc.). Avec un petit nombre de minuties (15 ou 20) correctement localisées, il est possible d'identifier une empreinte parmi plusieurs millions d'exemplaires.

Généralement, chaque minutie occupe un espace de 16 octets sans compactage ni compression. Ceci explique la taille de chaque flchier signature, 240 octets pour 15 minuties et 1600 octets pour 100 minuties.

Les bifurcations et les terminaisons sont les deux types de minuties les plus utilisés car ils sont facilement détectables, mais surtout parce qu'ils sont très aisément représentables par le modèle de coordonnées, où chaque minutie est représentée par les coordonnées (x, y) de son emplacement et l'angle θ.

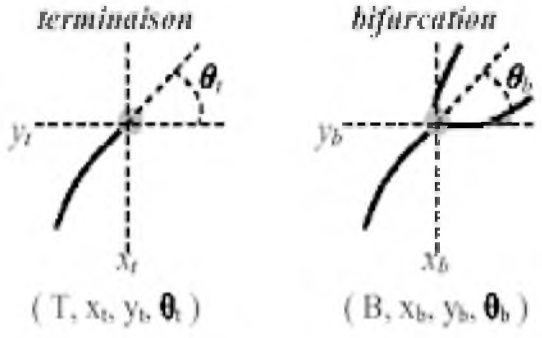


FIGURE 11:REPRESENTATION DES MINUTIES PAR LE MODELE DE COORDONNEES [9].

Lors du processus d'extraction, on détecte initialement 100 minuties en moyenne, parmi lesquelles environ 60 % correspondent à de fausses minuties qui seront identifiées lors d'un processus ultérieur. Généralement les logiciels extraits donc une quarantaine de minuties réelles de l'empreinte. Cette valeur est nettement supérieure aux minima, ce qui augmente la fiabilité. De plus, ce chiffre est loin du total de minuties détectées, ce qui laisse supposer que n'ayant conservé que les plus fiables, on a éliminé les minuties erronées qui auraient pu détériorer le comportement du système.

Les minuties sont généralement extraites à partir du squelette de l'image, il existe une approche permettant de détecter les minuties ainsi que leurs types (Terminaison, Bifurcation) en calculant l'indicateur Crossing-Number(CN) en fonction du voisinage de chaque pixel. Selon la valeur de CN le type du point est déterminé : Continuité ou Discontinuité (minutie) .[9]

Cette méthode ne retient que l'emplacement des minuties les plus pertinentes.

En analysant le squelette binaire de l'image de l'empreinte, on remarque que les pixels correspondant aux minuties possèdent un ***crossing-number***différent de 2. Le ***crossing-number***d'un pixel p se calcule par la formule suivante :

$$CN = 0.5 * \sum_{i=1}^{8} |P_i - P_{i+1}|$$

P9 = P1, Pi est la valeur des pixels dans le voisinage 3*3 de P.
p0, p1, …. P7 sont les 8 pixels au voisinage de p
En effet le coefficient CN présente des caractéristiques qui permettent d'identifier la nature d'une minutie en fonction du résultat obtenu lors du calcul de CN.

CN	NATURE DE LA MINUTIE EN P
0	*Erreur* =>Point isolé
1	Terminaison
2	*Erreur* =>Point ϵ Sillon
3	Divergence ou bifurcation
4	*Erreur*=>Minutie à 4 branches

TABLE 1:IDENTIFICATION D'UNE MINUTIE A PARTIR DU CALCUL DE CN

Ces méthodes restent tout de même très pratiques car elles sont faciles à mettre en place et donc moins coûteuses. Elles sont souvent combinées avec d'autres méthodes plus complexes pour assurer la fiabilité et la robustesse du système.

2.3- Comparaison des minuties

La phase de comparaison des *minuties* s'apparente à du « point pattern matching ». Le problème majeur des nombreux algorithmes proposés dans ce domaine, c'est la croissance exponentielle de leur complexité en fonction du nombre de points à traiter.

A partir de deux ensembles de minuties extraites, le système est capable de donner un indice de similitude ou de correspondance qui vaut :

- 0 % si les empreintes sont totalement différentes.
- 100 % si les empreintes viennent de la même image.

Dans le chapitre suivant nous allonsétudier les différents algorithmes proposés dans la littérature pour résoudre le problème de comparaison.

CHAPITRE 2:

ETAT DE L'ART

1- Introduction:

La reconnaissance d'empreintes digitales est une technique biométrique mature pour toute application d'identification ou de vérification d'individus. Cette technique d'authentification attire l'attention des chercheurs depuis quelques décennies, et reste encore et toujours un sujet de recherche attractif et très ouvert. Beaucoup de connaissances dans les domaines de la reconnaissance des formes, du traitement d'images, des statistiques ont été appliquées au domaine de la reconnaissance des empreintes.

Dans la littérature il existe beaucoup d'algorithmes proposés (plus de 120 algorithmes) pour la reconnaissance des empreintes digitales chacun de ces algorithmes à des caractéristiques "point faible et point fort" par rapport aux autres.

L'université de Bologne a fait une collaboration avec l'université de Michigan pour la réalisation de FVC2004 (Figner verificationComptition) [8] le but est de crée plusieurs bases de données, chaque base avec un capteur des empreintes différents pour le but de la réalisation des algorithmes spécialisés à la reconnaissance d'empreinte.

Le principe de la reconnaissance des empreintes digitales consiste à comparer une empreinte fournie au système, à une ou plusieurs autres empreintes dont le système dispose préalablement dans sa base de données biométrique. Le système biométrique renvoie un résultat positif au cas où l'empreinte fournie à l'entrée correspond à l'un des modèles, et un résultat négatif dans le cas contraire.

La comparaison entre les empreintes elle basé sur la recherche de la différence entre les minuties d'image d'entrer et les autres dans la base de données.

La méthode généralement utilisée pour détecter les minuties consiste à mettre l'image de l'empreinte en noir et blanc, c'est la binarisation de l'image, et à donner une même taille aux lignes de l'empreinte c'est la squelettisation. Une fois que l'on dispose de l'image binaire squellettisée, les minuties (singularités) sont mieux visibles, on procède alors à leur détection.

2- Les techniques de reconnaissance:

2.1- Efinger :

Cet algorithme a été programmé par [14], Le capteur utilisé du type «Secugen » enregistre la forme de l'empreinte digitale à partir des variations électriques produites par les monts et les vallées du doigt avec une qualité d'image en 256 niveaux de gris, et une résolution de 260*300 pixels. Nous décrivons les différentes étapes de l'algorithme d'authentification des empreintes dans ce qui suit.

2.1.1- Prétraitement:

EFINGER présente les mêmes phases de prétraitement des images que nous déjà expliqué dans le chapitre précédant, il effectue les traitements suivant:

Le prétraitement est effectué lors de l'ajout de l'empreinte dans la base de données:

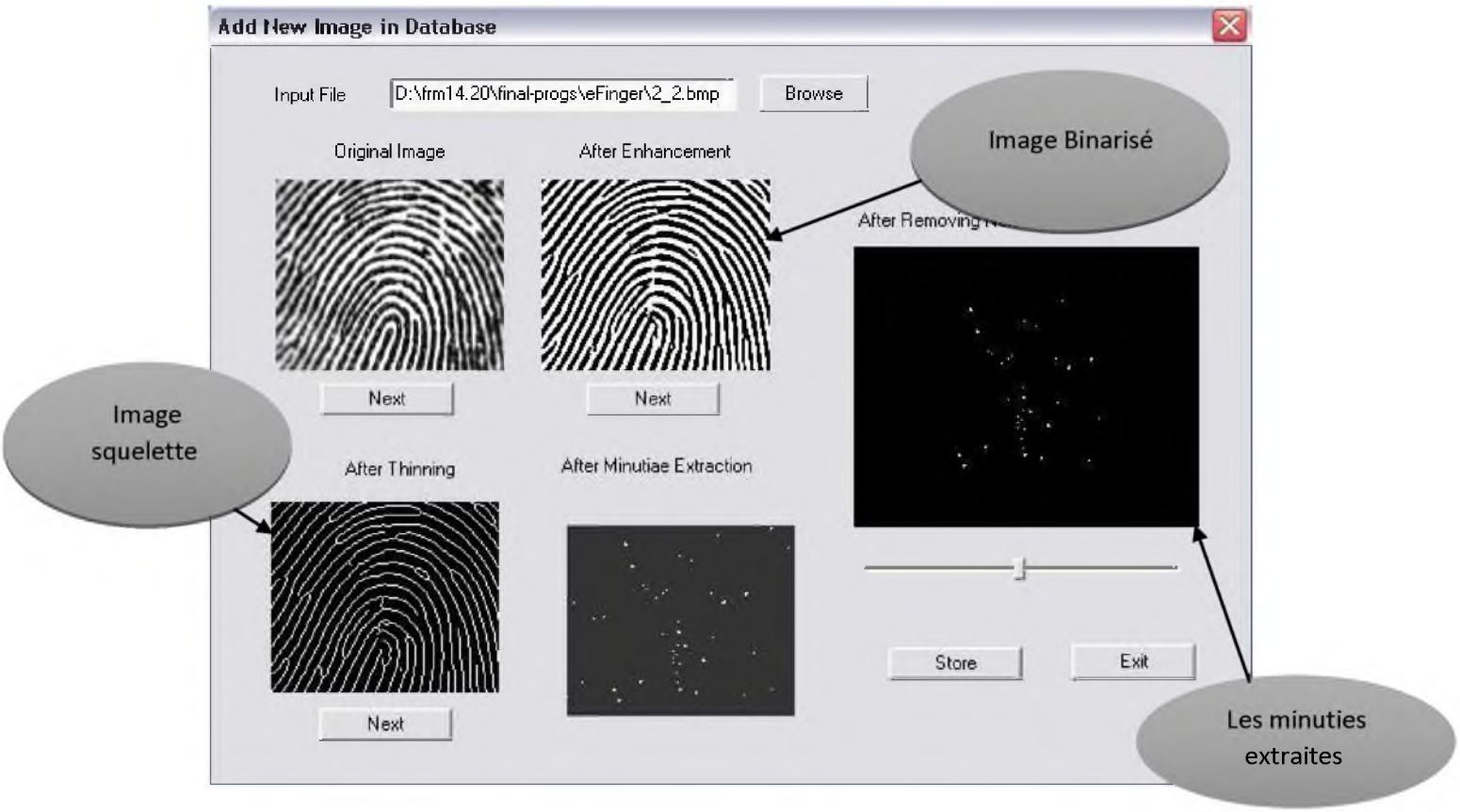


FIGURE 12: L'INTERFACE DE EFINGER DANS I A PARTIE PRETRAITEMENT ET STOCKAGE [14]

2.1.2- Extraction des minuties:

La dernière étape est la localisation des minuties par la création d'une image de points.

Après ils ont obtenaient l'image de points, ils ont proposé la fonction ***Makeminutiae***pour extraire les minuties, le principe de fonctionnement de cette méthode c'est de parcourir le fichier pixel par pixel, à la recherche des coordonnées de ces points. Ces coordonnées seront stockées dans un fichier texte pour chaque empreinte. Ces fichiers seront utilisés lors des tests de comparaisons d'images d'empreintes. Nous avons vu que en général, les algorithmes utilisait la connectivité pour déterminer les minuties.

2.1.3- Comparaison des minuties

Efinger dispose 3 méthodes permettant d'effectuer des comparaisons d'image d'empreintes digitales :

- MIN DISTANCE
- IMAGE MAPPING
- QUAD TREE

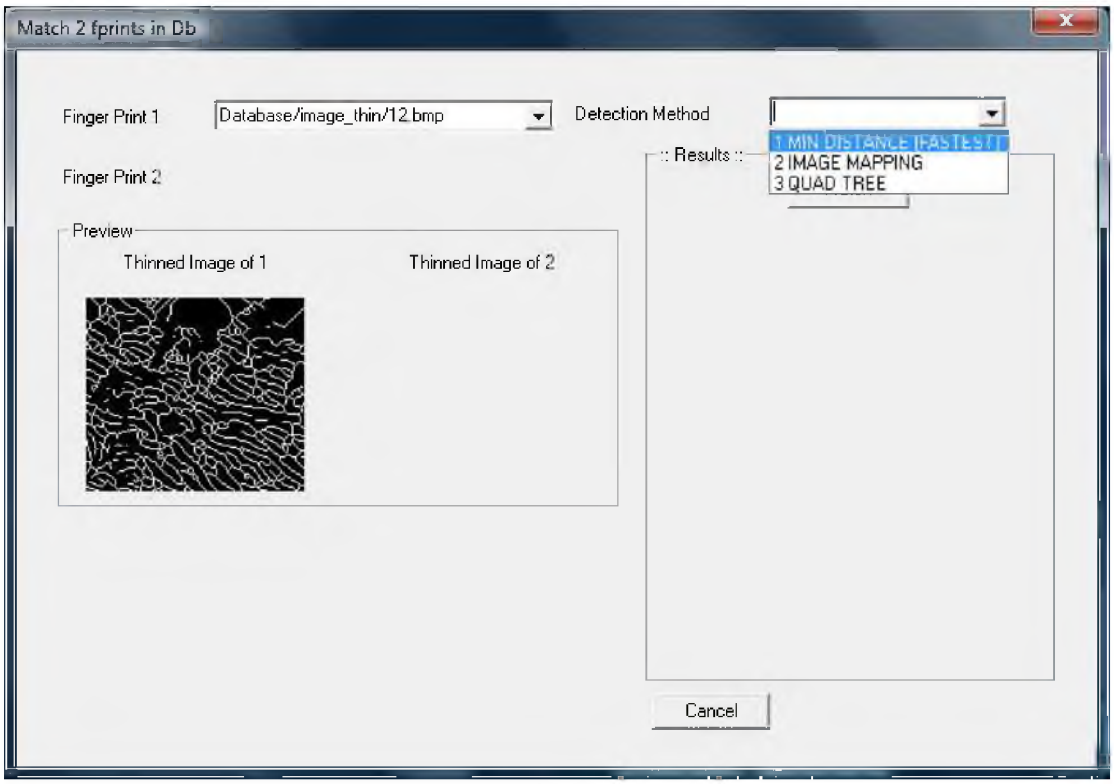


FIGURE 13:L'INTERFACE EFINGER PERMETTANT DE CHOISIR LA METHODE DE COMPARAISON D'EMPREINTE [14].

A l'aide de l'interface proposé par [14] Figure13, ils ont choisir une empreinte et comparent à toute la base de données d'empreintes utilisons les trois méthodes de comparaison proposé.

D'après [14], la méthode **MIN DISTANCE** permet de calculer un pourcentage en comparant les valeurs similaires de deux tableaux de minuties. En effet, comme dit précédemment chaque empreinte dispose d'un fichier.TXT contenant les coordonnées des minuties de ces empreintes.

D'après [14], la méthode **MIN DISTANCE** prend en argument deux tableaux mindata1 et mindata2 à deux dimensions contenant les coordonnées x et y des minuties. Ces tableaux sont construits à partir des fichiers .TXT.
La méthode Min Distance effectue une comparaison des distances entre les minuties de l'empreinte à comparer et celle des minuties des empreintes de la base. Un score est donné pour être comparé à un seuil préétabli.

L'algorithme de comparaison **IMAGE MAPPING**, effectue quant à lui une rotation de l'image squellettisée et la compare avec celles de la base de données.

La méthode **QUAD TREE** segmente l'image en 4 parties et compare ces parties à celles des empreintes de la base de données.

2.1.4- Phase de test:

D'après [14], le temps du traitement de chaque empreinte varie autour de 5 à 6 secondes.

Comme vu précédemment il faut savoir que cet algorithme est muni de trois méthodes de reconnaissances, voirTableau 2.

	Min distance	Image Mapping	Quad Tree
Temps de comparaison avec 10 empreintes	Entre 0,7s et 2,60s	Entre 1,45 min et 1,53 min	Entre 0,65s et 1,10s
Temps de comparaison avec 40empreintes	Varie entre 0,9s et 7,20 s Moyenne est autour de 3s	Varie entre 6,38 min et 6,53 Moyenne est autour de 6,44 min	Varie entre 6,38 min et 6,53 Moyenne est autour de 1,8s

TABLEAU2:COMPARAISON PAR RAPPORT AU TEMPS DES 3 ALGORITHMES DE COMPARAISON DES MINUTIES [14].

Remarque:

Comparaison avec une base de 10 empreintes :

D'après le tableau 2, la méthode « ***min distance*** » sort des résultats aléatoires pour chaque comparaison d'empreintes.
Pour ce qui concerne la méthode***Image Mapping***, il faut savoir qu'ils ont trouvent une tranche de temps équivalent pour les dix empreintes différentes.
Après avoir le tableau 2, la méthode***Quad Tree***,nous pouvons voir que celle-ci est la plus rapide au niveau comparaison des empreintes, par contre elle est moins performante en terme d'efficacité.

Comparaison avec une base de 40 empreintes :

Méthode « Quad Tree » est la plus rapide, par contre elle est moins performante en terme d'efficacité.
- L'augmentation de la base d'empreinte a une grande influence sur le temps de comparaison sur l' « Image Mapping ».

2.2- Apprentissage artificiel

Un ensemble d'ingénieurs [15] de l'université de Nahrain /Baghdad/Iraq Ont utilisé l'approche par réseaux de neurones pour la reconnaissance automatique des empreintes digitales.

Ce projet a été réalisé avec Matlab et les images utilisées sont de petite résolution 188*240 pixels pour évaluer la performance du système, la base de données a été divisée en deux parties, apprentissage et test.

Dans la phase de prétraitement il faut ajuster les images pour adapté l'entrée du réseau de neurone.

Les étapes du prétraitement sont pratiquement similaires aux autres algorithmes, la seule différence est dans la phase de comparaison.

Les minuties de l'empreinte digitale sont extraites à partir de son squelette en calculant la «Connectivité » CN qu'on a expliqué précédemment, le nombre des minuties extraites dans cet algorithme sont 12 .

La base de données utilisée dans ce projet contient 100 images, 50 pour l'apprentissage et 50 pour le test.

Dans la partie apprentissage l'un des objectifs c'est l'apprentissage du réseau En utilisant l'algorithme de back-propagation.

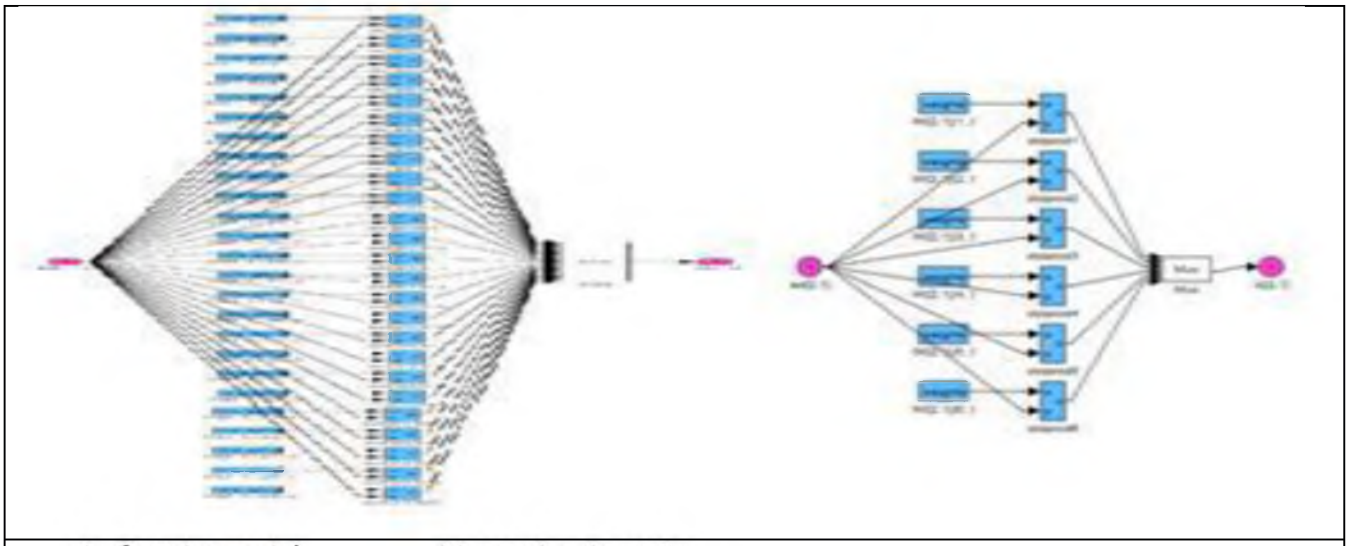
back-propagation est la méthode la plus utilisé pour entrainer le réseau de neurone

Cette méthode fonctionne comme suit:

Le modèle d'entrée sur laquelle le réseau doit être entraîné est présenté à la couche du réseau d'entrée et le réseau est exécuté normalement pour voir ce que la sortie produit effectivement. La sortie réelle est comparée à la sortie désirée pour ce modèle d'entrée. Les différences entre la forme réelle et désirée un modèle d'erreur [16].

Cet algorithme utilise 12 paramètres d'entrée et 6(les caractéristique ou les minuties) paramètres de sortie. Le chiffre 6 a été extrait à partir du nombre d'apprentissage $50 \approx (2^6)$.

La figure 14 nous donnons une idée sur le fonctionnement de cette approche.



a) Couche caché b) couche de sortie

FIGURE 14:IMPLEMENTATION DU RESEAUX DE NEURONE [15]

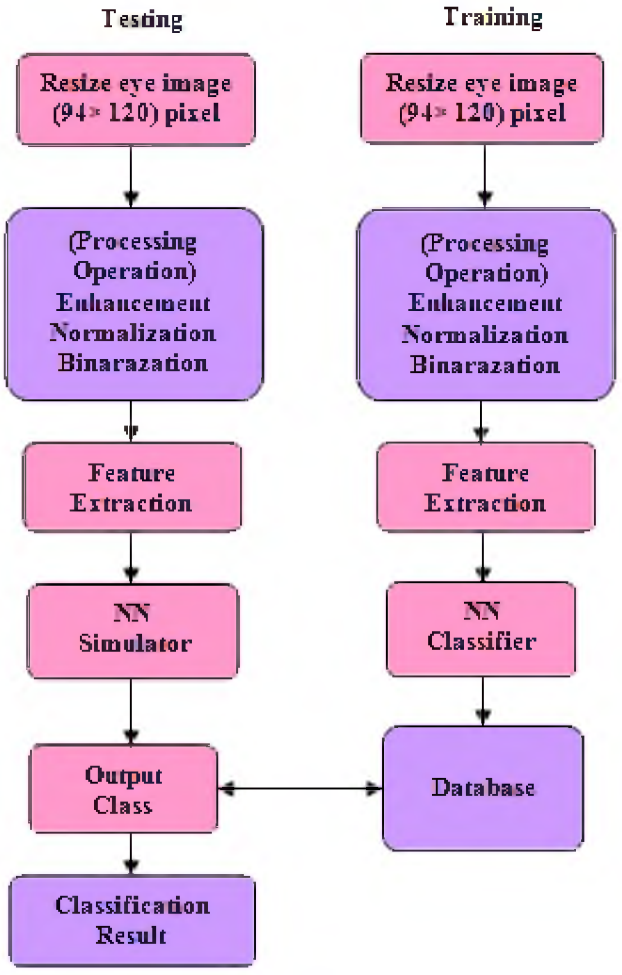


FIGURE 15:SYSTEM DE CLASSIFICATION DES EMPREINTES PROPOSENT PAR [15]

2.2.1- classification des empreintes:

La classification est l'étape final pour les systèmes qui utilisent l'extraction des caractéristiques à partir des images, dans l'objectif de les catégoriser.

Dans cette étude ils ont utilisent MLP (Multilayer Perceptron) pour la classification des modèles.

D'après [15], l'extraction des caractéristiques est l'une des tâches les plus importantes pour un système de reconnaissance. MLP est conçu pour détecter les caractéristiques de l'image d'empreinte digitale de la taille 188x240 pixels. La première couche du réseau comprend 12 neurones associés aux composantes du vecteur d'entrée. La couche cachée à 25 neurones et la couche de sortie a 6 neurones. La figure 15 montre la structure en trois couches MLP.

La mise en réseau sera entraînée en utilisant l'algorithme de rétro-propagation (back-propagation).

L'étape suivante consiste à saisir les images d'empreintes digitales prototype pour extraire les caractéristiques.

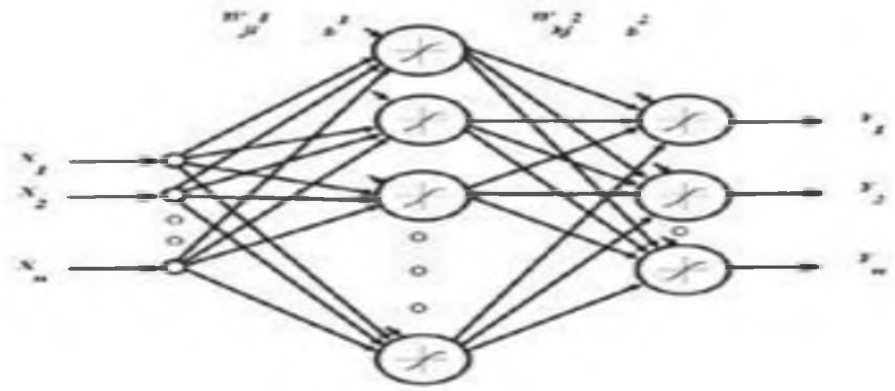


FIGURE 16:LA STRUCTURE EN TROIS COUCHES MLP [15].

2.2.2- Phase de test:

D'après [15], l'implémentation de cet algorithme est réalisée avec une machine de 2.1 GHz Pentium 4 Windows 7 et MATLAB 8.0, et la base de données utilisée dans ce projet contient 100 images avec une résolution 188×240 pixels, ils ont devisé cette base en deux partie, 50 pour l'apprentissage et 50 pour le test.
Le taux de reconnaissance estimé est à 100%, ce système a été considéré comme l'un de meilleure pour la reconnaissance des empreintes digitales, c'est pour cela, plusieurs algorithmes sont basés sur ce principe.

Type	*N°.* échantillon	*Taux de reconnaissance*
Apprentissage	50 images	
Test	50 images	100%
Le taux général de la reconnaissance		100%

TABLE 2:LE TAUX DE RECONNAISSANCE [15].

2.3 Conclusion:

Dans ce chapitre nous avons présenté les approches les plus connues de la reconnaissance des empreintes digitales. Comme déjà évoqué dans l'introduction, ce chapitre n'a pas pour finalité de décrire tous les algorithmes de reconnaissance des empreintes mais nous sommes contentés de présenter les algorithmes qui ont introduit une nette évolution dans le domaine de cette biométrie, permettant ainsi une véritable amélioration des performances.

CHAPITRE 3: CONTRIBUTION

1- Introduction:

Comme déjà présenté dans le premier chapitre, l'engouement pour les systèmes biométriques a connu un grand essor au début des années 2000. Les dépôts de brevet dans ce domaine se sont multipliés et des systèmes applicatifs à grande échelle ont été mis en place, comme le système utilisé lors des élections présidentielles au Venezuela en 2004 ou le système US Visit, devenu opérationnel depuis 2004. Ces systèmes restent basés sur les empreintes digitales, et ceci est dû aux évolutions techniques des algorithmes pour la biométrie basée sur les empreintes digitales [21].

2- Proposition d'un algorithme de reconnaissance d'empreinte:

La méthode la plus répandue consiste à extraire les minuties à partir d'un squelette de l'image. Comme la montre la Figure17 l'image est d'abord préparée à l'étape d'extraction au moyen d'une binarisation et d'une squelettisation, ensuite un fichier signature est extrait de l'empreinte après la détection et l'extraction des minuties.
Cet algorithme a été implémenté en java à l'aide d'Eclipse.
Sur la question pourquoi utiliser le langage Java plutôt que le langage C (ou C++), le Java possède des API bien plus documentée que la plupart des langages de programmation et la majorité des ordinateurs possèdent la JVM pour exécuter des programmes Java.

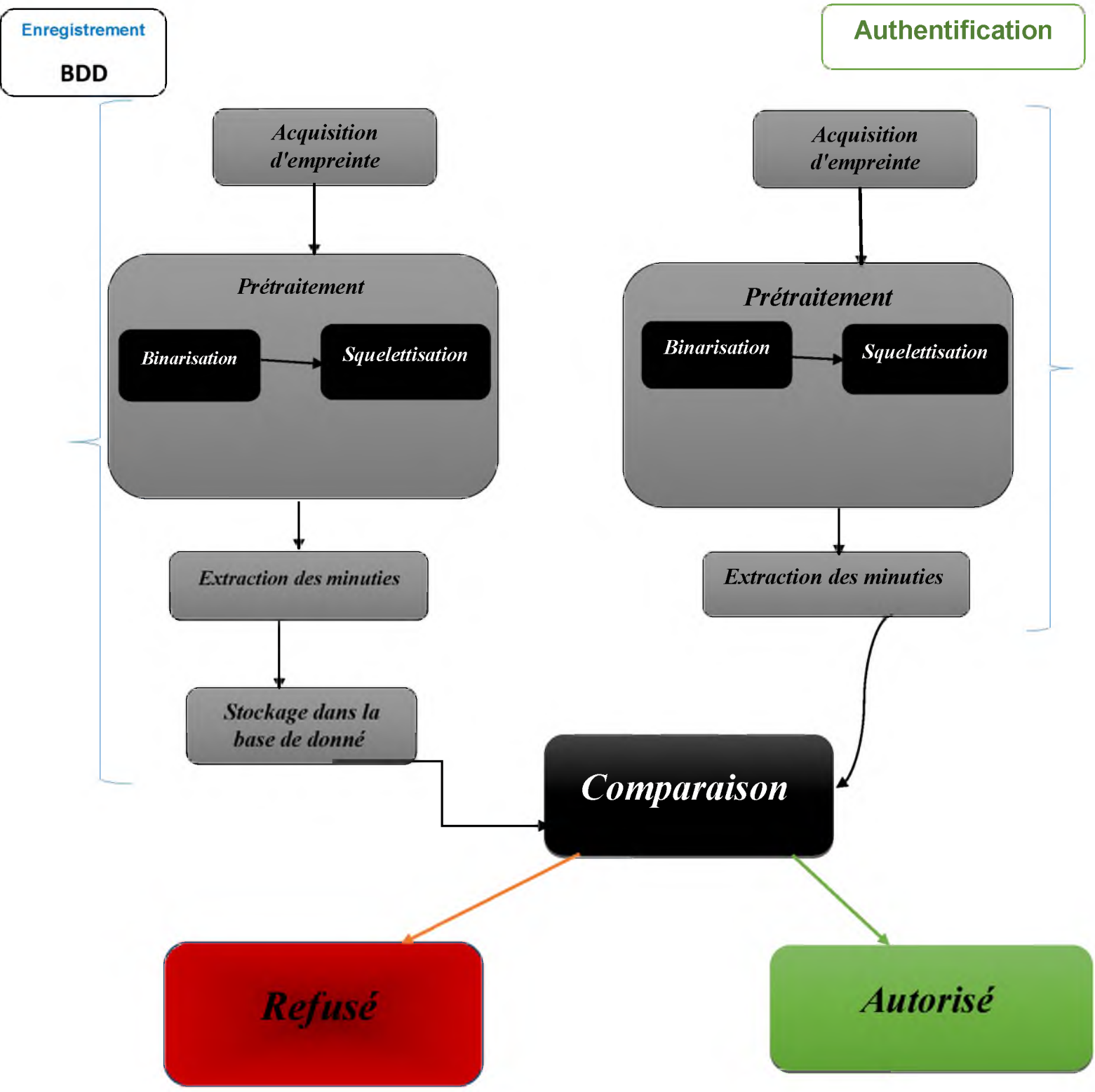


FIGURE 17:PROCESSUS SUIVIE DANS UN SYSTEM DE LA RECONNAISANCE DES EMPREINTES

2.1. Prétraitement des images d'empreintes digitales :

Basé sur la nature des bases de données proposées par [8] qui contient les différentes empreintes on observe que toutes les images nécessitent un traitement.

Le format TIF ou TIFF (Tagged Image File Format) est un format de fichier graphique bitmap de Windows utiliser dans les quatre bases de données proposer par [8], pour cela on a convertir le format des images aux .JPG pour un traitement par java.

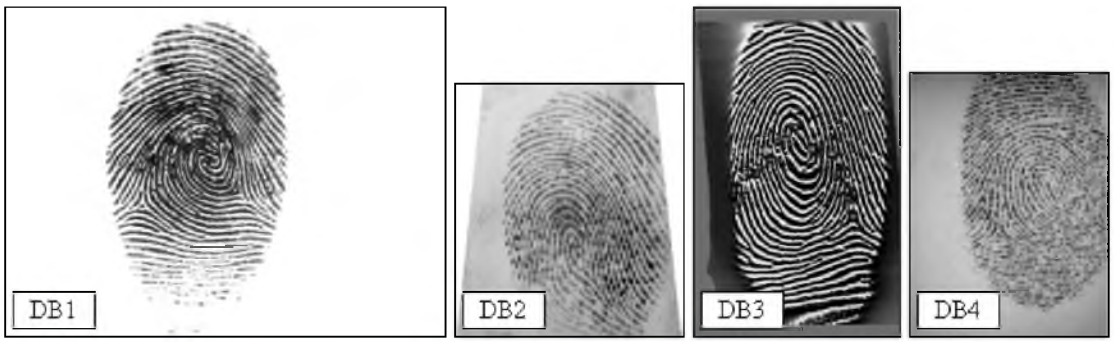


FIGURE 18:ECHANTILLANT DES QUTRES BASE DE DONNEES PROPOSE PAR[8]

2.1.1.Binarisation:

Pour permettre la squelettisation, l'image doit d'abord être binarisée, c'est-à-dire que l'image en 256 niveaux de gris dont nous disposons à ce stade est transformée en Image binaire où les pixels noirs correspondent aux stries et les pixels blancs aux vallées .

Il existe de nombreuses techniques de binarisation d'images.

a) Binarisation d'images par la méthode d'Otsu

Une des méthodes les plus importants pour déterminer le seuil global est la sélection de seuil Otsu. Dans un seuil global, nous choisissons une valeur de seuil unique pour l'ensemble des images.

La méthode d'Otsu recherche exhaustivement le seuil qui minimise la variance intra-classe. Le but de seuillage est de diviser les pixels d'une image donnée en deux classes Noir et Blanc (binarisation). Si l'on considère une image grise mise à l'échelle en tant que couches de niveaux de gris, alors nous avons besoin de déterminer un niveau de gris au-dessus duquel les pixels seront noirs, et le reste sera blanc.

D'après [18], Supposons que nous ayons L couches d'échelle de gris. t * est la couche sur laquelle toutes les couches sont considérées comme Noir. Et le reste est considéré comme blanc. Donc, si nous considérons Co pour le blanc et le noir C1 puis, Co = {0,1,2, ...} .t et C1 = {t + 1, t + 2, ... L-1}.

Après la phase d'implémentation avec java on remarque (Figure19) que dans les cas où la qualité d'une image d'empreinte est très faible la méthode de seuil global ne peut pas garantir des résultats interprétable est-il est nécessaire de trouvé un seuil spécial qui a un effet suffisant pour déduire une image résultant acceptable est utilisable dans la prochaine phase de la squelettisation..

a) Image original | b) Binarisation avec la méthode Otsu

FIGURE 19:EXEMPLE DE BINARISATION AVEC LA METHODE OTSU

Le seuil peut être fixé dès le départ (seuillage global). La méthode d'Otsu nous donne la possibilité de changer le seuil de façon manuelle

sellons la nature de l'image. Pour le choix du seuil il est à noter qu'on fait des tests (Figure 20) pour extraire la valeur du seuil qu'il nous donne des résultats acceptable

<table>
<tr>
<td> 

<pre>
static final int WINDOW_HEIGHT = 780;
static int THRESHOLD = 220;
static double pThreshold = 0;
BufferedImage bi=null;
BufferedImage si = null;
</pre>
Le 1ère test : avec un seuillage = 220.</td>
<td> 

<pre>
static final int WINDOW_WIDTH = 640
static final int WINDOW_HEIGHT = 78
static int THRESHOLD = 117;
static double pThreshold = 0;
BufferedImage bi=null;
BufferedImage si = null;
</pre>
Le 2ème test avec un seuillage =117</td>
<td> 

Le 3ème test est le même seuil 117 mais avec une autre image d'empreinte:</td>
</tr>
</table>

FIGURE 20:EXEMPLE DE L'IMPLEMENTATION DE LA METHODE D'OTSU SUR JAVA.

- La variété des images dans cette base de donné nous oblige de maitre on évidence l'importance de modifié de façon automatique le seuil ça dépend sur les caractéristiques de l'image.

A partir de cette problématique nous somme obliger de penser à une autre solution!

✓ **La solution:**

Nous allonschoisir des méthodes de binarisation local qu'il va appliquer le seuil dans chaque canal séparément.

Il existe autre algorithmes similaire spécialisé à résoudre le problème de la binarisation local comme:

- Bradley Local Threshold
- Bernsen Threshold.
- Maximum Entropy Threshold.

Ce sont des techniques de seuillage utilisé lorsque le fond est uniforme ou les différentes parties d'un document ont différentes origines.

A ce point nous allonschoisir une des trois méthodes pour l'utiliser dans notre système.

b) Bernsen local Threshold:

Principe de fonctionnement:

On utilise des méthodes de seuillage local.

Par «local», on entend ici que le seuil est calculé pour chaque pixel en fonction des caractéristiques de l'image à l'intérieur d'une fenêtre de rayon r autour d'elle.

Le procédé de seuillage local du Bernsen calcule le minimum local et le maximum pour un voisinage autour de chaque pixel.

Le procédé utilise un seuil de contraste fourni par l'utilisateur. Si le contraste local (max-min) est supérieur ou égale au seuil de contraste, le seuil est fixé à la valeur de gris moyenne locale (la moyenne du minimum et le maximum des valeurs de gris dans la fenêtre locale) .

Si le contraste local est inférieure au seuil de contraste du voisinage est considérée comme étant formée uniquement d'une classe et le pixel est situé à l'objet ou de fond en fonction de la valeur du gris moyen [22].

La phase segmentée est toujours affiché en blanc (255)

a) Image original

b) Binarisation avec la méthode Bersen

FIGURE 21:EXEMPLE DE BINARISATION AVEC LA METHODE DE BERSEN

2.1.2.Squelettisation:

Pour faciliter l'extraction des minuties, l'image doit être squelettisée.

Les deux méthodes de squelettisation Zhang et Shapiro qu'on a cité dans le chapitre précédant nous allons les expérimenter sur des images binaire, l'objectif étant d'extraire les minuties.

a) l'algorithme d'amincissement de Zhang-Suen

D'après [19], Supposant qu'on a une image 3*3 démontré comme suit :

P9	P2	P3
P8	P1	P4
P7	P6	P5

A(P1)= nombre de pixel 1 ou 0 qui dans l'entourage de P1, dans notre cas P2, P3, P4, P5, P6, P7, P8, P9, P8.

B(P1)= nombre des pixels noire ou 1 qui dans l'entourage de P1.

On ajoute la 1ère condition pour sélectionner les pixels noire pour supprimer

Condition 1: $2 \leq B(P1) \leq 6$

Condition 2: A(P1)=1.

Condition 3: P2. P4. P6 = 0

Condition 4: P4.P6.P8=0

Cette itération est répétée jusqu'à stabilité, i.e. jusqu'à ce qu'il n'y ait plus de point simple.

b) l'algorithme d'amincissement de Hilditch

D'après [20], Supposant quand on 'a une image 3*3 démonter comme suit :

P9	P2	P3
P8	P1	P4
P7	P6	P5

Nous voulons décider de décoller p1 ou garder comme partie du squelette résultant. A cet effet, nous organisons les 8 voisins de p1 dans un ordre d'horloge-sage et nous définissons les deux fonctions:

B (p1) = nombre de voisins non nuls de p1

Et

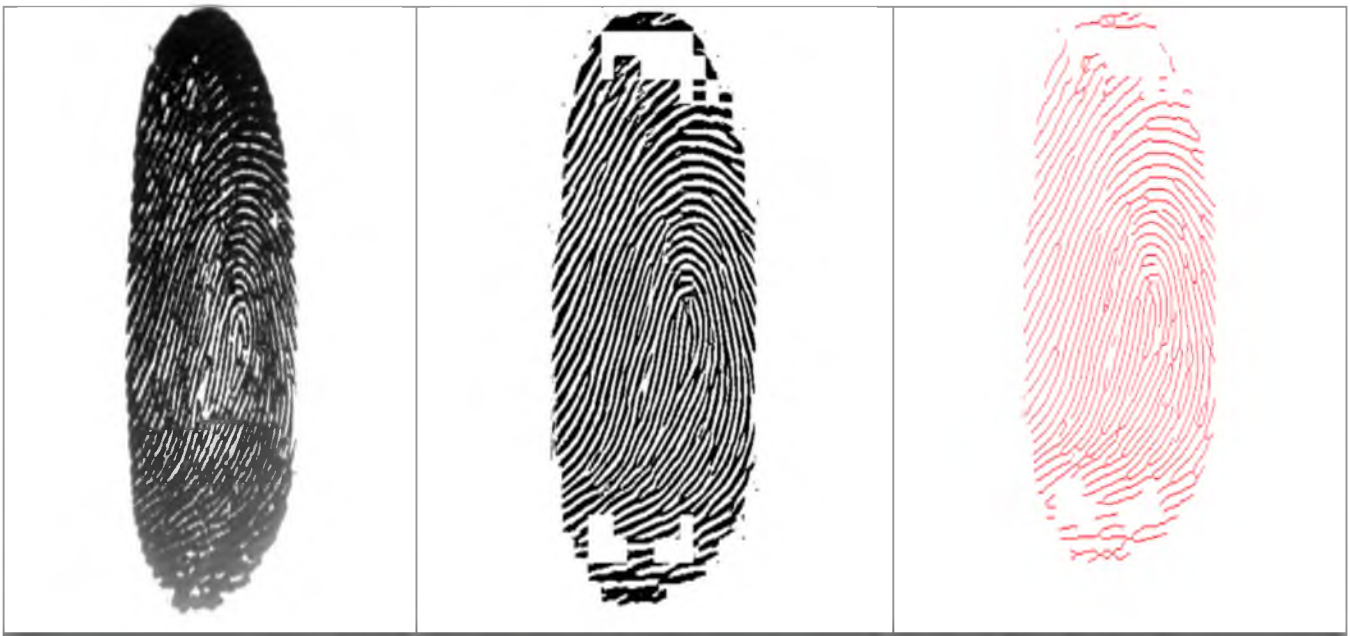
A (p1) = nombre de motifs 0,1 dans la séquence p2, p3, p4, p5, p6, p7, p8, p9, p2

Il existe deux versions pour l'algorithme de Hilditch, une en utilisant une fenêtre de 4x4 et l'autre à l'aide d'une fenêtre de 3x3.

L'algorithme de Hilditch consiste à effectuer plusieurs passes sur le modèle et à chaque passage, l'algorithme vérifie tous les pixels et décident de changer un pixel du noir au blanc si elle satisfait les quatre conditions suivantes [20]:

- 2 <= B (p1) <= 6
- A (p1) = 1
- p2.p4.p8 = 0 ou A (p2)! = 1
- p2.p4.p6 = 0 ou A (p4)! = 1

Stop lorsque rien ne change (pas plus de pixels peut être retiré)



A) image original	*B) Binarisationavec Bersen*	*C) Squelettisation avec Hilditch*

FIGURE 22:EXEMPLE DE PROCESSUS SUIVIE DANS LE PRE-TRAITEMENT

2.2. Extraction des minuties:

Les deux étapes de préparation à l'extraction (binarisation et squelettisation) ont grandement facilitécette phase. En effet nous disposons maintenant d'une image binaire squelettisée: un pixel noir prendla valeur **1**, un pixel blanc prend la valeur **0** et la largeur des stries est égale à **I** pixel. Si l'on calcule le nombre de transitions divisé par 2 entre un pixel blanc et un pixel noir pour chaque point du squelette, on obtient le nombre CN de stries partant de ce point (CrossingNumber) et nous pouvons donc déterminer simplement le type d'un pixel (voir Figure 24).

$CN=0.5*\sum_{i=1}^{8}|pi - pi_{-1}|$, avec $p_8=p_0$ et $p_i \in \{0,1\}$

P_1	P_2	P_3
P_8	P	P_4
P_7	P_6	P_5

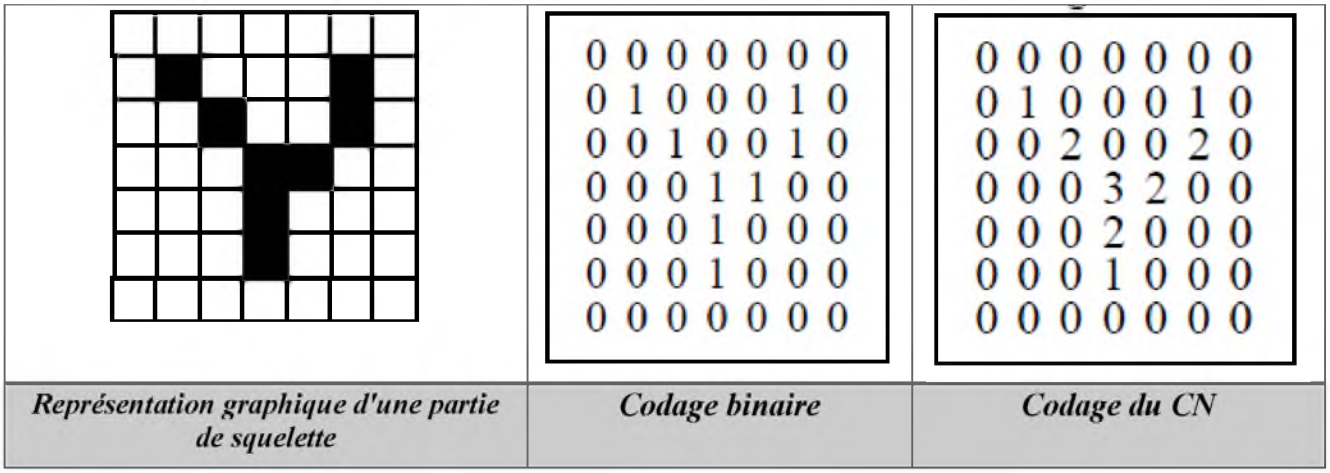


FIGURE 23:LES DIFFERENTES REPRESENTATION DU SQUELETTE [21].

CN(P) 1 : dans ce cas nous avons à faire à une minutie de type **terminaison**.

CN(P) 2 : c'est le cas le plus courant, le pixel se situe sur une strie, il n'y a pas de minutie.

CN(P) 3 : nous sommes en présence d'une **bifurcation** triple.

CN(P) 4 : nous sommes en présence d'une **bifurcation** quadruple. Ce type de minutie étant assez rare il est probablement dû à du bruit et nous l'ignorons.

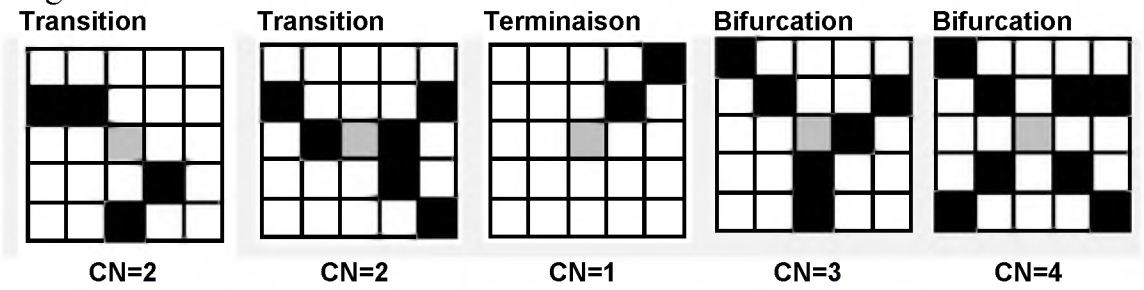


FIGURE 24: EXEMPLE DE DETERMINATION DU TYPE DE MINUTIES EN FONCTION DU CALCULE DE CN [21].

Bien que l'utilisation du nombre CN facilite grandement la détection elle provoque aussi la détection d'un nombre très important de minuties (quelques centaines) introduites pour la plupart lors des étapes de binarisation et de squelettisation (Figure 25). On ne peut donc extraire directement la signature.

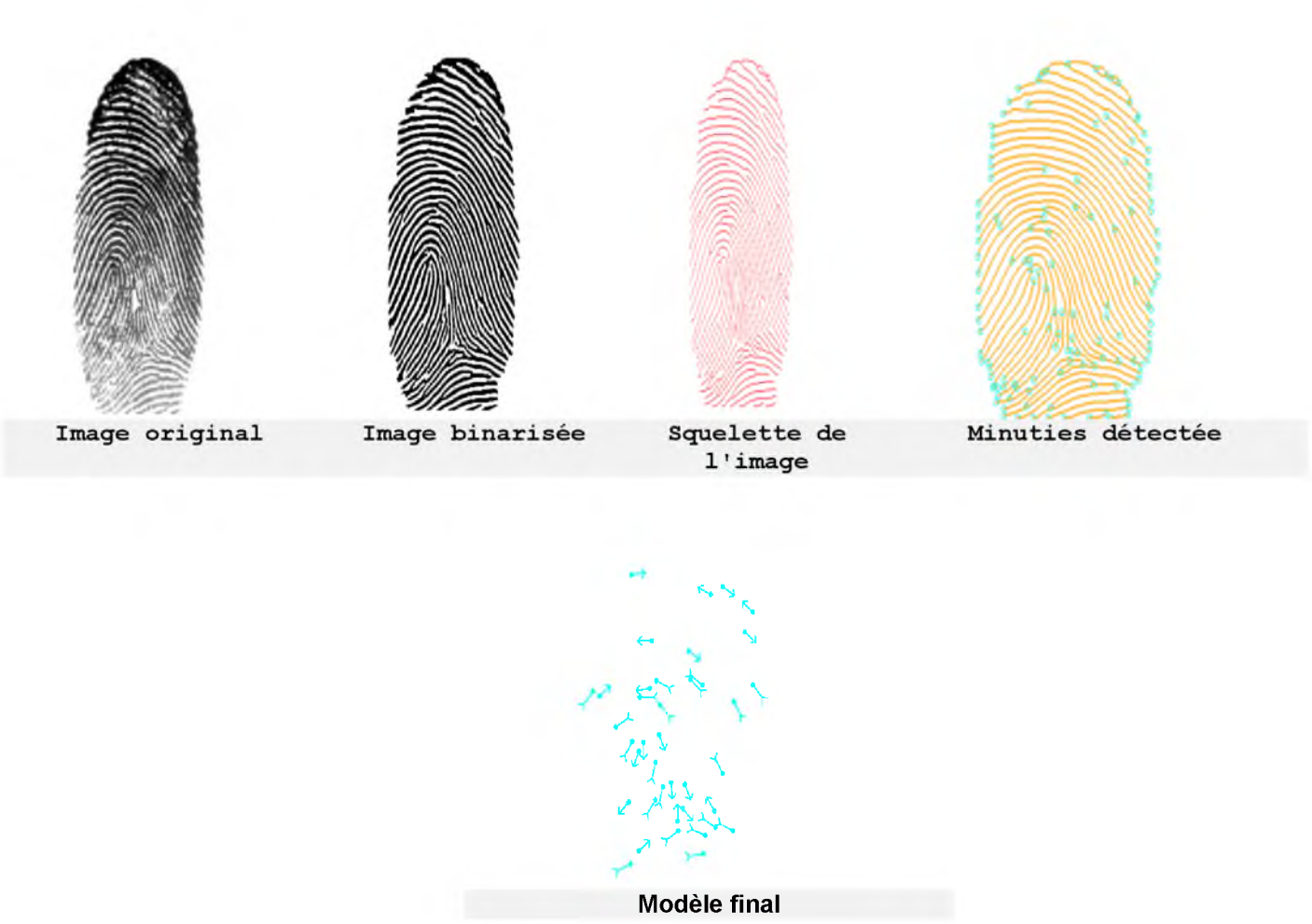


FIGURE 25:EXEMPLE DU PROCESSUS DE L'EXTRACTION DES MINUTIES.

2.2.1.L'implémentation de l'extraction des minuties avec java:

Dans la section précédente nous avons vu qu'un traitement supplémentaire est nécessaire pour éliminer la multitude de fausses minuties produites au cours des étapes de binarisation et de squelettisation (Figure 25).
Dans notre application on a limité le nombre des minuties aux 8 minutiesselon laquelle il est statistiquement impossible de trouver 2 individus présentant les mêmes 8 points caractéristiques, même en considérant une population de plusieurs dizaines de millions de personnes.

Comme le montre la Figure 26 montre l'implémentation avec java de la méthode de ***CrossingNumber*** pour extraire les minuties, Bifurcation représenté par CN=3, et Terminaison par CN=1.

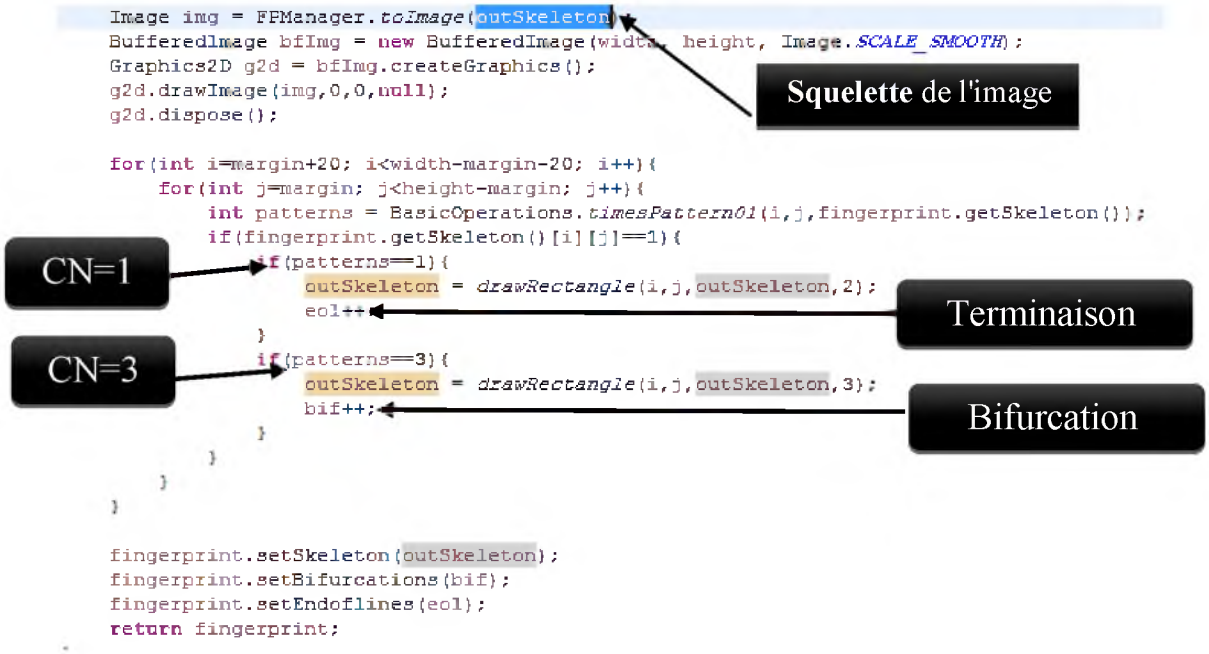


FIGURE 26: CODE JAVA DE LA METHODE DE CROSSING NUMBER.

2.3. *Comparaison des minuties:*

Dans la conception d'un système qui fait la reconnaissance des empreintes digitale les premiers objectifs que nous pensons à atteindre c'est la précision et la rapidité, donc si on va choisir une méthode basé sur la comparaison des images pixel par pixel il peut s'avérer assez lent, donc on a utilisé une méthode qui prend les coordonnée des minuties qu'on a stocké sous format texte dans une table de bases de données, donc la partie authentification ou comparaison l'empreinte va se réduire à une simple comparaison de minuties. Il faut souligner que nous avons créé une table qui contient

le **nom**, **prénom**, **id** et les **coordonnées** des minuties d'un utilisateur voir Figure 27. Nous avons aussi saisie une base de données qui contient 82 personnes avec le SGBD SQlite.

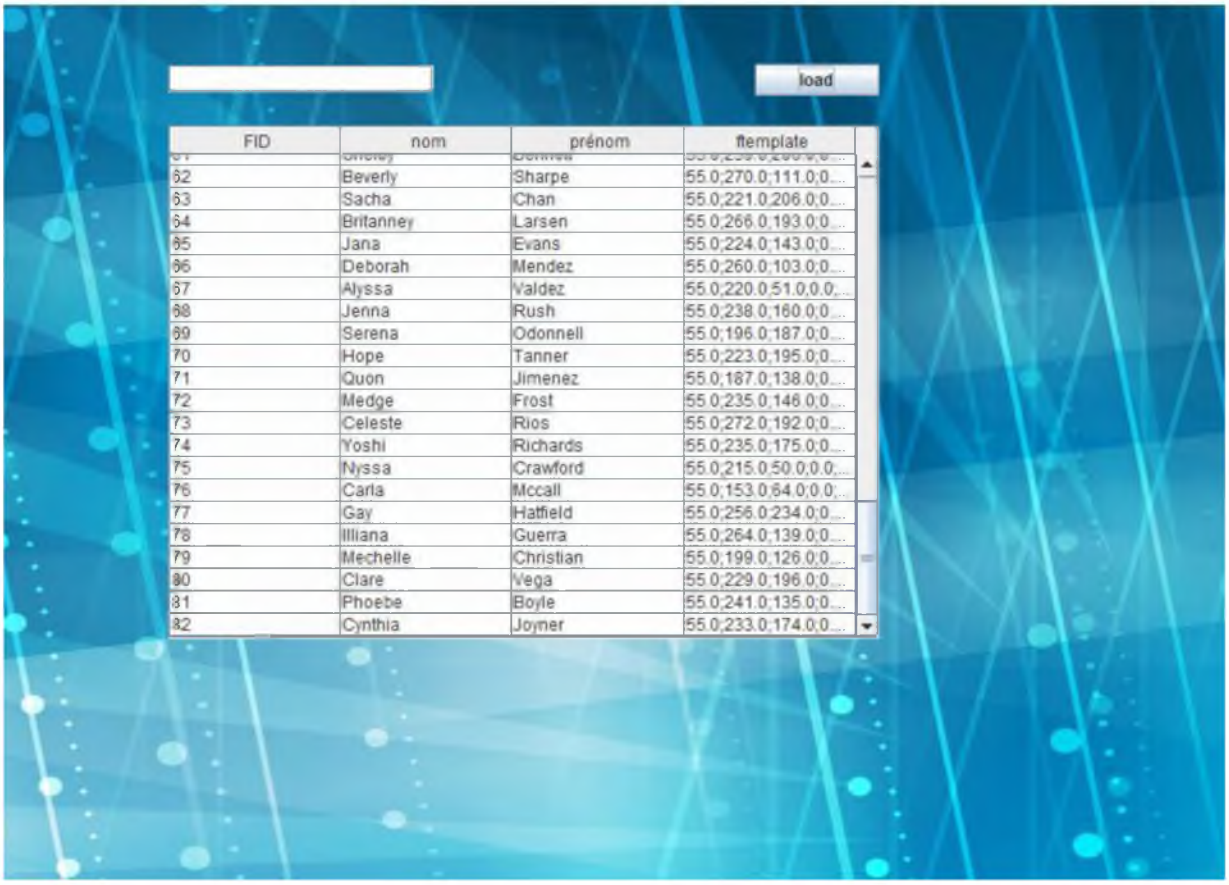


FIGURE 27:LA BASE DE DONNEES UTILISE DANS CETTE L'APPLICATION

Dans la partie d'authentification la recherche d'une empreinte parmi les empreintes de la base de données est basée sur le principe d'un moteur de recherche, les coordonnées des minuties sont stockées dans le champ ***ftemplate*** .

On à utiliser l'API r2xml pour faire la recherche

2.4. Description de l'application:

⇨ Notre application se compose de trois grandes parties principales et d'autres fonctionnalités supplémentaires (voir Figure 28):

FIGURE 28:L'INTERFACE PRINCIPALE DE L'APPLICATION.

1. ***Analyse de l'empreinte***: C'est une partie pour faire le traitement (binarisation→ squelettisation) et l'extraction des minuties de façon **manuelle**, nous pouvons examiner étroitement la nature des bifurcations et terminaison (voir Figure 29).

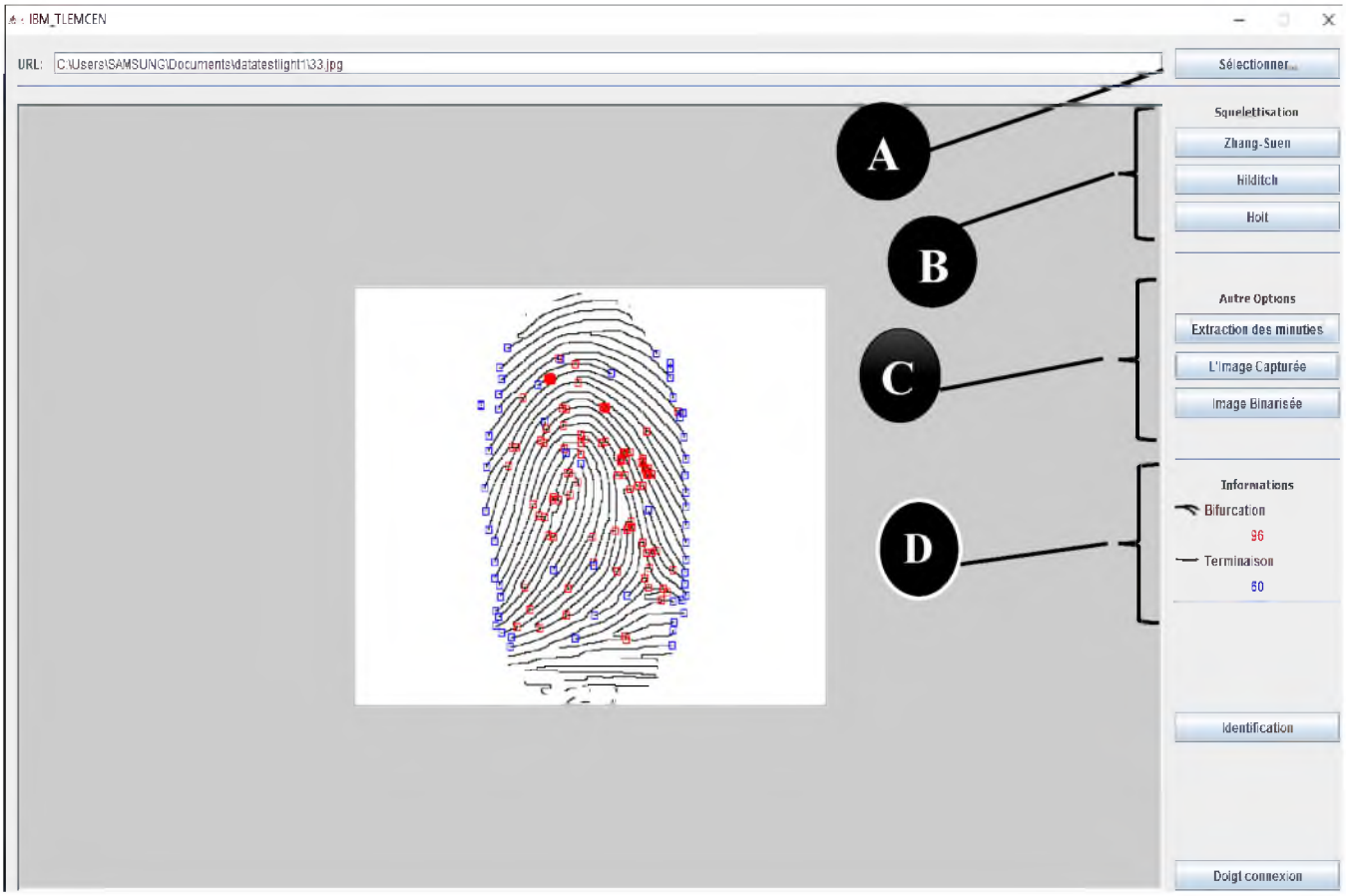


FIGURE 29:L'INTERFACE DU PARTIE D'ANALYSE DE L'EMPREINTE

A: bouton pour sélectionné une image d'empreinte.

B: cette partie est pour faire la squelettisation avec les différentes méthodes.

C: Autre Options: un mélange des commandes tel que l'extraction des minuties, binarisation de l'image et restauration de l'image au format initial.

D: Information sur le nombre des bifurcations et terminaison dans l'empreinte

2. ***Authentification***: Permet de faire le prétraitement et l'extraction des minuties, quand l'utilisateur met une nouvelle image d'empreinte, la fenêtre d'application affichera automatiquement l'image squelettisée avec les minuties choisie à gauche et à droite l'image original de l'empreinte et aux dessous les coordonnées des minuties dans une TextArea (voir Figure 30).

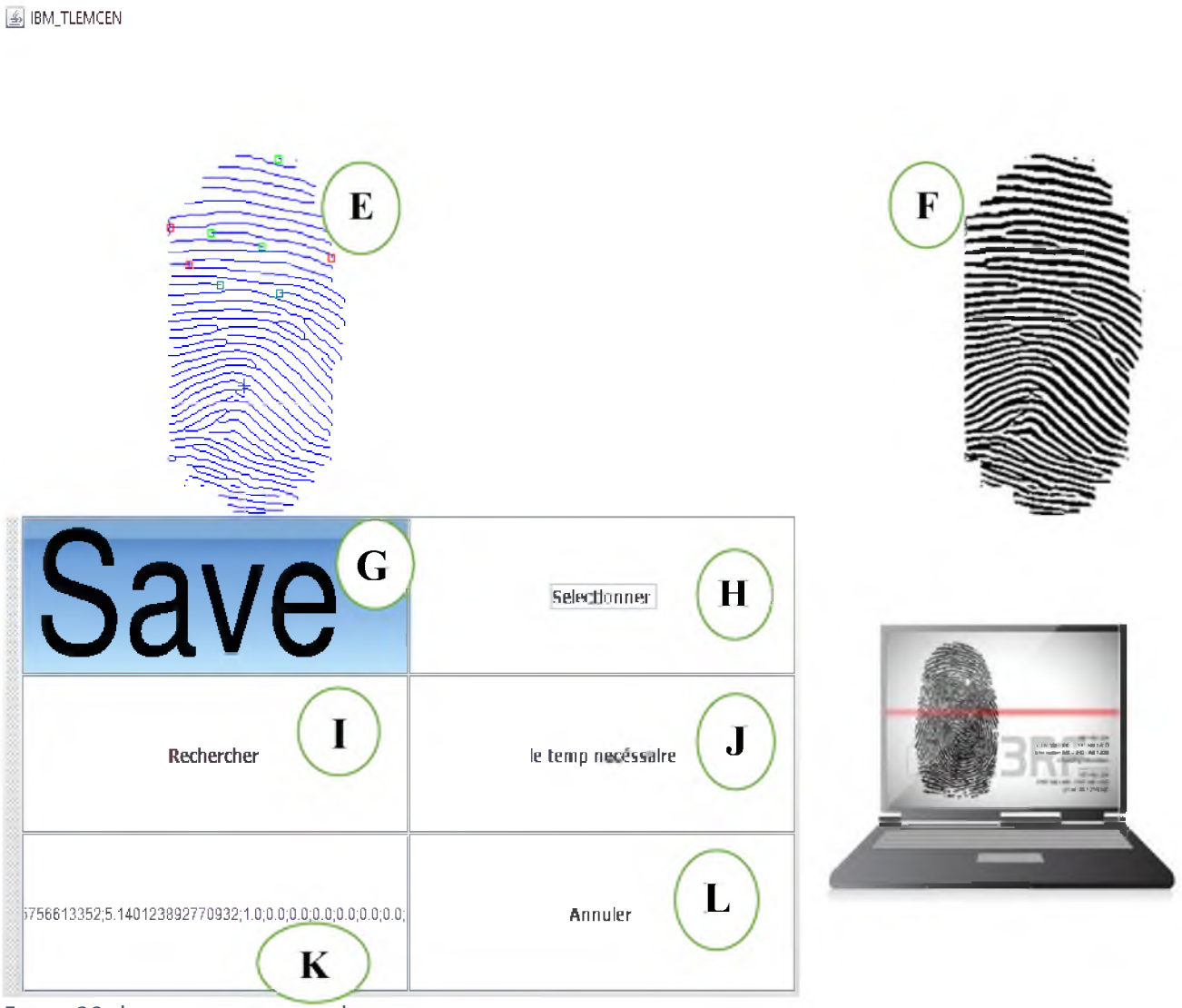


FIGURE 30:L'INTERFACE DU PARTIE D'AUTHENTIFICATION.

E: l'image de l'empreinte squelettisée avec les minuties choisie s'affichera automatiquement quand on met une nouvelle image.

F: image original de l'empreinte.

G: bouton Save pour enregistré l'empreinte d'une personne (voir Figure 31), La nouvelle fenêtre qui apparait nous donne la possibilité d'enregistrer une nouvelle empreinte (Id-Nom-Prénom) ou modifier les informations d'une empreinte existante ou supprimer une empreinte. Le chemin info sur empreinte contient les coordonnées qui représentent une empreinte.

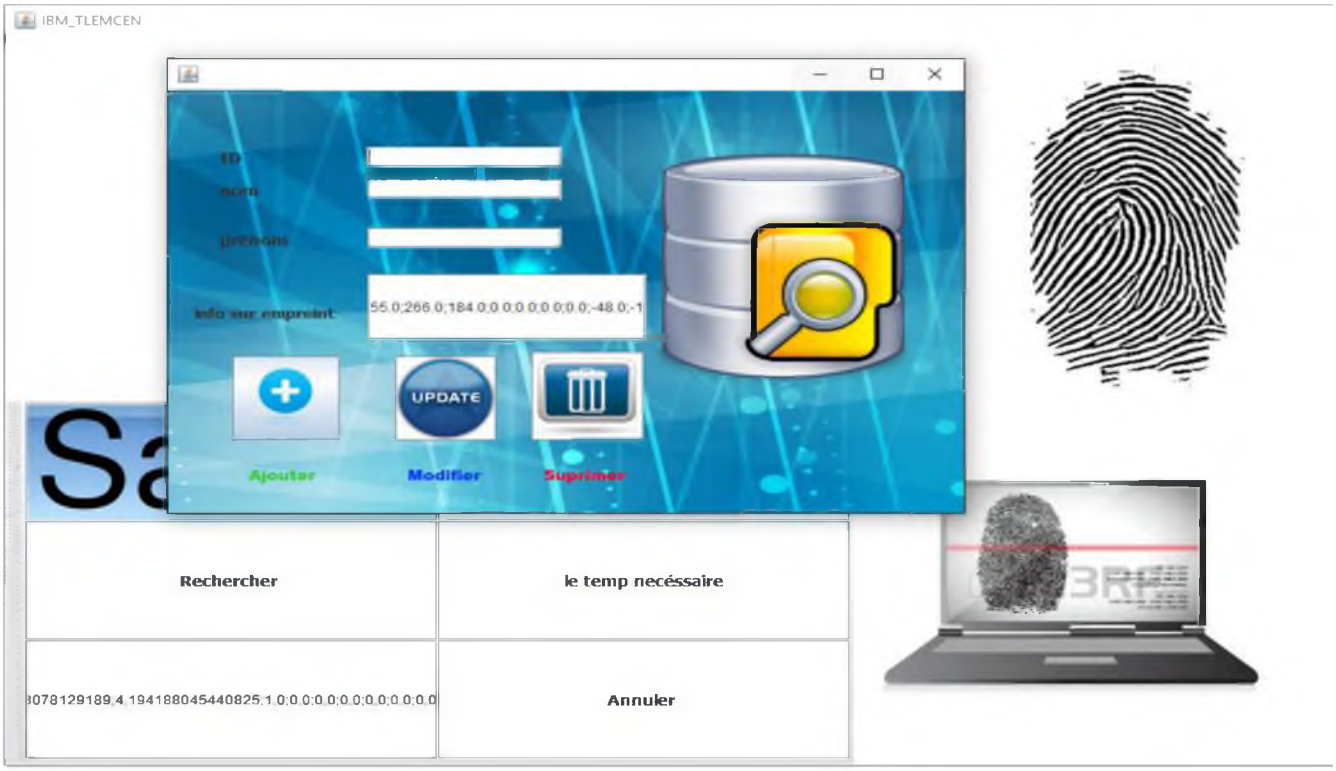


FIGURE31:L'INTERFACE RESPONSABLE AU L'ENREGISTREMENT.

H: pour sélectionné une nouvelle image d'empreinte.

I: cette commande pour rechercher si l'empreinte sélectionnée existe dans la base de donné (voir Figure 32).

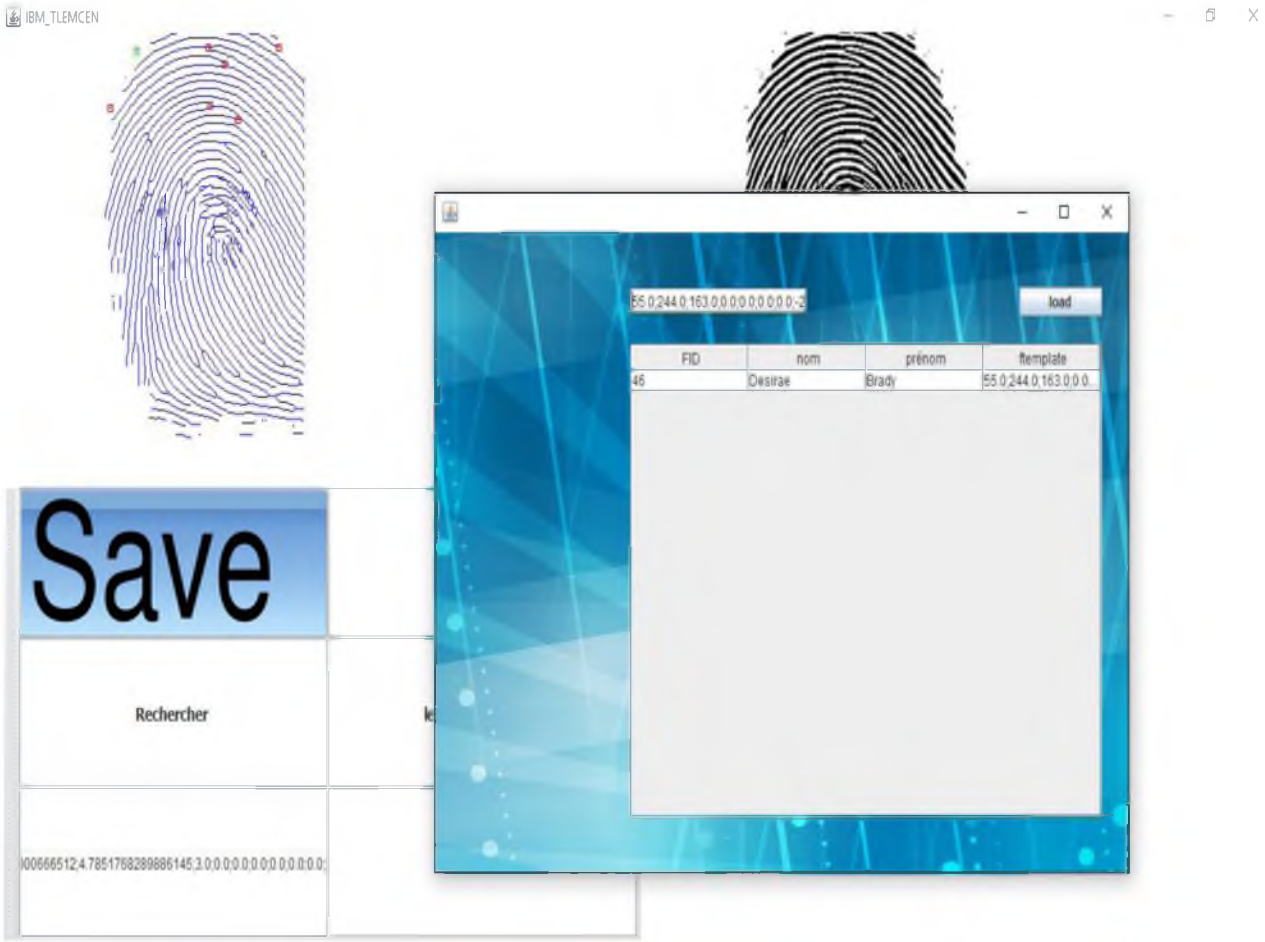


FIGURE 31:L'INTERFACE QUI DONNE LA RECHERCHE D'UNE EMPREINTE APARTIR DU BDD.

J: le temps nécessaire pour authentifier une personne.

K: Les coordonnées des minuties d'empreinte.

L: Annulé.

3. ***Comparer***: Cette partie permet de faire la comparaison entre deux empreintes digitales, dans l'authentification on a utilisé 8 minuties mais dans la comparaison entre deux empreintes on va mètre 60 minuties pour garantir un taux de reconnaissance significatif, et on a ajouté d'autres fonctionnalités telles que le temps de faire 500 comparaisons et le taux de comparaison (voir Figure 33).

   On utilisé une API standard "***Regex*** "Pour la comparaison entre deux empreinte qui nous donne la possibilité de rechercher le taux de similarité entre deux ensemble des chaines de caractère dans notre cas les coordonnées des minuties.

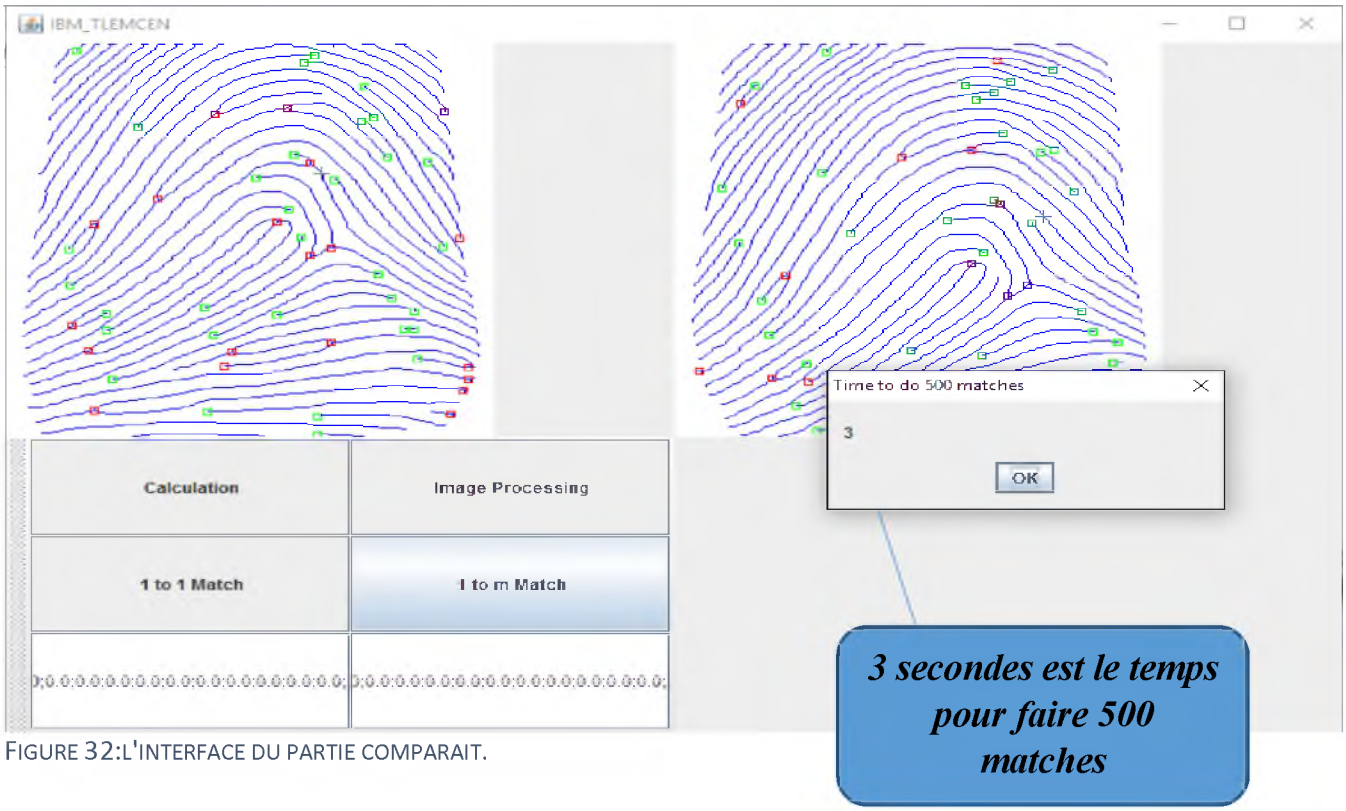


FIGURE 32:L'INTERFACE DU PARTIE COMPARAIT.

```
//match one print
try
{
  JOptionPane.showMessageDialog (null,Double.toString(m_finger1.Match(finger1 , finger2,65,false)),"Match %",JOptionPane.
}
```

FIGURE 33:L'IMPLEMENTATION SUR JAVA DU PATIE MATCH ENTRE DEUX EMPREINTE.

4. Cette partie présente un mini moteur de recherche qui permet de rechercher le nom d'un utilisateur dans notre base de données (Figure 35).

On a utilisé pour cela l'API ***r2xml***.

FIGURE 34:L'INTERFACE AFFICHER LORS QUAND FAIT UNE RECHERCHE.

5. Nous avons ajouté d'autres fonctionnalités supplémentaires (voir figure 36) afin de faciliter l'utilisation de cette application.

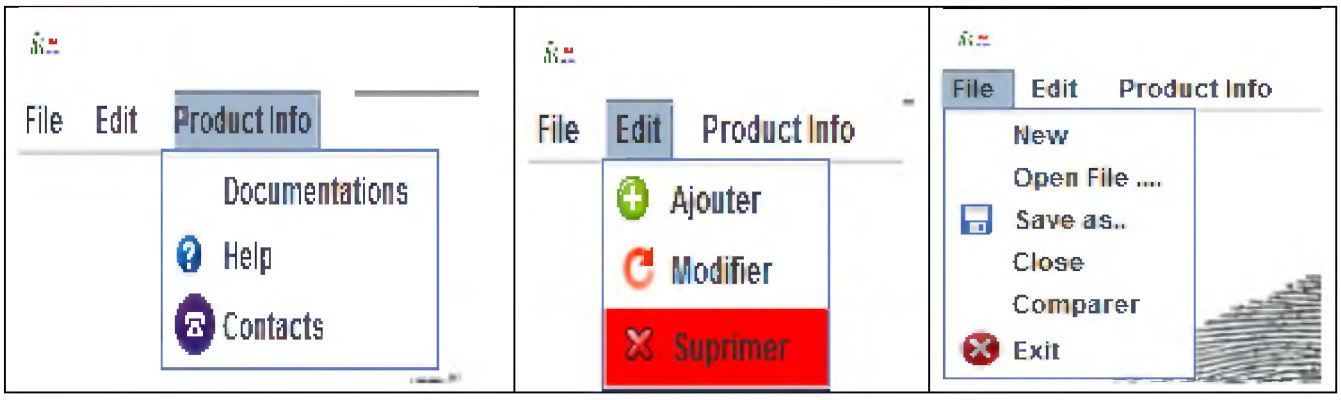


FIGURE 35: LES OPTIONS SUPPLIMENTAIRE

2.4.1. *Diagramme UML:*

Pour la création du diagramme UML on a utilisé un plugin sur Eclipse Mars 2015 s'appelle "***ObjectAid UML Diagramme***". Avec ce plugin nous allons créer automatiquement un diagramme UML de notre projet voir Figure 37.La classe **maintest** est la classe principale de notre application.

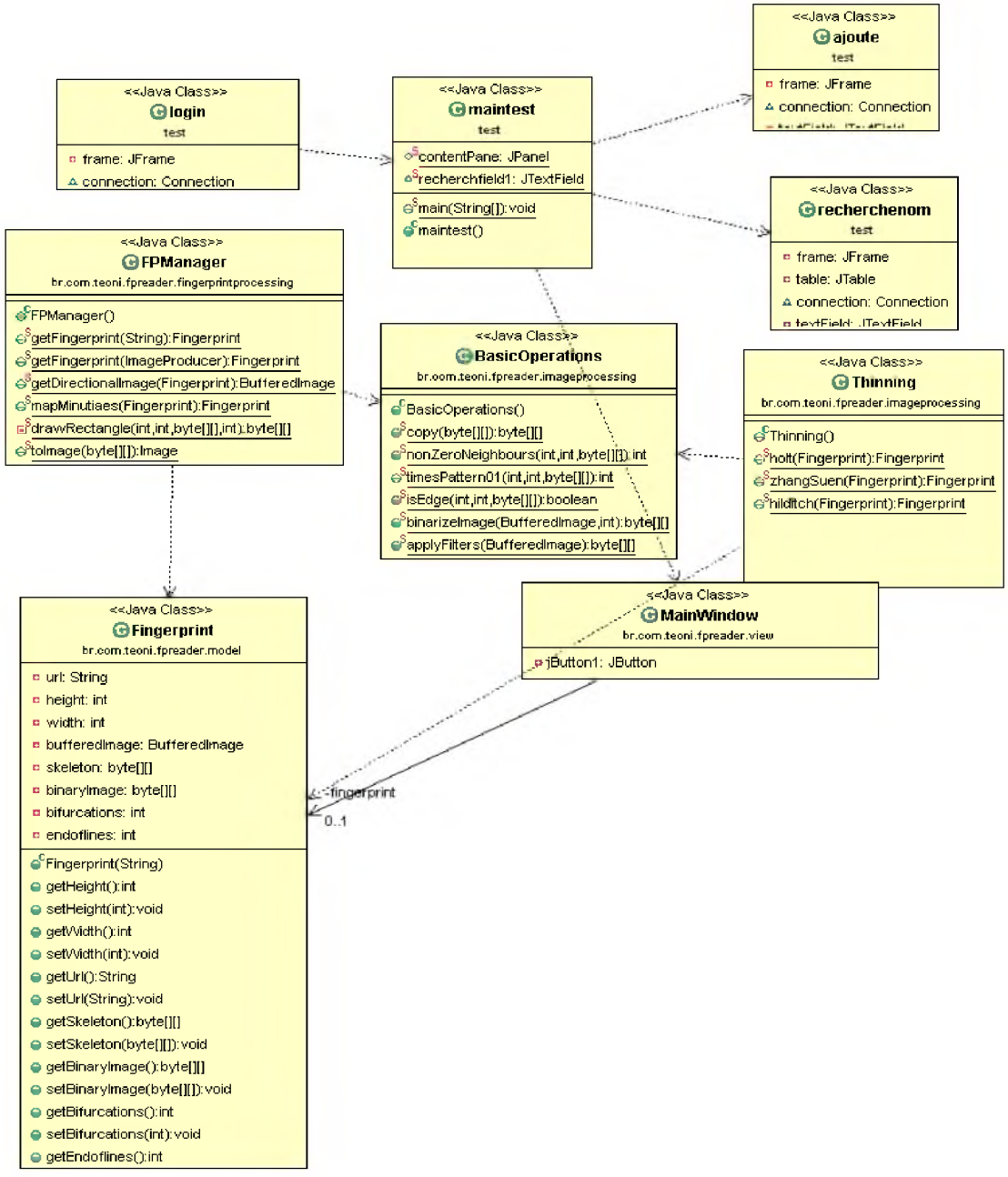


FIGURE 36:DIAGRAMME UML DE NOTRE APPLICATION DES CLASSE PRINCIPALE UTILISONS *OBJECTAID UML DIAGRAMME.*

2.5. *Partie test:*

L'expérimentation montre que le taux de reconnaissance obtenu est de l'ordre de 100%.
L'opération de comparaison prend maximum 3 second pour un nombre de minuties égale à 60.

FIGURE38:LES RESULTATS OBTIENT LORS DU COMPARAISON.

2.6. Conclusion:

Dans ce chapitre nous avons décrit la partie implémentation de notre application pour la reconnaissance des empreintes digitales. Il reste maintenant la validation et le déploiement de cette dernière.

CONCLUSION GENERALE

Au cours de ce travail nous avons étudié les différents algorithmes proposés dans la littérature et les déférents problèmes qui se posent durant la procédure de la reconnaissance des empreintes digitales.

Dans le deuxième chapitre, on a commencé par exposer l'état de l'art des méthodes de la reconnaissance des empreintes digitales. On a constaté que certain algorithme a ses propres avantages et inconvénients.

Le troisième chapitre est consacré à l'étude des différents algorithmes pour la reconnaissance. D'abord, on a appliqué une binarisation. Après, nous avonsappliqué la binarisation avec la méthode de **Bersen**, ensuite dans la squelettisation nous avonsappliqué deux méthode **Zhun-sen** et **Hilditch**. Lors de l'obtention du squelette le **CN** (crossing-number) est appliqué pour extraire les minuties, sachant que nous avons limité le nombre des minuties à 8. Dans la phase de comparaison nous avons proposé une méthode basée sur l'API **r2xml**.

Enfin nous n'avons traité que quelques points choisis. En particulier, nous n'avons pas abordé la reconnaissance des fausses empreintes, qui pourra faire l'objet d'un futur projet.

Références:

[1] MaltoniDavide, Dario Maio, Anil K. Jain, SalilPrabhakar, Handbook of fingerprint recognition, Springer, New York, 2003.

[2] Biosentis, www.biosentis.com, consulté le 25/04/16

[3] Francis Galton, Fingerprint, McMillan, London, 1892.

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

[5] LA POLICE SCIENTIFIQUE, Caractéristiques d'une empreinte digitales et différenciation, http://la-police-scientifique.e-monsite.com/ consulté le 26/02/2016.

[6] M. Patrick ISOARDI Serrure biométrique, Reconnaissance d'empreintes digitales, Fao Frédéric-Liméry Lionel-Guiraud Ludovic.

[7] MaltoniDavide, Dario Maio, Anil K. Jain, SalilPrabhakar, Handbook of fingerprint recognition, Springer, New York, 2003.

[8] FVC2004 (FingerprintVerificationCompetition) www.bias.csr.unibo.it/fvc2004 consulter le 25/01/2016

[9] Christel-Loïc TISSE, Lionel MARTIN, Lionel TORRES, Michel ROBERT, Système automatique de reconnaissance d'empreintes digitales. Sécurisation de l'authentification sur carte à puce,**A**dvanced **S**ystem **T**echnologyLaboratory.

[10] Notions de traitement et d'analyse d'image www.foad-mooc.auf.org/ consulté le 26/04/16.

[11] Christophe LOHOU, Contribution à l'analyse topologique des images : étude d'algorithmes de squelettisation pour images 2D et 3D, selon une approche topologie digitale ou topologie discrète, Informatique Fondamentale et Applications, 20 décembre 2001.

[12] J.R. Parker. *Algorithms for image processing and computer vision*. Wiley & Sons, Novembre 1996.
[13]Haralick, Robert et Shapiro. *Computer and robot vision*. Vol. 1, Addison-Wesley, 1992.

[14] LiméryLionel,FaoFrédéric,GuiraudLudovic,Reconnaissance d'empreintes digitales Sérrurebiométrique,IUP GMI D'AVIGNON, 2005.

[15] Hamsa A. Abdullah, Fingerprint Identification System Using Neural Networks, Nahrain University, College of Engineering Journal (NUCEJ) Vol.15 No.2, 2012 pp234 - 244

[16] A. Askarunisa, Sankaranarayanan. K, Sundaram. R and Sathick .M. Batcha, —**Finger Print Authentication Using Neural Networks**‖, MASAUM Journal of Computing, Vo. 1,No. 2, 2009.

[17] Futronicwww.futronic-tech.com/product_fs80.html consulté le 02-05-2016.

[18] N. Otsu, *A threshold selection method from grey scale histogram*, IEEE Trans. on Syst. Man and Cyber, vol 1, pp 62-66, 1979

[19] Mohamed Cheriet, NawwafKharma, Cheng-Lin Liu and Ching Suen, Character Recognition Systems: A Guide for Students and Practitioners, AJOHN WILEY&SONS,2007.

[20] Danielle Azar, Pattern Recognition course: Hilditch's Algorithm for SkeletonizationmProf. Godfried Toussaint.1997

[21] Nicolas Galy, Etude d'un système complet de reconnaissance d'empreintes digitales pour un capteur microsystème à balayage, INSTITUT NA TIONAL POLYTECHNIQUE DE GRENOBLE, 14 avril 2005.

[22] Nayef's Blog www.nayefreza.wordpress.com, consulté le 03/03/2016.

[23] Les expressions régulières avec l'API Regex de Java www.cyberzoide.developpez.com/tutoriels/java/regex/, consulté le 02/05/2016.