وزارة البحث العلمي والتعليم العالي

MINISTERE DE L'ENSEIGNEMENT SUPEREUR ET DE LA RECHERCHE SCIENTIFIQUE

جامعـة عبد الحميد بن باديس مستغانم

Université Abdelhamid Ibn Badis Mostaganem

كـلـيـة الـعـلـوم و الـتـكـنـولـوجـيـا

Faculté des Sciences et de la Technologie

DEPARTEMENT DE GENIE ELECTRIQUE

N° d'ordre : M...../GE/2019

MEMOIRE

Présenté pour obtenir le diplôme de

MASTER EN GENIE ELECTRIQUE

Option : Electronique des systèmes embarqués

Présenté Par :

Nom et Prénoms : KARABENTA Alpha Aboubacar Sidiki

MAÏGA Hamidou

Intitulé du sujet : Conception d'un logiciel d'authentification des individus par reconnaissance faciale

Soutenus le Mardi 09/07/2019 devant le jury composé de :

Président :	Mr BOUKORT Aek.	Pr	Université de Mostaganem
Examinatrice :	Mme MEHIDI A.	Mcb	Université de Mostaganem
Examinatrice :	Mme BERRADJA KH.	Mca	Université de Mostaganem
Rapporteur :	Mr SOLTANE BENALLOU A.	Maa	Université de Mostaganem
Rapporteur :	Mr BENAOUALI Med.	Maa	Université de Mostaganem

Année universitaire 2018 - 2019

Remerciements

Nous remercions le Seigneur Tout Puissant de nous avoir donné la force et les moyens de mener à bien notre projet de fin d'étude ainsi que de ses Bienfaits à notre égard au quotidien.

Sans les interventions conscientes et bienveillantes d'un grand nombre de personnes (nos familles respectives, amis, connaissances, collègues, professeurs, encadrants...), ce mémoire n'aurait jamais pu s'achever dans les conditions souhaitées. C'est aussi grâce à elles que nous sommes aujourd'hui capables de relever les défis de la société et de cette nouvelle ère de technologies. C'est donc avec le cœur rempli de gratitude et le sourire aux lèvres que nous remercions tous ceux qui ont participé de près ou de loin à notre éducation depuis notre âge de raison jusqu'à ce jour, ainsi que tous ceux ayant contribué à l'élaboration de ce travail.

Nos vifs remerciements à nos encadrants et membres du jury, qui nous espérons trouveront ici un hommage digne de leur précieuses contributions ainsi que de leur engagement à toutes épreuves pour le succès de ce projet de fin d'étude.

Dédicace

Nous dédions ce modeste travail :

- *A nos chers parents : grâce auxquels nous sommes aujourd'hui capable d'affronter la vie,*
- *A nos familles : qui ont été notre plus grand soutien durant toute notre vie,*
- *Aux nombreuses personnes qui nous ont soutenu et conseillé, envers qui nous n'exprimerons jamais assez notre gratitude,*
- *A notre pays le grand Mali, qui nous a tant donné*
- *A ce pays, l'Algérie qui nous bien accueilli et envers qui nous seront éternellement reconnaissant.*

Sommaire

Introduction générale

Chapitre I : La biométrie et les techniques de reconnaissance faciale

Chapitre II : Expérimentation et discussion de résultats

Chapitre III : Implémentation du logiciel

Conclusion générale

Bibliographie

Annexes

Table des figures

Chapitre I : La biométrie et les techniques de reconnaissance faciale

Chapitre II : Expérimentation et discussion de résultats

Chapitre III : Implémentation du logiciel

Liste des tableaux

Liste des abréviations

ACP : Analyse en composante principale

ADN : Acide Désoxyribonucléique

FAR : False Acceptance Rate

FARES_MAK : Face Recognition System of Maïga And Karabenta

FRR : False Rejection Rate

ICA : Independant Component Analysis

K-PCA : Kernel Principal Component Analysis

LBP : Local Binary Patterns

LDA : Linear Discriminent Analysis

PCA : Principal Component Analysis

SVM : Support Vector Machine

I. Introduction Générale

De l'ère de la préhistoire à nos jours, l'Homme a fait des progrès remarquables. De l'utilisation de la pierre et de la hache, Il s'est hissé à un stade encore plus évolué, une science jusqu'alors inexplorée, une science capable de le sauver mais aussi avec de lourdes conséquences. Le concept abordé dans ce rapport ne date pas d'aujourd'hui, il existait 3000 ans avant Jésus Christ et peut être plus.

Lors des échanges commerciaux à Babylone (3000 avant Jésus Christ) et dans la Chine antique (7ème siècle), la validation des transactions se faisait par une signature (l'empreinte digitale), pour le pourquoi de la question, le mystère reste encore entier (Internaute, s.d.). [1]

La biométrie, consistant à identifier une personne à partir de ses caractéristiques physiologiques et ou comportementales, part du principe vrai que les données biométriques d'une personne sont individuelles, contrairement aux mots de passe, badge… pouvant être volés, oubliés, ou utilisés par d'autres personnes, une donnée biométrique reste la meilleure alternative de clé inviolable de nos jours, voici un peu son histoire selon une source :

L'explorateur européen Joao de Barros a enregistré le premier exemple connu d'empreintes digitales, qui est une forme de biométrie, en Chine au 14ème siècle. Les commerçants chinois utilisaient l'encre pour prendre les empreintes digitales des enfants à des fins d'identification.

En 1823 un médecin et anatomiste tchèque Johan Evangelista Purkinje révéla au grand jour le fait qu'une empreinte digitale pouvait identifier de manière quasi-absolue un individu.

En 1890, Alphonse Bertillon, un poste de police parisien, étudie la mécanique corporelle et les mesures pour aider à identifier les criminels. La police a utilisé sa méthode, la méthode Bertillonnage, jusqu'à ce qu'elle identifie faussement certains sujets. La méthode Bertillonnage fut rapidement abandonnée au profit des empreintes digitales, remises en usage par Richard Edward Henry de Scotland Yard.

Karl Pearson, un mathématicien appliqué a étudié la recherche biométrique au début du 20e siècle à l'University College de Londres. Il a fait d'importantes découvertes dans le domaine de la biométrie en étudiant l'histoire statistique et la corrélation, qu'il a appliquées à

l'évolution des animaux. Son travail historique comprenait la méthode des moments, le système de courbes de Pearson, la corrélation et le test du chi-carré.

Dans les années 1960 et 1970, des procédures d'authentification biométrique de signatures ont été élaborées, mais le domaine biométrique est demeuré fixe jusqu'à ce que les agences militaires et de sécurité étudient et développent la technologie biométrique au-delà des empreintes digitales.

En 2001, Super Bowl à Tampa, en Floride – chaque image faciale des 100 000 fans qui passaient à travers le stade a été enregistrée via des caméras de vidéosurveillance et vérifiée électroniquement contre des photos de la police de Tampa. Aucun criminel n'a été identifié et la vidéosurveillance a conduit de nombreux défenseurs des libertés civiles à dénoncer les technologies d'identification biométrique.

Après le 11 septembre (les attentats aux Etats Unis), les autorités ont installé des technologies biométriques dans les aéroports pour identifier les terroristes présumés, mais certains aéroports, comme Palm Beach International, n'ont jamais atteint le statut d'installation complète en raison des coûts du système de surveillance.

Le 7 juillet 2005 Londres, Angleterre – Les forces de l'ordre britanniques utilisent des technologies de reconnaissance faciale biométrique et des caméras vidéo à 360 degrés pour identifier les terroristes après quatre attentats à la bombe dans un métro et dans un bus à impériale. En fait, Londres a plus de 200 000 caméras de sécurité et caméras de surveillance qui sont utilisées depuis les années 1960 (actualitéhitech.com, s.d.). *[2]*

Parmi les données biométriques, nous nous intéresseront spécialement aux caractéristiques physiologiques du visage, étant le trait biométrique le plus utilisé par les humains. Pour un être humain, il est aisé de reconnaître un visage, mais garder en mémoire le visage d'une multitude de personnes ainsi que leur nom respectif est une tâche très difficile et nous pesons nos mots, c'est pourquoi nous feront appel à une machine pour ce travail (relevant de l'Intelligence Artificielle.

Ce travail portant sur l'authentification des individus par reconnaissance faciale, c'est-à-dire identifier une personne grâce à son visage puis reconnaitre cette même personne déjà identifiée, se heurtera au même défi rencontré par les anciens systèmes de sécurité (satisfaire la sécurité de l'intégrité physique et morale d'une personne), car ce système a beau être

performant, il a des failles comme tout système électronique et informatique, mais cela dépend toutefois de beaucoup de paramètres.

Les systèmes biométriques sont utilisés en général :

- Par le gouvernement **:** la carte d'identité nationale, le permis de conduire, le contrôle des passeports, etc.
- Pour une application légale **:** la recherche scientifique criminelle, l'identification de terroriste, l'identification de corps, etc.
- La reconnaissance faciale est aujourd'hui une technologie utilisée dans plusieurs casinos aux Etats Unis et en France (identification des joueurs interdits ou individus fichés), aéroports, stades (refoulement de supporteurs connus et dangereux), centres commerciaux ou grands magasins.
- Une application commerciale : parmi lesquels on peut citer : l'e-commerce, l'accès Internet, la carte de crédit, l'ouverture de réseau informatique, la sécurité de données électroniques, le contrôle d'accès physique, la gestion des registres médicaux, etc.

Mais pour des usages à grande échelle, la reconnaissance faciale est le plus souvent associée à une autre technique biométrique. Le colonisateur israélien utilise un système combiné (forme de la main / reconnaissance faciale) pour filtrer les palestiniens en provenance de la bande de Gaza, et en Ouganda, les élections se déroulent à partir d'un fichier de "visages" d'électeurs.

La grande question est : une machine peut-elle authentifier efficacement un Homme grâce aux données biométriques de son visage ? Nous vous invitons à porter votre jugement après avoir parcouru ce rapport claire, précis et détaillé.

Ce projet de fin d'étude a pour objectifs :

- ✓ La conception d'un logiciel pouvant identifier et authentifier une personne par reconnaissance faciale.
- ✓ La perspective d'utiliser ce résultat pour interagir dans le monde réel. Exemple : Autoriser ou refuser un accès.

II. La biométrie et les techniques de reconnaissance faciale

II.1. Introduction de chapitre

Les toutes premières formes d'utilisation de la biométrie remontent à bien plus longtemps que ce que la plupart des personnes ne le croient. Les historiens ont des traces d'échanges commerciaux babyloniens utilisant les empreintes digitales pour la transaction de biens, depuis 3000 avant Jésus-Christ, en office de signature, ou encore vers le 7ème siècle en Chine antique pour les mêmes raisons.

Ce chapitre nous en apprendra davantage sur la biométrie, sa définition étymologique, son principe de fonctionnement, les techniques qu'elle utilise, sa structure générale, la façon de déterminer sa performance et son utilisation dans la vie courante. Nous verrons aussi en détail le réel but de ce rapport, après un état de l'art de la reconnaissance faciale suivra le choix de la méthode que nous avons utilisé ainsi que la justification de ce choix.

II.2. Concepts clés de la biométrie :

Pour une meilleure compréhension de ce rapport, nous définirons ci-dessous les concepts clés liés à la biométrie tels que la méthode de détection de viola et Jones, une classification, un apprentissage machine, la covariance etc. qu'il est nécessaire de connaître pour ne pas se perdre par la suite.

II.2.1. La biométrie :

La biométrie, originaire d'une contraction de deux mots grecs : « bios » et « metrie », signifiant respectivement la vie et la mesure, est la science permettant d'établir l'identité d'une personne basée sur ses attributs physiques et ou comportementaux (autrement nommés données biométriques). Il n'est pas exagéré d'affirmer qu'elle a vu jour dans le but de combler les manques des systèmes de sécurité classiques. Il existe toutefois plusieurs définitions de la biométrie, pour n'en citer que deux :

- « La reconnaissance automatique d'une personne à partir de son comportement ou d'une caractéristique physique ».
- « La biométrie recouvre l'ensemble des procédés tendant à identifier un individu à partir de la mesure de l'une ou de plusieurs de ses caractéristiques physiques, physiologiques ou comportementales ».

II.2.2. La biométrie physiologique :

Empreintes digitales, visage, géométrie de la main, ADN, odeur corporelle, voix, iris, rétine…. Sont les principales données biométriques physiologiques (voir figure 1), c'est une mesure des attributs physiques et unique d'un individu.

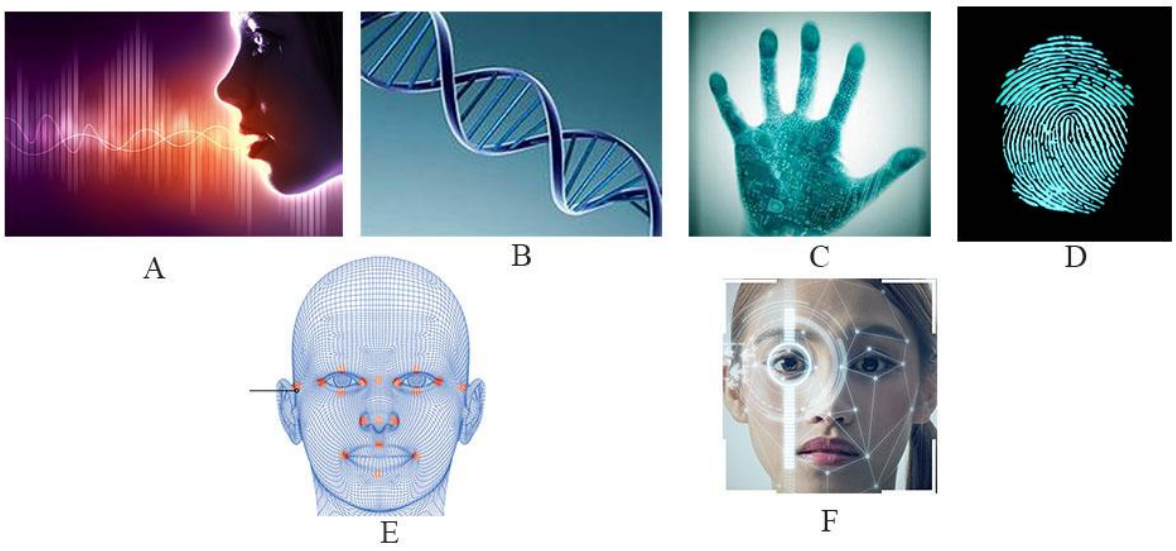

Figure 1 : Quelques biométries physiologiques : A- Voix, B- ADN, C- Forme de la main, D- Empreinte digitale, E- Visage, F- Iris

II.2.3. La biométrie comportementale :

La démarche, signature, dynamique de frappe au clavier, sont les principales données biométriques comportementales, elle se définit alors comme étant une mesure de l'attitude d'un individu.

II.2.4. La multi modalité :

Elle permet de combiner plusieurs biométries afin de diminuer les restrictions du système de reconnaissance monomodal. Un système multimodal a une architecture composée de deux ou plusieurs autres systèmes monomodaux qui peut prendre diverses configurations :

- Plusieurs programmes traitant une même donnée biométrique acquise : **multi-algorithme**
- L'acquisition de la même donné est faite par plusieurs capteurs : multi-capteur (exemple : empreinte digitale optique et thermique)

- Une acquisition indépendante de plusieurs données biométriques de la même personne : multi-biométrie (exemple : visage et empreinte digitale) …

Si l'acquisition et le traitement sont faits successivement, on parle d'architecture en série (voir figure 3), sinon d'architecture en parallèle s'ils sont faits simultanément (voir figure 2).

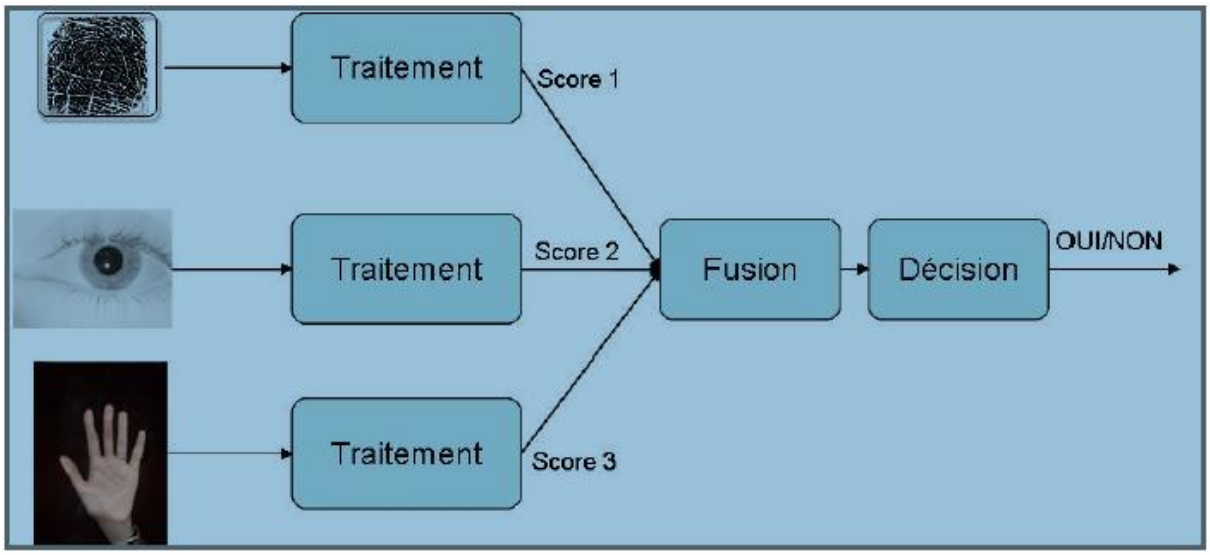

Figure 2 : Architecture de fusion en parallèle

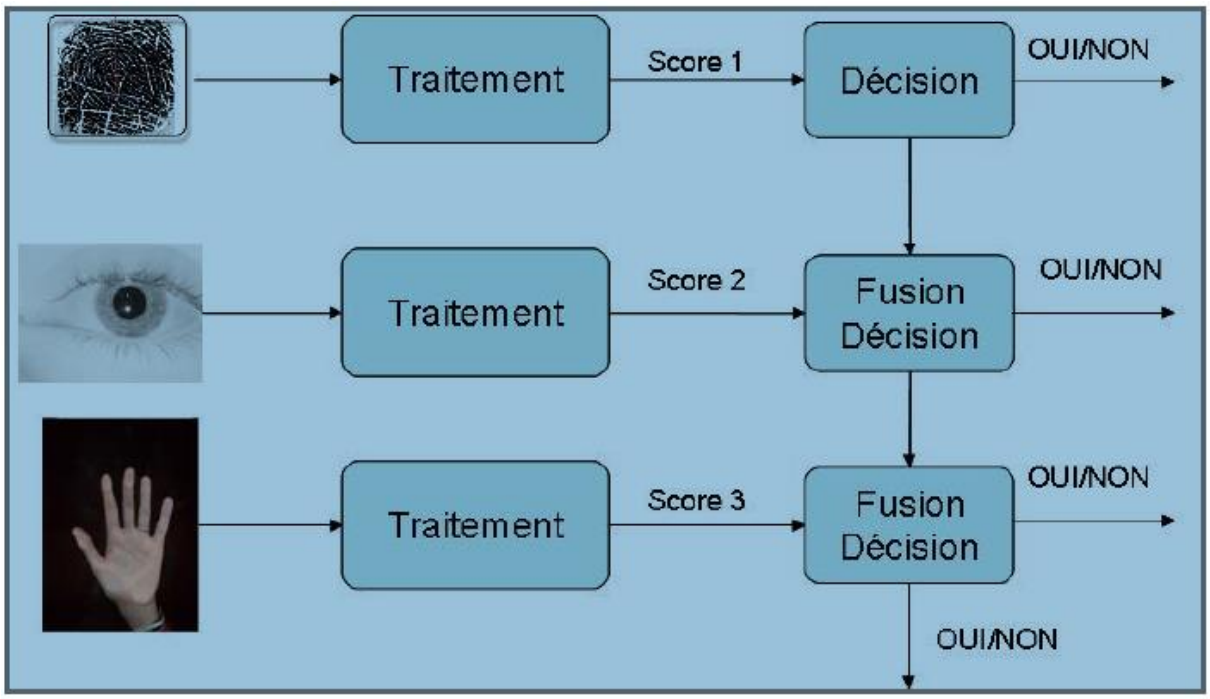

Figure 3: Architecture de fusion en série

II.2.5. Une donnée biométrique :

Une donnée biométrique est un attribut physique et ou comportementale individuel et inimitable d'un être vivant.

II.2.6. L'identification d'un individu par donnée biométrique :

Beaucoup de personne font l'erreur courante de confondre l'identification et l'authentification. Lors d'une authentification, le système connait déjà l'identité de l'utilisateur et doit uniquement vérifier cette identité ; c'est-à-dire qu'il doit décider si l'utilisateur est à priori un imposteur ou pas. Lors de l'identification, l'identité n'est pas connue : le système doit décider lesquelles des images stockées dans une base de données ressemblent la plus à l'image à identifier, il est donc possible d'avoir plusieurs résultats.

II.2.7. L'authentification d'un individu par données biométrique :

Il s'agit de comparer un vecteur X de caractéristiques (extraites à partir de l'image d'un visage par exemple) pour vérifier avec un modèle de vecteurs semblables Yi de caractéristiques extraites déjà enregistrées, ou stockées dans une base de données. La mesure de similarité peut se faire de différentes manières (distance euclidienne, la mesure en angle, la mesure de corrélation). Par exemple si la distance euclidienne entre X et Yi est inférieure à un seuil, le visage dont X est extrait sera considéré comme identique au visage auquel Yi est extrait. Un seuil trop petit mènera à un taux élevé de faux rejet « TFR », et un seuil trop grand mènera à un taux élevé de fausse acceptation « TFA » (voir figure 17).

II.2.8. La reconnaissance faciale :

C'est une technique biométrique consistant à reconnaître quelqu'un par sa photo (celle du visage). Une webcam capture l'image du visage présenté, et l'envoie à un logiciel pour la numériser. Une comparaison avec la base de données déjà existante donnera un résultat positif ou négatif en fonction de la réponse de l'opération.

II.2.9. Un prétraitement :

Le prétraitement est l'ensemble des opérations que subit une image afin d'être utilisé lors d'un traitement, exemple : niveau de gris, égalisation de l'histogramme, recadrage … Il a pour but de maximiser le taux réussite des opérations que subira une donnée.

II.2.10. L'apprentissage supervisé et non supervisé :

II.2.10.1.(a) L'apprentissage supervisé :

Nous avons une image binaire en entrée qui doit être associée à une des classes que nous avons définies. Les classes sont connues à l'avance et la base d'apprentissage est étiquetée avec ces classes.

II.2.10.1.(b) L'apprentissage non supervisé :

On ne connait pas à l'avance les classes possibles. La base d'apprentissage est composée d'images ayant les mêmes caractéristiques. On souhaite regrouper ces images dans une classe.

II.2.11. Classifieur

IL permet de classer dans des groupes (des classes) les échantillons qui ont des propriétés similaires, mesurées sur des observations.

II.2.12. L'algorithme de viola et Jones :

La **méthode de Viola et Jones** est une méthode de détection d'objet dans une image numérique, proposée par les chercheurs Paul Viola et Michael Jones en 2001. Elle fait partie des toutes premières méthodes capables de détecter efficacement et en temps réel des objets dans une image. Inventée à l'origine pour détecter des visages, elle peut également être utilisée pour détecter d'autres types d'objets comme des voitures ou des avions. En tant que procédé d'apprentissage supervisé, la méthode de Viola et Jones nécessite quelques centaines à plusieurs milliers d'exemples de l'objet que l'on souhaite détecter, pour entraîner un classifieur. Une fois son apprentissage réalisé, ce classifieur est utilisé pour détecter la présence éventuelle de l'objet dans une image en parcourant celle-ci de manière exhaustive, à toutes les positions et dans toutes les tailles possibles grâce à des filtres (voir figure 4) (Wikipedia.org, s.d.). [3]

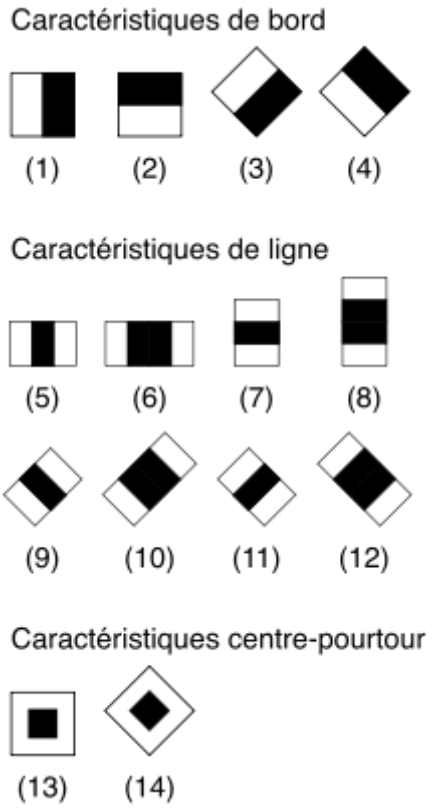

Figure 4: Pseudo-Haar de Lienhart

II.2.12.1. Principe de la méthode Viola et Jones :

La méthode de Viola et Jones est basée sur une approche par recherche exhaustive sur l'ensemble de l'image, qui teste la présence de l'objet dans une fenêtre à toutes les positions et à plusieurs échelles. Cette approche est cependant extrêmement coûteuse en calcul. L'une des idées-clés de la méthode pour réduire ce coût réside dans l'organisation de l'algorithme de détection en une cascade de classifieurs. Appliqués séquentiellement, ces classifieurs prennent une décision d'acceptation :

- Si la fenêtre contient l'objet, l'exemple est alors passé au classifieur suivant
- Sinon si la fenêtre ne contient pas l'objet et dans ce cas l'exemple est définitivement écarté (voir figure 5 et 6).

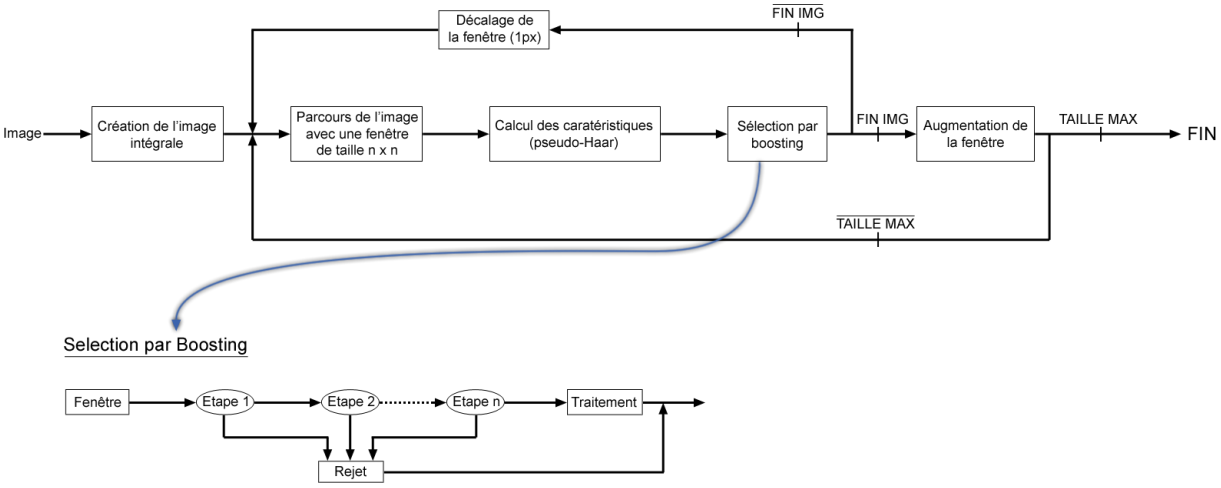

Figure 5: Etapes de détection d'un visage

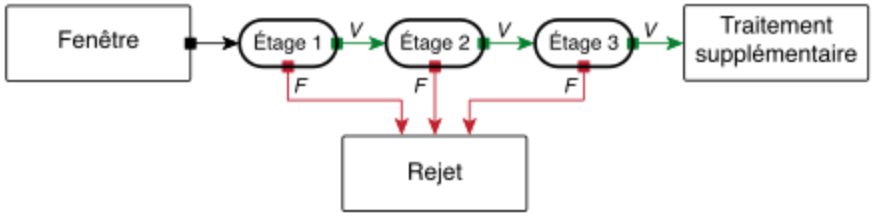

Figure 6: Cascade de Viola & Jones

L'idée est que l'immense majorité des fenêtres testées étant négatives (c'est-à-dire ne contenant pas l'objet), il est avantageux de pouvoir les rejeter avec le moins possible de calculs. Ici, les classifieurs les plus simples, donc les plus rapides, sont situés au début de la cascade, et rejettent très rapidement la grande majorité des exemples négatifs. Cette structure en cascade peut également s'interpréter comme un arbre de décision dégénéré, puisque chaque nœud ne comporte qu'une seule branche.

En pratique, la cascade est constituée d'une succession d'étages, chacune étant formée d'un classifieur fort appris par AdaBoost. L'apprentissage du classifieur de l'étage **n** est réalisé avec les exemples qui ont passé l'étage **n-1** ; ce classifieur doit donc faire face à un problème plus difficile : plus on monte dans les étages, plus les classifieurs sont complexes.

II.2.13. Corrélation, variance et covariance :

La variance permet d'étudier les variations d'une variable par rapport à elle-même et la covariance, quant à elle, permet d'étudier les variations simultanées de deux variables par rapport à leur moyenne respective. La covariance est en statistiques une valeur qui permet de connaitre dans quelle mesure les variables d'une série statistique double évoluent ensemble.

Comme exemple concret, prenons un anthropologue qui se proposerait d'étudier la relation entre la taille et le poids d'individus appartenant à une même communauté. Chaque individu se voit alors doté d'une paire de données, sa taille et son poids, qu'on notera sous la forme d'une paire (x, y). En prenant tous les individus et en utilisant la formule de calcul de la covariance, ce scientifique va peut-être pouvoir établir un lien entre taille et poids.

La formule de la covariance est la suivante (Wikihow.com, s.d.)[4]:

$$C_{xy} = \frac{1}{n-1} \sum_{i}^{n} (xi - xmoy)(yi - ymoy)$$

II-1

Avec :

$C_{x, y}$: La covariance de x et y

X_i : Poids d'un individu

X_{moy} : Poids moyen

Y_i : Taille d'un individu

Y_{moy} : Taille moyenne

II.3. Types de biométrie :

La plupart des concepts clés étant maintenant expliqués, nous nous intéresseront aux types de biométrie auxquels nous pouvons avoir à faire. Nous distinguons deux principaux types de biométrie : la biométrie comportementale et physiologique (définis en concepts clés : page 5). Selon le cas, le système biométrique conçu est programmé pour effectuer l'une des analyses suivantes : biologique, morphologique ou comportementale.

Une analyse morphologique lorsque le traitement de la donnée tient compte de la forme et de la texture comme il en est question avec les empreintes digitales, l'iris, le visage, la main…

Une analyse biologique lorsque le traitement passe outre les traits physiques et que sa nature peut être qualifiée de chimique. Exemple : l'ADN, l'odeur corporelle…

Une analyse comportementale lorsqu'il est question d'une habitude et attitude de l'individu, pour n'en citer que quelques exemples : il s'agit de la signature, démarche, dynamique de frappe au clavier qui sont tout aussi unique que l'empreinte digitale (voir figure 12). Chaque personne a sa propre démarche et signe avec une vitesse, un ordre d'écriture, une pression exercée sur la feuille, une accélération … qui lui est propre.

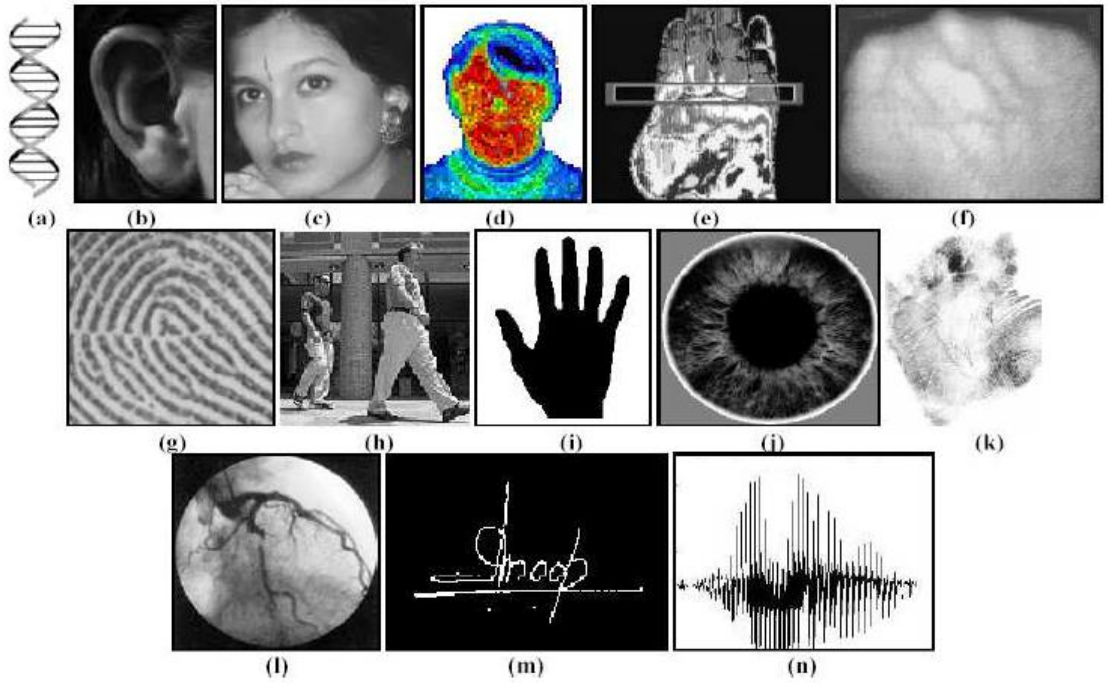

Figure 7 : Caractéristiques biométriques : a) ADN, b) Oreille, c) visage, d) visage infrarouge, e) Thermo gramme main, f) Veine main, g) Empreinte digitale, h) marche, i) geste, j) Iris, k) Empreinte de la paume, l) Rétine, m) Signature, n) Voix

II.4. Techniques biométriques

Un système biométrique en générale est utilisé pour la reconnaissance des données biométriques d'un individu. Ce qui va suivre portera sur le mode de fonctionnement des systèmes de ce type ainsi que la mesure de leur performance. Nous devons noter qu'il existe principalement deux techniques biométriques :

II.4.1. Techniques intrusives :

Un contact direct avec le module d'acquisition est requis de l'individu pour l'identifier. Exemple : les empreintes digitales, l'ris…

II.4.2. Techniques non intrusives :

Aucun contact n'est requis avec le module d'acquisition de données biométriques : le visage, la voix…

II.5. Mode de fonctionnement d'un système biométrique :

Un système biométrique peut compter trois modes :

II.5.1.1. Le mode d'enrôlement :

C'est la phase d'acquisition et d'apprentissage des données (biométriques) grâce à un capteur biométrique, qui va ensuite les numériser pour qu'elles soient exploitables lors d'un stockage en tant que base de données. La quantité et la qualité de ces données ont une influence capitale lors de la reconnaissance surtout lorsqu'on tient compte de la variation temporelle et de l'environnement.

II.5.1.2. Le mode d'authentification :

Dans ce mode, le système est confronté à une comparaison de données dite : 1 à 1. Les données biométriques acquises lors de la précédente phase servent de modèle de comparaison avec les nouvelles données saisies. Selon le résultat de cette comparaison, le système sait si oui ou non l'identité de la personne est vérifiée (car elle existe déjà dans la base de données). Exemple : Carte magnétique, badge, puce électronique…

II.5.1.3. Le mode d'identification :

Dans ce mode, le système tente d'associer à la donnée biométrique saisie une identité (modèle déjà existant dans la base de données) en effectuant une comparaison dite « 1 à N » afin de répondre à la question : qui est cette personne ? L'identité de la personne peut ne pas être dans la base de données.

II.6. Structure d'un système biométrique :

Un système biométrique comporte en générale quatre modules, chacun ayant une tâche bien définie. Les explications qui suivront sont illustrés sur la figure 8 :

II.6.1. Le module capteur biométrique :

Ce module sert à saisir ou lire une donnée biométrique, son rôle étant l'acquisition de données d'apprentissage grâce un capteur de données biométrique.

II.6.2. Le module d'extraction des données :

Les données biométriques déjà acquises sont soumises à un traitement pour en extraire les caractéristiques utiles.

II.6.3. Le module de création de données numériques :

Aussi appelé module de signature, ce module créé un modèle (stockable dans une base de données) à partir des caractéristiques de la donnée biométrique acquisse. Le modèle représentera désormais l'individus en question.

II.6.4. Le module de comparaison :

Il permet de comparer le modèle présent dans la base de données par rapport au modèle nouvellement saisi lors d'une authentification ou d'une identification :

II.6.5. La base de données :

C'est là où est stocké tous les modèles à la fin du processus d'enrôlement.

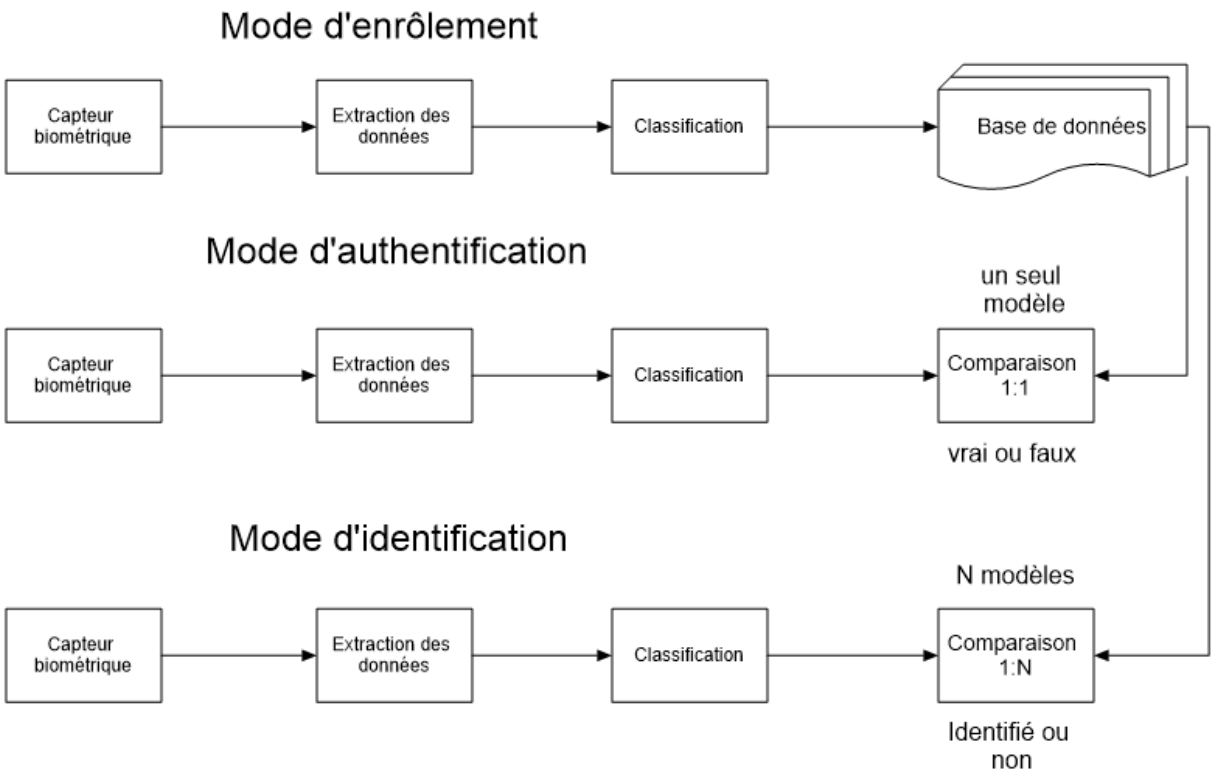

Figure 8: Structure d'un système biométrique

II.7. Performance d'un système biométrique

Tout système biométrique doit être validé par un test de vérification et d'identification avant d'être qualifié fiable. Pour évaluer les performances d'un système biométrique, il faut effectuer plusieurs tests puis faire le rapport « identification correcte sur nombre total de test ». L'erreur commise par ce type de systèmes est d' attribuer à l' individu présenté une identité autre que la sienne.

$$P = \frac{nombre\ \ d'authentifications\ correctes}{nombre\ total\ de\ test\ effectué}$$

II-2

II.7.1. Vérification d'une identité (authentification) :

L'identité de la personne étant déjà déterminée, ce test vise à confirmer ce qui est déjà établi. Grâce auquel, nous pouvons aborder deux nouveaux concepts : le taux de faux rejet et de fausses acceptations.

II.7.1.1. Le taux de faux rejet : False Rejection Rate (FRR)

Le taux de faux rejet (FRR) exprime le pourcentage de rejet inapproprié, il s'agit de l'erreur qui survient lorsque l'identité est rejetée alors qu'elle devrait être acceptée.

II.7.1.2. Le taux de fausses acceptations : False-Aceptance Rate (FAR)

A l'inverse du taux de faux rejet (FRR), il s'agit de l'erreur qui survient lorsqu'une identité est acceptée alors qu'elle devrait être rejetée (voir figure 9).

Soit S le score de similarité entre le vecteur caractéristique V_C de l'identité proclamée I de la personne « P » et le vecteur caractéristique V_I de l'identité stockée dans la base de données. Soit α le seuil du taux d'exactitude croisé (Equals Error Rate EER) :

$$P(I,VC) = \begin{cases} accepté\ si\ S(VI,VC) \geq \alpha \\ réfusé\ sinon \end{cases}$$

II-3

I : Identité de la personne P

V_C : Vecteur caractéristique de la personne P

V_I : Vecteur caractéristique déjà enregistré

S : Score de similitude

α : Seuil du taux d'exactitude croisé

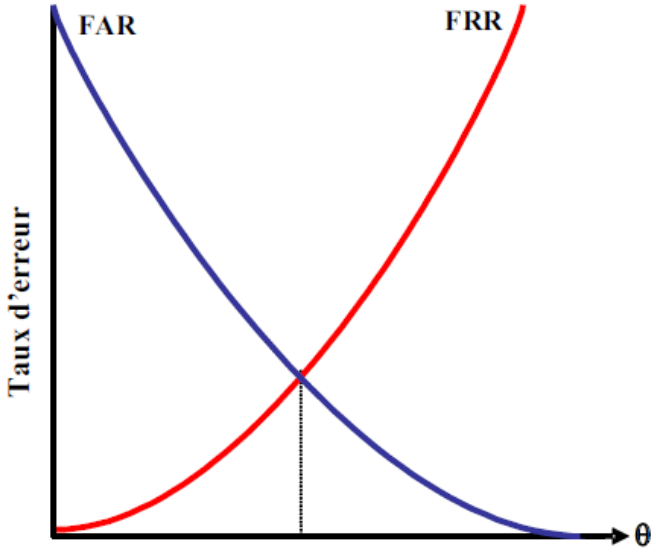

Figure 9: Courbe du point d'équivalence des erreurs dans un système biométrique

Le seuil est pris au croisement des deux courbes car s'il est trop petit, cela engendre une augmentation du FRR, et s'il est trop grand, cela engendre une augmentation du FAR (voir figure 10).

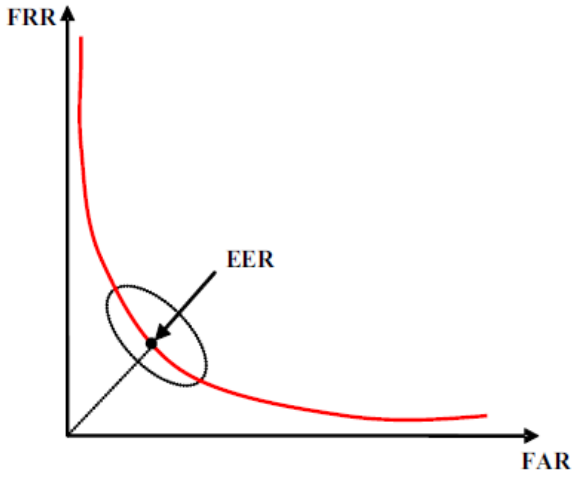

Figure 10: Courbe FRR en fonction du FAR "Detection Error trade-off (DET)"

II.7.2. Identification d'une personne :

Très Couramment utilisée, il cherche l'identité I d'une personne P parmi N dans la base de données en comparant les caractéristiques VC et VI avec un seuil α, en cas d'erreur, il est toujours possible de savoir si le bon choix se trouve parmi les *N premières réponses du système*.

$$P(I,VC) = \begin{cases} Ik \ si \ max \ \ S(VIk,VC) \geq \alpha \\ IN+1 \ \ sinon \end{cases} \quad k = 1 \ldots N \ \ldots\ldots\ldots \ II\text{-}4$$

I : identité de la personne P

V_C : Vecteur caractéristique de la personne P

V_I : Vecteur caractéristique déjà enregistré

S : Score de similitude

α : seuil du taux d'exactitude croisé

I1…IN : identités enrôlées

I_{N+1} : Identité rejetée

II.8. Etat de l'art de la reconnaissance faciale :

La reconnaissance faciale a fait l'objet de plusieurs travaux ces deux dernières décennie, plusieurs techniques ont été développées pour améliorer les résultats obtenus jusqu'alors et sont classées selon deux groupes : la reconnaissance faciale à deux dimensions (2D) et trois dimensions (3D). La détection est une étape importante de la reconnaissance des formes, un algorithme a donc été conçu permettant de reconnaitre diverses formes grâce à des filtres : l'algorithme de Viola et Jones (que nous utiliserons dans notre travail pour détecter les visages). Il existe cependant plusieurs autres algorithmes tel que :

- Les approches basées sur des caractéristiques invariantes
- Le template matching
- L'apparence (Viola & Jones)
- La détection de visages basée sur l'analyse de la couleur de la peau
- Les approches paramétriques
- Les approches semi-paramétriques…

Chaque algorithme procède d'une manière qui lui est propre, mais en générale, ceux qui utilisent un masque tienne compte de ces principaux points voir plus encore sur le visage (voir figure 11) :

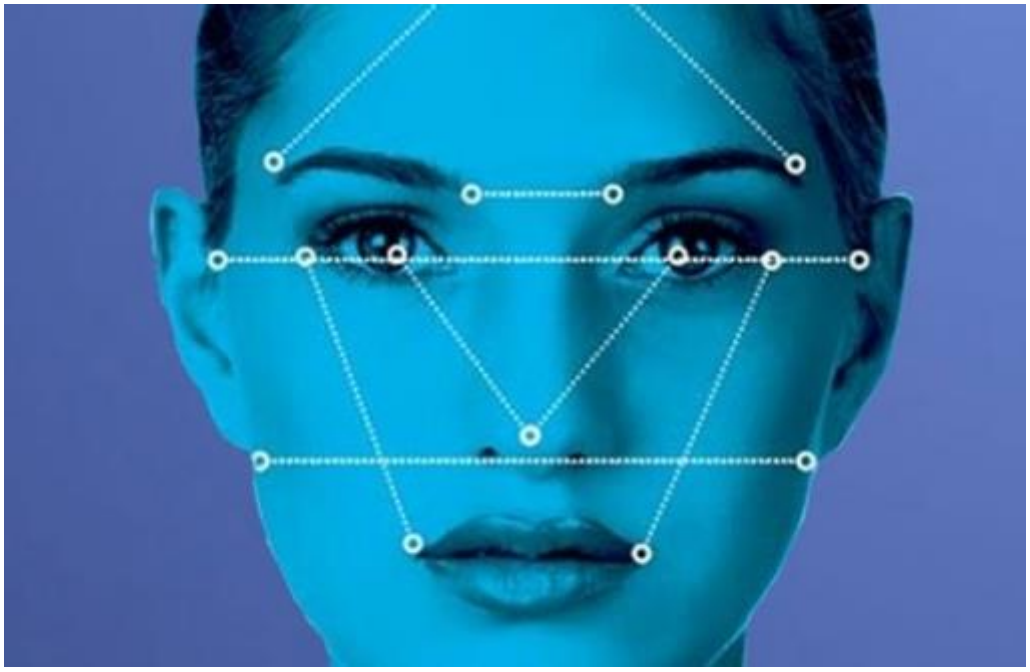

Figure 11 : quelques points caractéristiques du visage

II.8.1. Quelques principales techniques de reconnaissance faciale 2D

II.8.1.1. Approche locale :

Les méthodes locales extraient les caractéristiques locales du visage telles que la bouche, le nez et les yeux puis utilisent leur géométrie et/ou leur apparence comme donnée d'entrée du classificateur. Les méthodes locales peuvent être classées en deux catégories :

- Celles basées sur les points d'intérêt : on extrait les caractéristiques après détection de ces points
- Celles basées sur l'apparence du visage : on extrait les caractéristiques sur les patchs (petites régions servant à diviser le visage.

II.8.1.1.(a) Exemple 1 : Motifs binaires locaux (LBP) :

Les motifs binaires locaux (Local Binary Patterns) utilise des descripteurs se basant sur la comparaison d'un pixel ou du moins son niveau de luminescence par rapport celles de ses proches pour la classification des textures, la détection et le suivi des objets mobiles dans une séquence d'image. Sa valeur est la somme pondérée par un certains poids (code binaire).

Exemple : Tous les voisins dans une région de taille R (3x3) prendront alors une valeur "1" si leur valeur est supérieure ou ´égale au niveau de gris de pixel à analyser (central) et "0" autrement. Les pixels de ce motif binaire sont alors multipliés par des poids (code binaire) et

sommés afin d'obtenir un code LBP du pixel courant, par conséquent on obtient une image sur 8 bits. La figure 12 illustre la procédure de calcule de LBP sur une fenêtre de taille 3x3. Le descripteur LBP de texture d'une image pourrait donc être décrit par l'histogramme de dimension 255, (Doanh, 2010) [5].

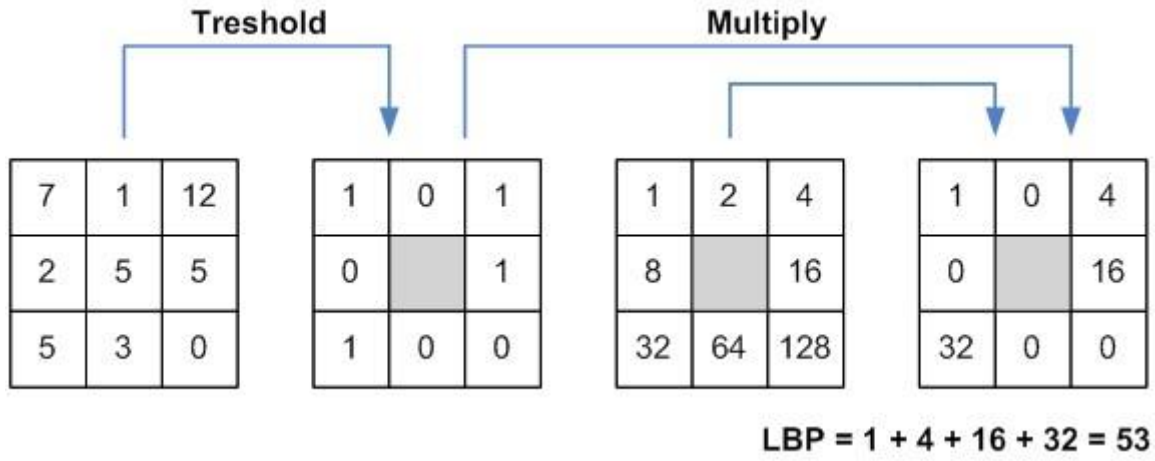

Figure 12 : Construction d'un motif local binaire et calcule du code LBP

II.8.1.1.(b) Exemple 2 : Machine à vecteur de support (SVM) :

Les machines à vecteurs de support ou séparateurs à vaste marge, sont un ensemble de techniques d'apprentissage supervisé destinées à résoudre des problèmes de discrimination et de régression. Les SVMs sont une généralisation des classifieurs linéaires. Les séparateurs à vastes marges sont des classificateurs qui permettent de traiter des problèmes de discrimination non linéaire, et de reformuler le problème de classement comme un problème d'optimisation quadratique. C'est une technique qui a été proposée par V.Vapnik en 1995, elle est utilisée dans plusieurs domaines statistiques (classement, régression, fusion,… etc.) et en reconnaissance faciale pour la classification en général.

Les SVMs peuvent être utilisés pour résoudre des problèmes de discrimination, c'est-à-dire décider à quelle classe appartient un échantillon, ou de régression (prédire la valeur numérique d'une variable).

II.8.1.2. Approche globale (techniques linéaires)

Les méthodes globales (linéaires et non linéaires) utilisent la région entière du visage comme entrée du système de reconnaissance, la zone d'intérêt est représentée par un seul vecteur de grande dimension en concaténant les niveaux de gris de tous ses pixels. Elles peuvent ainsi

conserver toutes les informations (texture et forme) nécessaires pour identifier des visages mais aussi elles tiennent compte de la structure globale du visage.

II.8.1.2.(a) Analyse en composantes principales (ACP) :

L'analyse en composantes principales vise à réduire la dimension des variables, en exploitant la corrélation potentielle entre les variables initiales. Pour cela, elle propose de nouvelles variables non corrélées et ordonnées de tel façon que les premières retiennent le plus possible la variation présentée dans les données initiales [6]. En terme plus mathématique, L'analyse en composante principale créé une matrice de covariance à partir des images de la base d'apprentissage, puis en extrait les différents vecteurs propres. Introduite en 1991 par De MA. Türk et AP.Pentland au MIT Media Lab, il utilise des vecteurs propres et des valeurs propres (respectivement Eigenvectors et Eigen values en anglais).

- **Le vecteur d'image :**

Une image X_i de dimension (m, n) correspond alors à un vecteur V_i (m*n, 1) (obtenu par concaténation des colonnes de X_i) de dimension (D = n*m) dans un espace vectoriel. L'ensembles des vecteurs forme la matrice V dont chaque colonne représente une image (visage) tel que :

$$X_i = \begin{bmatrix} a1,1 \dots a1,m \\ . \\ . \\ . \\ an,1 \dots an,m \end{bmatrix} \text{ donne le vecteur d'image } V_i = \begin{bmatrix} a1,1 \\ . \\ . \\ . \\ a1,m \\ . \\ . \\ . \\ an,m \end{bmatrix} \text{ qui forme la grande matrice :}$$

II-5

$$V = [V1\ V2 \dots VN] = \begin{bmatrix} a1,1 & b1,1 & & z1,1 \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ a1,m & b1,m & \dots & z1,m \\ . & . & & . \\ . & . & & . \\ . & . & & . \\ an,m & bn,m & & zn,m \end{bmatrix} \text{ avec N le nombre d'image}$$

II-6

- **L'image moyenne**

Le calcul du centre de gravité dite « Image moyenne » s'effectue par la moyenne de chaque image représentée par le vecteur V_i : soit M cette moyenne,

$$M = \frac{1}{N} \sum_{i=1}^{N} Vi \ldots\ldots\ldots\ldots \text{ II-7}$$

- **Ajustement des images**

Le Vecteur d'image M est ensuite soustrait du vecteur d'image Vi, c'est-à-dire de chaque vecteur d'image des N images : ce procédé est appelé ajustement par rapport à la moyenne

Soit $\mathbf{a_i}$ cette matrice :

$$\boldsymbol{a_i = V_i - M} \text{ avec i allant de 1 à N}$$

- **La matrice de covariance :**

La matrice de covariance est alors obtenue en sommant le produit matriciel de chaque image ajustée par son transposé : soit C la matrice de covariance,

$$C = \sum_{i=1}^{N} .a_i * a_i^T = AA^T \ldots\ldots\ldots\ldots \text{ II-8}$$

- **Le calcul des vecteurs et valeurs propres :**

Le principe de l'analyse en composante principale étant de réduire l'information en limitant les composantes, nous considèrerons une matrice $E = A^T A$ de taille m×m, dont nous trouverons les vecteurs propres e_i.

$$E\,ei = (A^T)\,Aei = \lambda_i e_i \ldots\ldots\ldots\ldots \text{ II-9}$$

Avec « λ_i » la valeur propre associée au vecteur propre « e_i ».

- **Calcul des vecteurs propres de la matrice de covariance C et l'obtention des visages propres :**

En multipliant cette équation par la matrice A : $AEei = (AA^T)\,Aei$ d'où $AEei = CAei = \lambda_i\,Aei$

- **Meilleur choix des valeurs et vecteurs propres associés :**

Pour e_i vecteur propre de la matrice E associé à la valeur propre λ_i, nous avons un vecteur propre de la matrice C associé à la même valeur propre λ_i : Aei. Nous avons d_i vecteur propre de C, avec :

$$d_i = Aei \ldots\ldots\ldots\ldots\ldots\ldots \text{II-10}$$

- **Détermination du poids des images d'entrée :**

Pour chacune des coordonnées correspondant à un visage d'apprentissage, nous devons trouver

$$P_i = d_i^T A_i \ldots\ldots\ldots\ldots\ldots\ldots \text{II-11}$$

- **L'espace visage :**

Pour chacun des m visages d'apprentissages, nous avons un vecteur π_i où i représente le i^{eme} visage :

$$\pi_i = \begin{bmatrix} p1 \\ p2 \\ pl \end{bmatrix} \ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{II-12}^e$$

Cette approche nous permettra de mieux interpréter la projection de l'image. Maintenant passons à la phase d'identification. Cette phase consiste tout d'abord à soustraire le visage moyen du vecteur image et obtenir les caractéristiques propres à ce visage :

a = V – M v étant le vecteur image

A partir de cette caractéristique propre nous pouvons calculer le poids qui est donnée par :

$$P_i = d_i^T a \ldots\ldots\ldots\ldots\ldots\ldots \text{II-13}$$

L'espace visage sera donné alors par :

$$\pi = \begin{bmatrix} p1 \\ p2 \\ pl \end{bmatrix} \ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{II-14}^e$$

Nous mesurons ensuite la distance euclidienne entre les points à comparer, après quoi nous cherchons le minimum de cette distance tel que :

$m = min \mid \pi - \pi i \mid$ ..... II-15 : *Distance minium entre les point du visage patron*

Cette valeur « m » est enfin comparée à notre valeur seuil, déterminée après plusieurs tests et dépendant fortement de la précision qu'on veut accorder au système.

II.8.1.2.(b) Analyse discriminante linéaire (LDA : Fisher faces) :

L'**analyse discriminante linéaire** ou **ADL** (en anglais, *linear discriminant analysis* ou *LDA*) fait partie des techniques d'analyse discriminante prédictive, l'algorithme LDA est né des travaux de Belhumeur et al. De la Yale University (USA), en 1997. Il s'agit d'expliquer et de prédire l'appartenance d'un individu à une classe (groupe) prédéfinie à partir de ses caractéristiques mesurées à l'aide de variables prédictives. La variable à prédire est forcément catégorielle (discrète). Les variables prédictives sont toutes continues a priori. Son principe est le suivant :

Étroitement lié à l'ACP (PCA en anglais : principal component analysis, car ils utilisent une combinaison linéaire l'une comme l'autre). Le Fisher faces essaye explicitement de modeler la différence entre les classes des données. L'Eigen faces quant à elle, ne tient pas compte des différences entre les classes.

Chaque visage, qui se compose d'un grand nombre de Pixel, est réduit à un plus petit ensemble de combinaisons linéaires avant la classification

Chacune des nouvelles dimensions est une combinaison linéaire des valeurs de pixel, qui forment un Template. Les combinaisons linéaires obtenues en utilisant FLD s'appellent les Fisher faces, en analogie avec les Eigen faces. LDA est une technique qui cherche les directions qui sont efficaces pour la discrimination entre les données [7].

II.8.1.3. Approche globale Technique non linéaire

II.8.1.3.(a) Kernel-PCA (Kernel Principal Component Analysis) :

L'intérêt porté à l'ACP (analyse en composante principale) pour résoudre des problèmes d'apprentissage a été récemment relancé par l'obtention d'une version non-linéaire de cet algorithme : (la Kernel PCA). La Kernel PCA permet d'exploiter des relations potentiellement non linéaires entre les variables. Le principe de la version non linéaire de l'ACP consiste à

envoyer préalablement les données Xj par une application $\phi : X \rightarrow H$ (appelée Feature map) dans un espace linéaire de grande dimension H muni d'un produit scalaire.

La Kernel PCA agit sur les ϕ(Xj) de la même façon que l'ACP agissait sur les Xi. Ainsi, la KPCA correspond à une ACP dans un espace de grande dimension H comme nous le montre la figure 13 ([Ben Khediri (2011)]). [8]

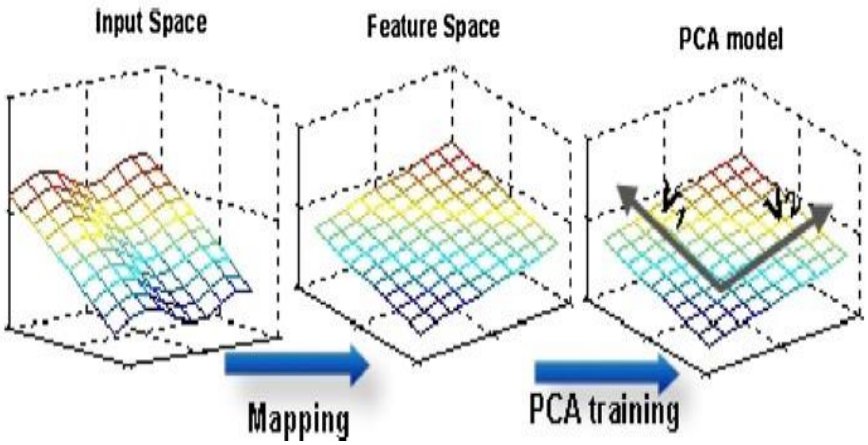

Figure 13 : Espace dimensionnelle de la K-PCA

II.8.1.4. Approche hybride :

Les méthodes hybrides permettent de combiner les avantages des méthodes globales et locales pouvant ainsi améliorer la stabilité et la performance de reconnaissance lors de changements de pose, d'éclairement et d'expressions faciales.

II.8.1.5. Observation :

Lors d'incessant expériences, il a été établit que les méthodes globales et locales réagissent différemment aux variations imposées par l'environnement (voir tableau 1) :

Tableau 1: Comparaison des méthodes locales et globales

Variations	Locales	Globales
Lumière 	Insensible	Sensible
Expression 	Insensible	Sensible
Pose 	Sensible	Très Sensible
Bruit	Très Sensible	Sensible
Petite variation	Insensible	Sensible

Il existe d'autres algorithmes ou méthodes pour la reconnaissance du visage (voir figure 14) :

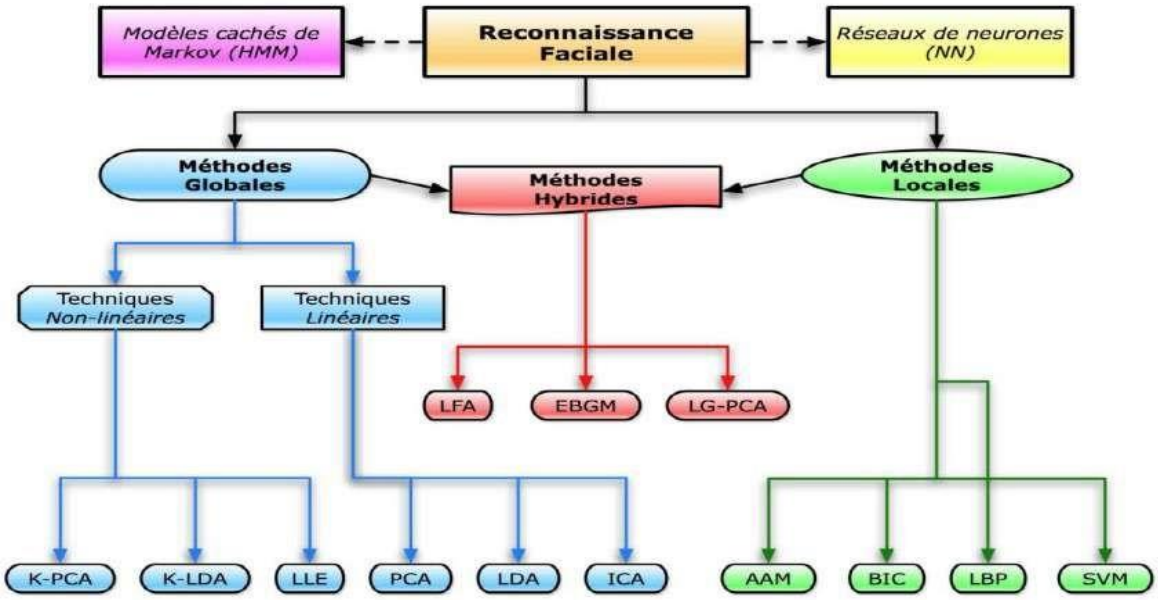

Figure 14 : Classification des algorithmes d'apprentissage

II.8.2. La reconnaissance faciale 3D :

Les approches de reconnaissance faciale à trois dimensions (3D) actuelles sont soit des extensions d'approches de deux dimensions (2D), comme l'analyse en composante principale ACP, soit des adaptations d'approches 3D de reconnaissance de formes rigides, comme l'ICP (par initialisation du recalage des surfaces faciales mais difficile à mettre en œuvre).

La grande différence avec la reconnaissance à deux dimensions (2D) est que cette dernière donne des informations sur la texture faciale contrairement au 3D fournissant des informations sur la structure même du visage.

Son acquisition peut se faire par des capteurs comme pour la reconnaissance 2D : caméras temps de vol, les scanners lidar, scanners 3D, systèmes de stéréo, etc.

II.9. Conclusion de chapitre

Ce chapitre se clôt par la justification de notre choix de méthodes ainsi que les difficultés liées à la reconnaissance faciale :

- La reconnaissance faciale, par rapport aux autres techniques biométriques est plus appréciée du public. Elle peut cependant rencontrer beaucoup de difficulté en deux dimensions lorsque les caractéristiques saisies sont soumises à une luminosité différente à celles de la base de données, ainsi que la pose de la personne lors de l'acquisition, son

expression, le fait qu'une partie du visage soit caché par une lunette ou autre chose (lorsqu'une méthode globale est utilisée pour la mettre en œuvre).

Le but principal de la biométrie est de rivaliser ou même surpasser les capacités de reconnaissance humaine selon une étude de 2006 qui fit la conclusion suivante (P. Sinha, 2006) [9] :

- Les capacités de reconnaissance d'un proche par un homme ne sont pas amoindries même quand l'image est de faible résolution, cela est dû à sa familiarité avec le même visage au quotidien.
- Pour obtenir une meilleure performance en reconnaissance faciale, la texture, la pigmentation et le contour sont tout aussi important que le contour de la forme du visage généralement codé de manière caricaturale et les sourcils sont d'une importance capitale
- Les caractéristiques faciales sont traitées de manière holistique
- La couleur joue un rôle important spécialement lorsque la forme est dégradée
- Les changements d'illumination influencent la capacité de généralisation
- Le mouvement des visages semble faciliter la reconnaissance de manière conséquente
- Le système visuel progresse d'une stratégie locale vers une stratégie holistique au cours des premières années de la vie
- Identité faciale et expressions sont traitées par des systèmes séparés

Compte tenu de toutes ces difficultés ainsi que des avantages respectifs des méthodes globales et locales, nous avons opter pour un système biométrique multimodale à base de la méthode globale d'analyse en composante principale (ACP) combinée à la technique multi-algorithme présentant les avantages suivants :

- L'ACP contrairement aux méthodes locales peut faire de la reconnaissance faciale même avec une seule image en donnée d'acquisition, toute fois sa performance s'en retrouve réduite, elle présente la possibilité d'être amélioré en reconnaissance 3D présentant moins de faille que la reconnaissance 2D mais difficile à mettre en œuvre.
- Un second algorithme, traitant les données d'acquisition, apporte de la stabilité et robustesse au premier programme qui présente une très forte sensibilité à l'illumination, la pose, les bruits, les expressions faciales… Son but est de remédier à cette sensibilité qui dans notre cas peut être nuisible pour la reconnaissance des individus. Le système ainsi conçu sera présent dans les prochains chapitres.

III. Expérimentation et discussion de résultat

III.1. Introduction

Dans ce chapitre, il sera question de matériel, de logiciel et de techniques d'implémentation de notre système de reconnaissance.

III.2. Environnement de travail

Nous avons fait nos différents traitements en langage python (version 3.7) avec Open CV. Ce choix se justifie facilement par la puissance de ce langage qui se veut performant et facile d'implémentation. Il compte plus de bibliothèques et est présent dans le développement web comme dans d'autres types de programmation. Ces dernières années plusieurs recherches ont été entreprises afin de l'améliorer et aujourd'hui il se trouve tout en haut de la liste parmi les langages les plus performants comme le Java, C#...

III.2.1. Caractéristique du matériel utilisé pour implémenter le programme

III.2.1.1. Camera

Nous avons utilisé une caméra d'une résolution de 5 mégapixels, lentille ajustable, reconnaissance de microphone externe, résolution d'image : 640x480, fréquence 60 HZ, USB 2.0.

III.2.1.2. L'ordinateur portable

L'ordinateur portable a les caractéristiques suivantes : 8 giga de mémoire Ram DDR4, processeur Intel core i5 HD graphic 620, septième génération, disque dur SSD.

III.2.2. Les packages et leur utilisation

Dans le cadre de l'élaboration de notre système, nous avons installé et utilisé les bibliothèques suivantes : Open CV, Matplotlib, numpy, PIL, OS et DLIB.

III.2.3. Open cv

C'est la bibliothèque dont nous nous serviront pour nos opérations de traitement d'image. Elle existe pour toutes les versions de python et même pour les IDLE comme Anaconda…

III.2.4. Matplotlib

Cette bibliothèque permet d'effectuer des opération graphiques telles tracer des courbes faire ressortir des figures etc.

III.2.5. Numpy

La bibliothèque « Numérique python » quant à lui effectuer les opérations mathématiques en générale telle que le calcul sur les matrices, l'arithmétique etc.

III.2.6. Os

L'Os, ou operating system nous permettra de manipuler les commandes systèmes dans notre ligne de code.

III.2.7. Dlib

Cette bibliothèque est une des plus puissantes car elle regroupe beaucoup de fonction aidant au codage d'apprentissage machine tel que l'ACP, LDA, SVM… ce qui nous facilitera grandement la tâche dans notre programme. Les détails de son installation sont fournis par le site : https://www.learnopencv.com/install-dlib-on-windows/

III.3. Phase d'apprentissage

L'apprentissage est une phase pouvant comporter jusqu'à quatre étapes ou plus (voir figure 15) selon le système de reconnaissance ou la méthode utilisée. En générale, il comprend :

III.3.1. Acquisition de la donnée biométrique

L'acquisition de la donnée biométrique, une image dans notre cas passe par plusieurs étapes dont les suivantes :

III.3.1.1. Acquisition de l'image :

A l'aide d'un module d'acquisition d'image tel qu'une cameras photo, nous établissons un visuel sur notre cible avant de déterminer la zone d'intérêt : il permet d'établir le visuel sans détecter ni capturer l'image qui sont des étapes à venir.

III.3.1.2. Détection du visage :

Une fois le visuel établi, nous nous intéresseront à une zone en particulier, le visage humain. Pour ça, nous importons un ficher xml aussi appelé classifieur haarcascade_frontalface

permettant de détecter un visage humain dans une image. Le mécanisme alors utilisé est dit "photographique", car dans le visuel établi, le système va d'abord trouver et enregistrer la position des yeux, puis plusieurs points caractéristiques sur le visage et évaluer la distance entre certains points pour permettre une meilleure identification. Le patron visage est ainsi obtenu (le visage transformé en point).

III.3.1.3. Capture du visage :

Une fois le patron obtenu, il est enregistré dans la base de données. Lors des tests d'identification ou d'authentification, on utilise alors une modélisation du patron visage dans le but de la comparer au modèle de la base de données et ainsi identifier ou authentifier l'individu sur la photo.

III.3.1.4. Base de données :

D'une part nous avons les photos stockées dans un dossier (Features) ou plus précisément un classifieur et d'autres parts un fichier comportant la liste des identifiants ainsi que leurs données personnelles qui sont liés par le programme d'apprentissage.

III.3.2. Prétraitement

Le prétraitement est l'ensemble des techniques utilisées afin d'améliorer la fiabilité et la robustesse du systèmes en réduisant au maximum les variations pouvant rendre plus complexe que nécessaire la reconnaissance tels que : la couleur, la lumière, les bruits, l'axe sur lequel se trouve les yeux …

Parmi les opérations de prétraitements nous avons utilisé :

- La conversion de l'image en niveau de gris et son recadrage (100x100 pixel)

- L'égalisation de l'histogramme

- 

- Mettre les yeux sur un même axe

- 

III.3.3. Extraction des caractéristiques et classification en fichiers XML

Une fois la base de données mise en place, nous créons un classificateur, un modèle à partir des points projetés et extraits du le visage.

III.3.4. Organigramme de la phase d'apprentissage

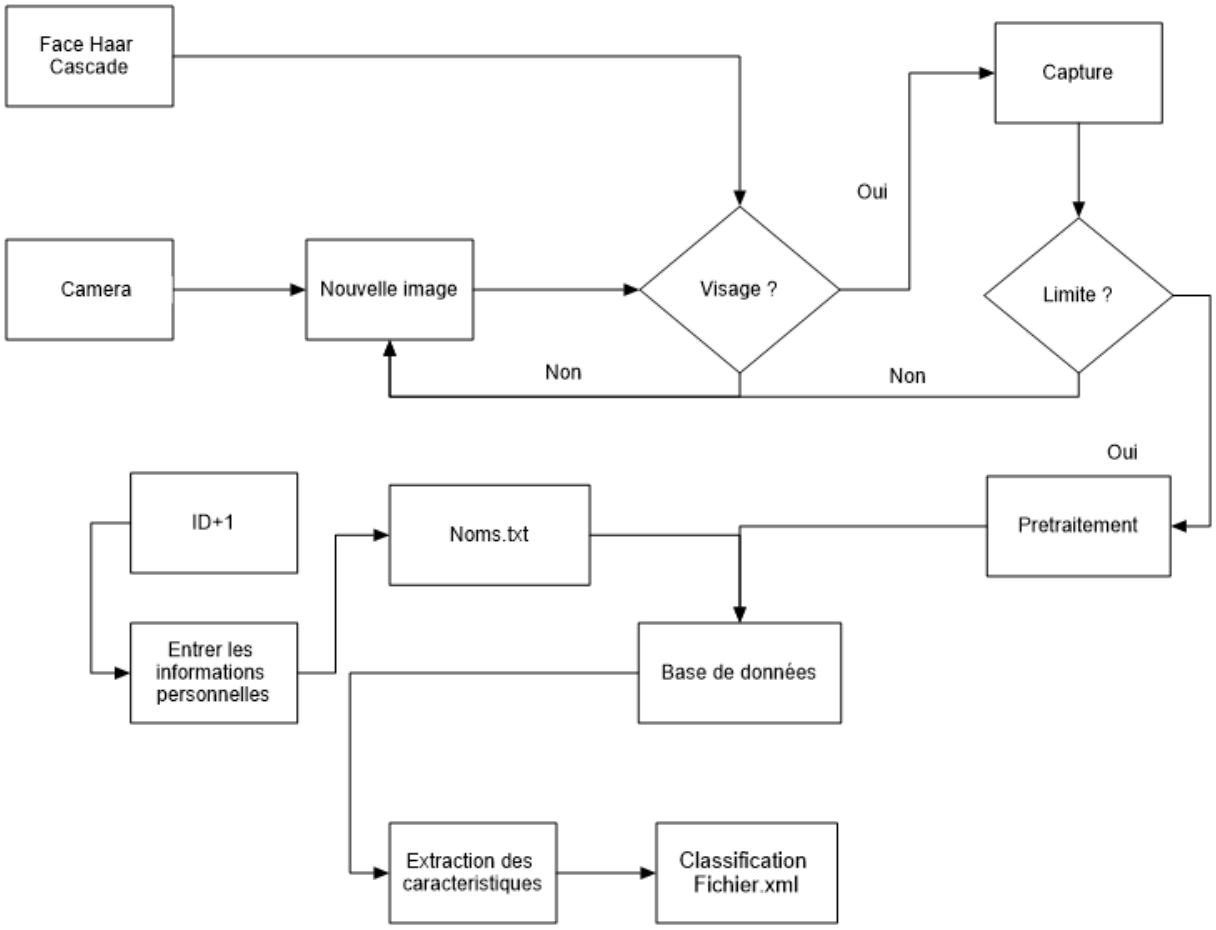

Figure 15 : Phase d'apprentissage

III.4. Phase de reconnaissance

La phase de reconnaissance est à peu près similaire à celle de l'apprentissage sauf qu'au lieu de classifier la donnée acquise, une comparaison (identification ou authentification) est effectuée avec la base de données (voir figure 16).

III.4.1. Organigramme de la reconnaissance par Eigen face

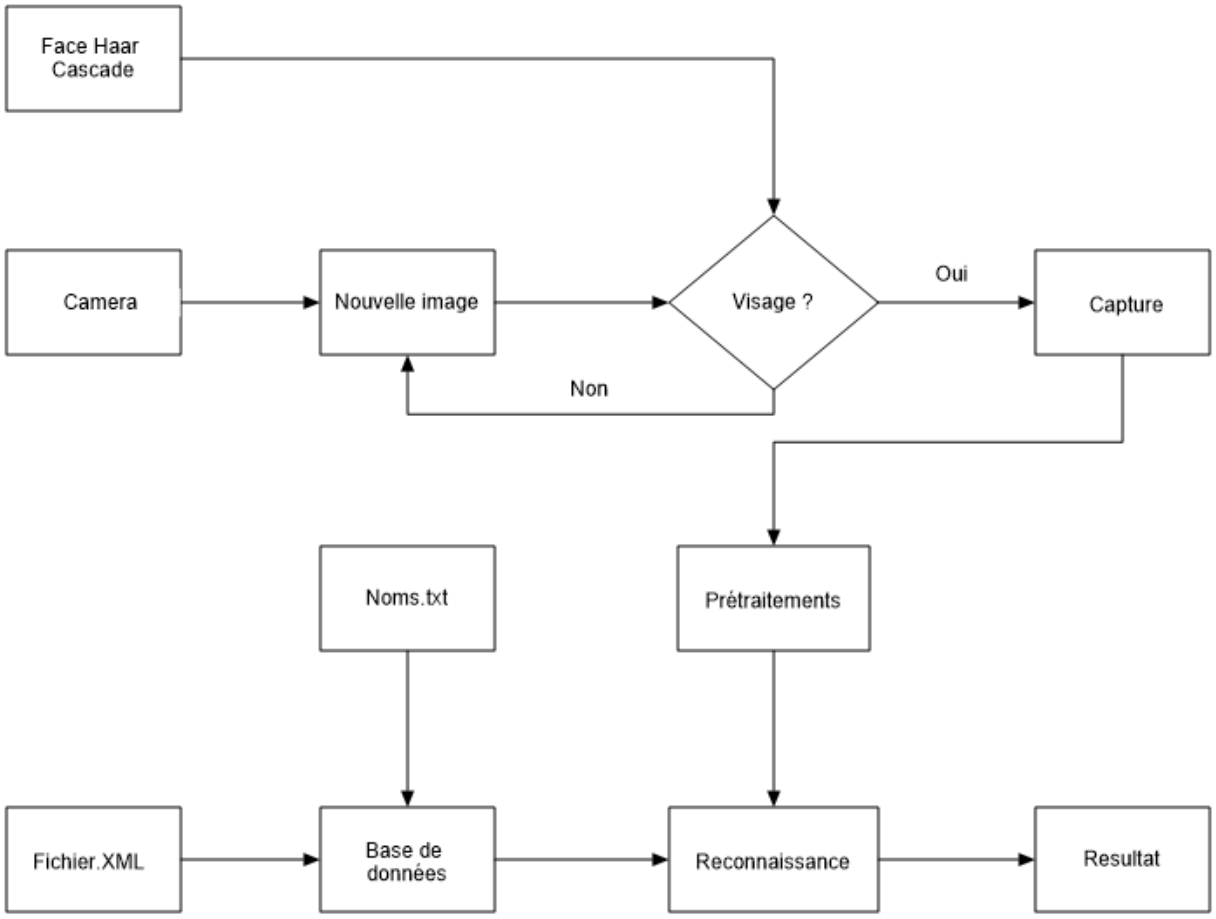

Figure 16 : Phase de reconnaissance

III.5. Conclusion

La reconnaissance faciale passe en générale par deux principaux phases de traitement, l'apprentissage machine qui consiste à entrainer l'ordinateur à reconnaitre un individu en question sur une image statique ou dynamique, et la reconnaissance : c'est le traitement consistant à donner un résultat lorsque la donnée biométrique faciale est comparée par un modèle en particulier dans la base de données ou tous les modèles.

IV. Implémentation

IV.1. Introduction

Il vous est présenté dans ce chapitre notre système de reconnaissance faciale que nous avons nommé FARES-MAK : Face Recognition System of Maïga And Karabenta. C'est une application monoposte dotée d'une interface assez simple et facile à manipuler. Pour implémenter ce système, nous avons utilisé le module d'interface graphique **PAGE** pour définir le design des différentes fenêtres, puis **Tkinter** pour configurer chaque bouton et option.

IV.2. Le système de reconnaissance conçu : Le FARES-MAK

Le FARES-MAK (Face Recognition System of Maïga And Karabenta) est un logiciel de reconnaissance de forme, plus spécifiquement du visage, destinée à une application sécuritaire. Par l'identifier et l'authentifier un individu donné.

IV.2.1. Interface du logiciel et mode de fonctionnement

Son interface comprend un accueil à l'ouverture donnant sur quatre boutons : Base de données, Détection, Reconnaissance faciale et Aide/A propos.

IV.2.1.1. Accueil

L'accueil du FARES_MAK compte quatre boutons de menu et un bouton de fermeture (voir figure 17).

Figure 17 : Interface d' accueil du FARES_MAK

Voici respectivement les taches qu'accomplissent les boutons : Base de données, Détection, Reconnaissance faciale, Aide/A propos :

IV.2.1.2. Base de données

L'option « Base de données » permet de créer une nouvelle base de données, modifier une donnée déjà existante ou carrément la supprimer, visualiser toutes les données et les mettre à jour (voir figure 18).

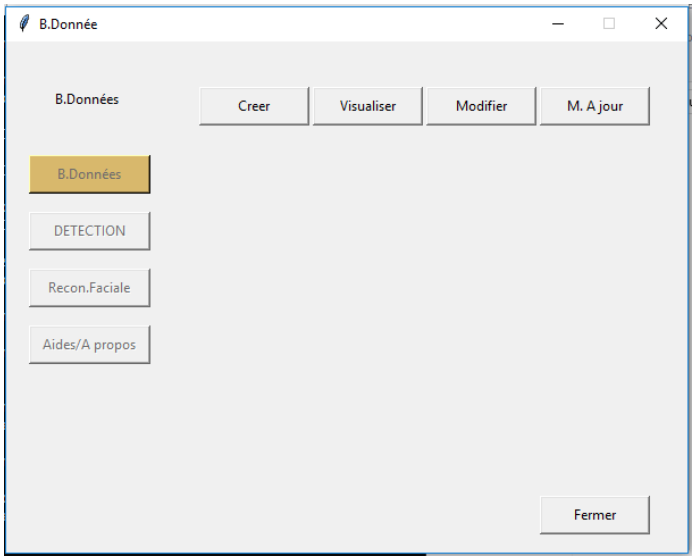

Figure 18 : La fenêtre base de données

IV.2.1.2.(a) Le bouton 'Créer'

Le bouton « **créer** » ouvre sur une fenêtre qui génère automatiquement l'identifiant de la nouvelle donnée à créer dans la base, dont les informations personnelles de la personne viennent compléter. Une fois les informations saisies, nous avons le choix entre importer une photo ou l'enregistrer (prendre une photo avec la webcam ou la caméra). La dernière action sera retenue, c'est-à-dire, après avoir pris une photo, si on en importe une autre alors l'action la plus récente (importation de photo) sera celle pris en compte par le logiciel. Ensuite le bouton '**Valider**' qui était désactivé jusqu'alors permettra de terminer le traitement de création d'une nouvelle donnée (voir figure 19).

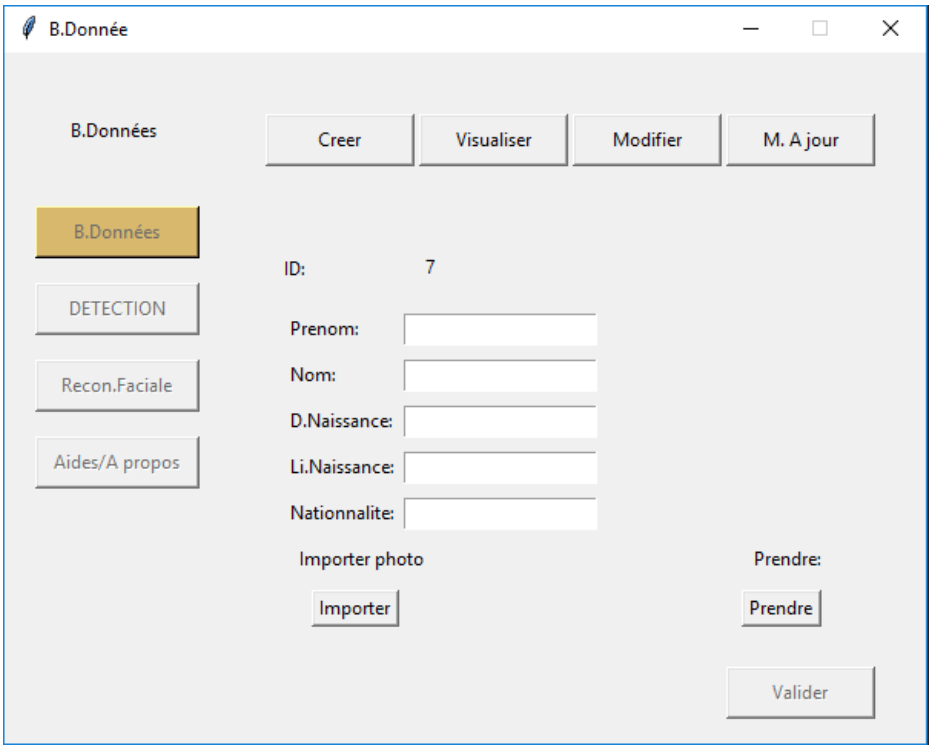

Figure 19 : Fenêtre créer

IV.2.1.2.(b) Le bouton 'Visualiser'

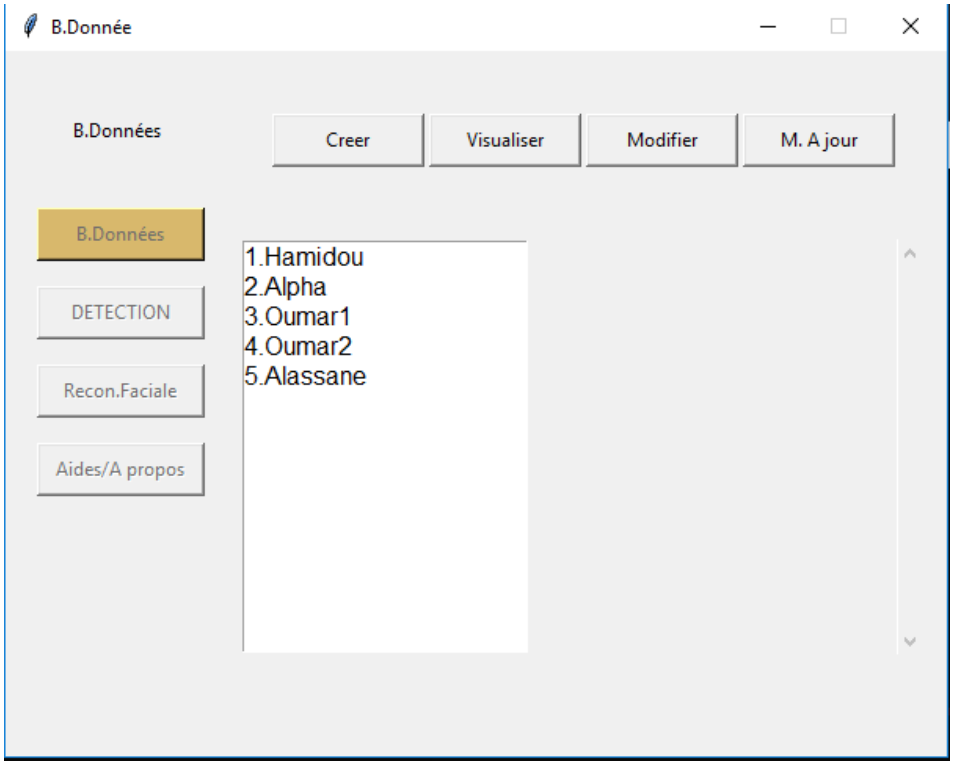

Figure 20 : Visualisation des identifiants et pseudo

Ce bouton permet de lire dans une liste tous les pseudos et identifiants enregistrées dans la bases de données. Ceux-ci sont le numéro matricule de la personne en quelques sorte, car pour obtenir les informations tel que le nom et prénom complet, la date de naissance, la nationalité etc. il faut ouvrir dans le dossier « data » le fichier (image) correspondant (le même nom de code) dans la partie reconnaissance faciale (voir figure 20 ci-dessus).

IV.2.1.2.(c) Le bouton 'Modifier'

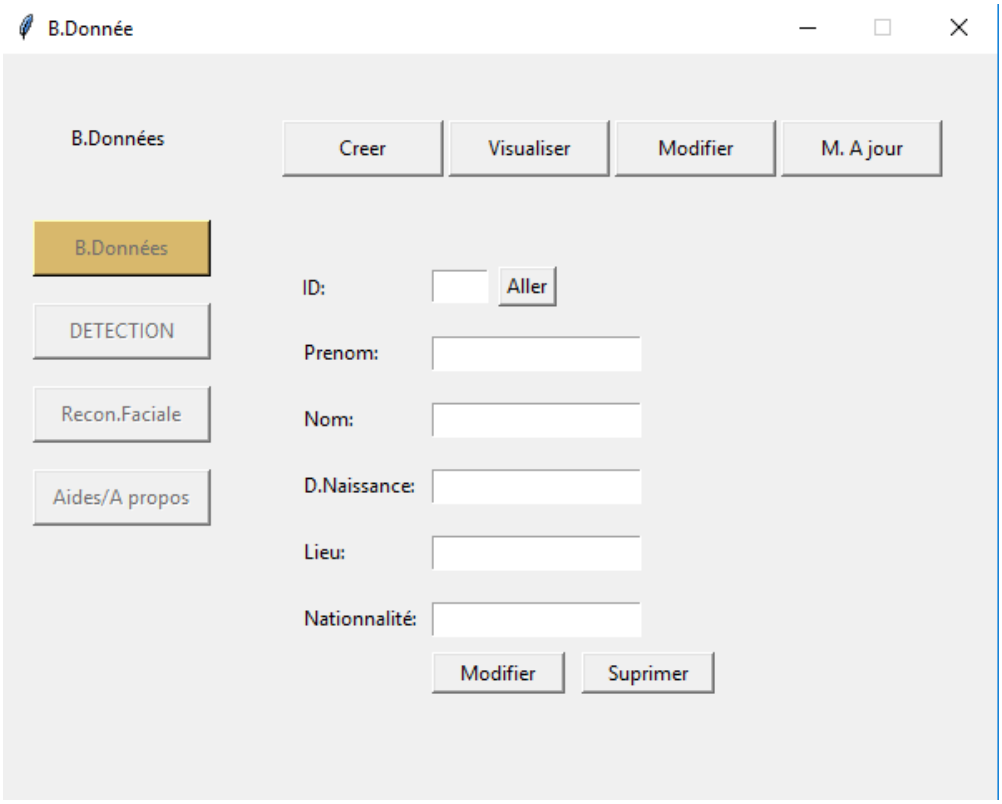

Figure 21 : Modification de la base de donnée

Celui-ci, quant à lui permet de redéfinir les informations personnelles d'un individu après avoir visualisé son nom de code. Pour mener cette opération, il faut d'abord saisir l'identifiant de la personne (un chiffre) puis sélectionner « aller » (voir figure 21 ci-dessus). La personne enregistrée sera affichée, sinon à la place des cases d'information il sera écrit « ERROR » si elle n'existe pas dans la base de données (voir figure 22). Si les informations personnelles de l'individus apparaissent, il suffit de les effacer et de les redéfinir, puis valider l'opération en cliquant sur le bouton « Modifier » (figure 23). Il est aussi possible de supprimer une personne de la base de données en cliquant sur le bouton « supprimer ».

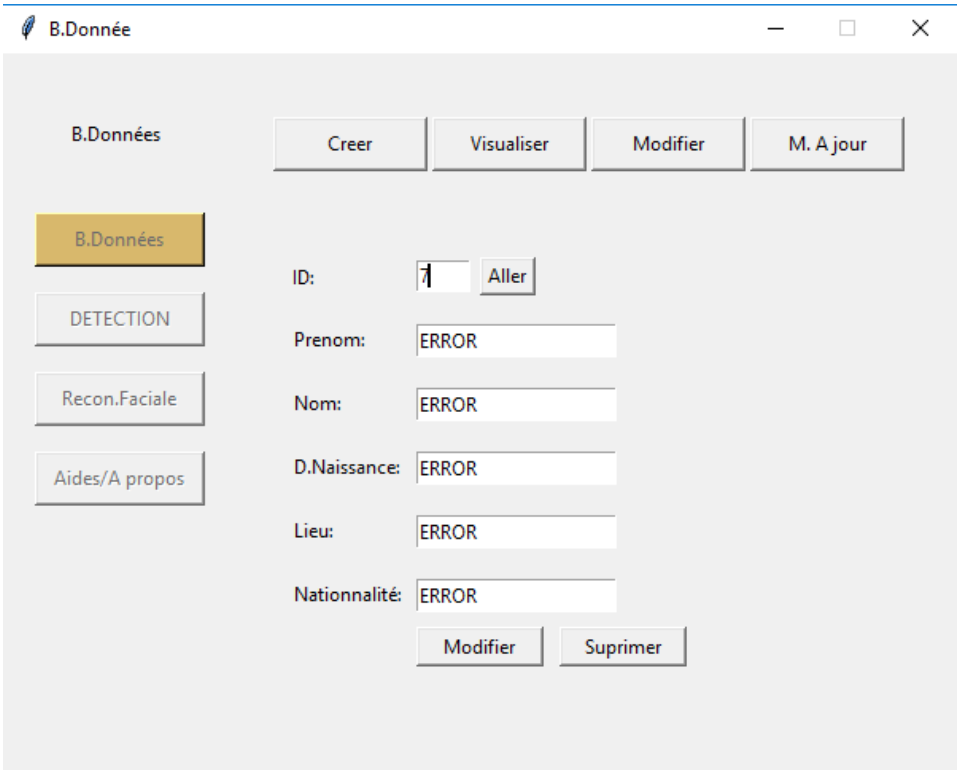

Figure 22 : Identifiant erroné

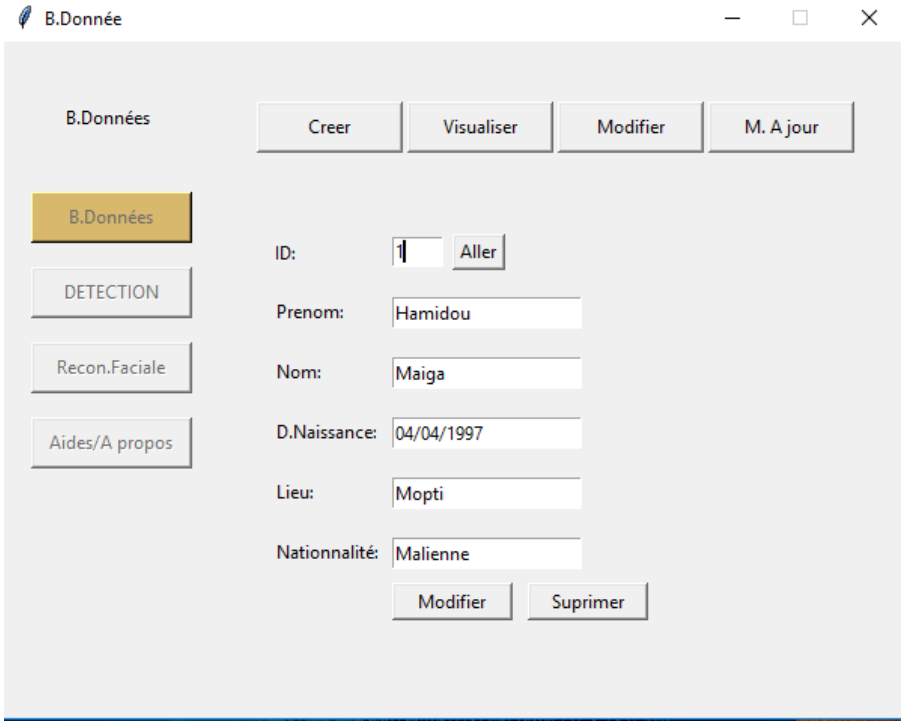

Figure 23 : Identifiant correct

IV.2.1.2.(d) Mettre à jour

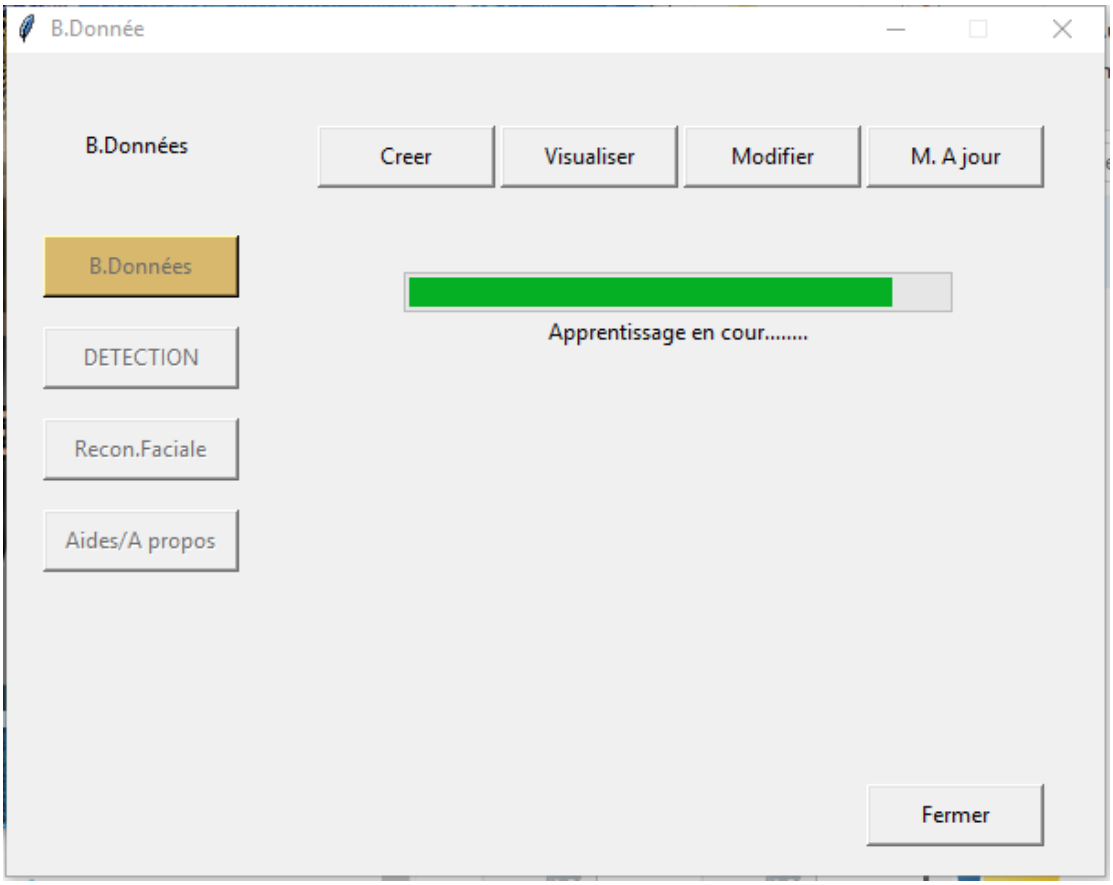

Figure 24 : Mise à jour

Le logiciel se met automatiquement à jour à chaque opération effectuée sur la base de données, néanmoins, il est possible de le faire manuellement en cliquant sur le bouton « M. A JOUR » permettant de rafraîchir et d'enregistrer les modifications apportées (voir figure 24 ci-dessus).

IV.2.1.3. La fenêtre 'Détection'

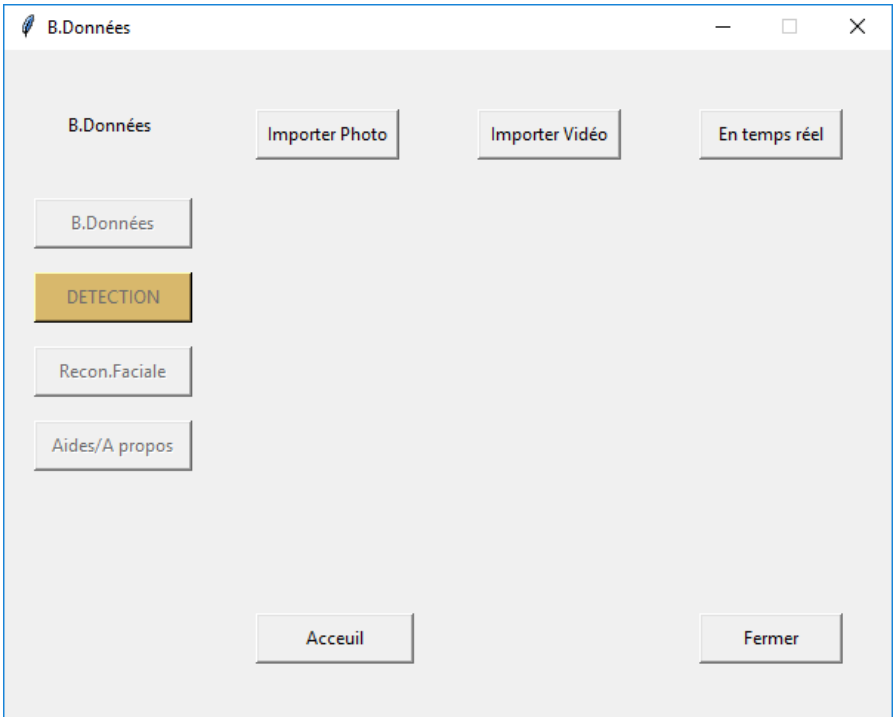

Figure 25 : Détection d'un visage humain

Son principal but est de détecter dans une image ou une vidéo importée un ou plusieurs visages. Il est aussi possible de le faire en temps réel grâce à une caméra (voir figure 25 ci-dessus).

IV.2.1.3.(a) Détection sur une photo

Après avoir cliqué sur le bouton « importer une photo », une fenêtre de sélection s'ouvre, nous permettant ainsi de sélectionner la photo à importer (une seule à la fois) (voir figure 26).

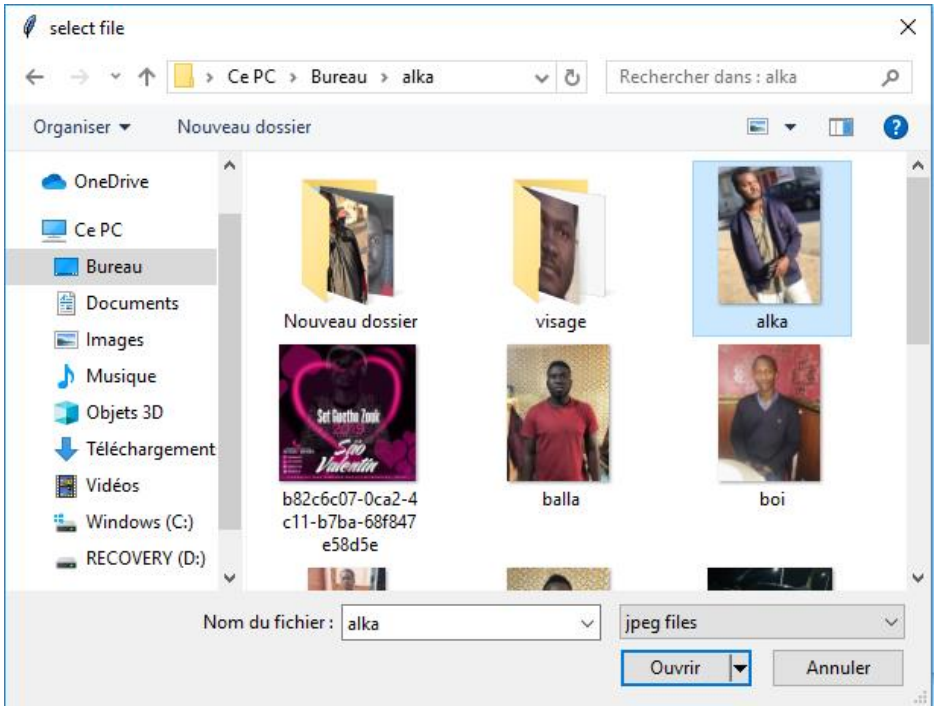

Figure 26 : Sélection de la photo sur laquelle le logiciel détectera les visages humains

Une fois la photo importée, le logiciel va encadrer tous les visages par un rectangle bleu (voir figure 27).

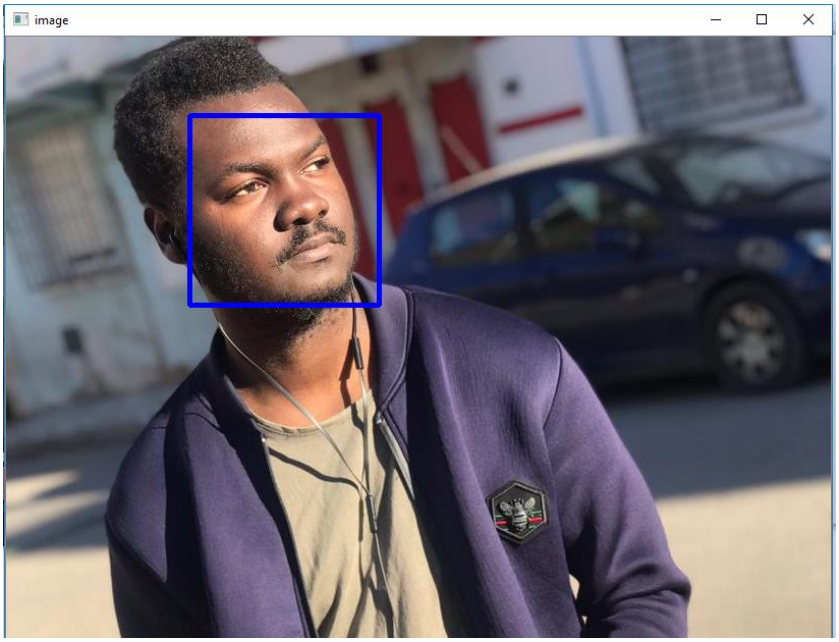

Figure 27 : Visage détecté

IV.2.1.3.(b) Détection sur une vidéo

Pour le cas de la vidéo, c'est le même principe sauf qu'il faudra sélectionner une vidéo au lieu d'une photo. Au cours de la lecture de cette dernière le logiciel va encadrer tous les visages qui apparaissent.

IV.2.1.3.(c) Détection en temps réel

Avec une webcam filmant en temps réel, le logiciel détecte lui aussi tous les visages apparaissant à l'écran en temps réel (voir figure 28).

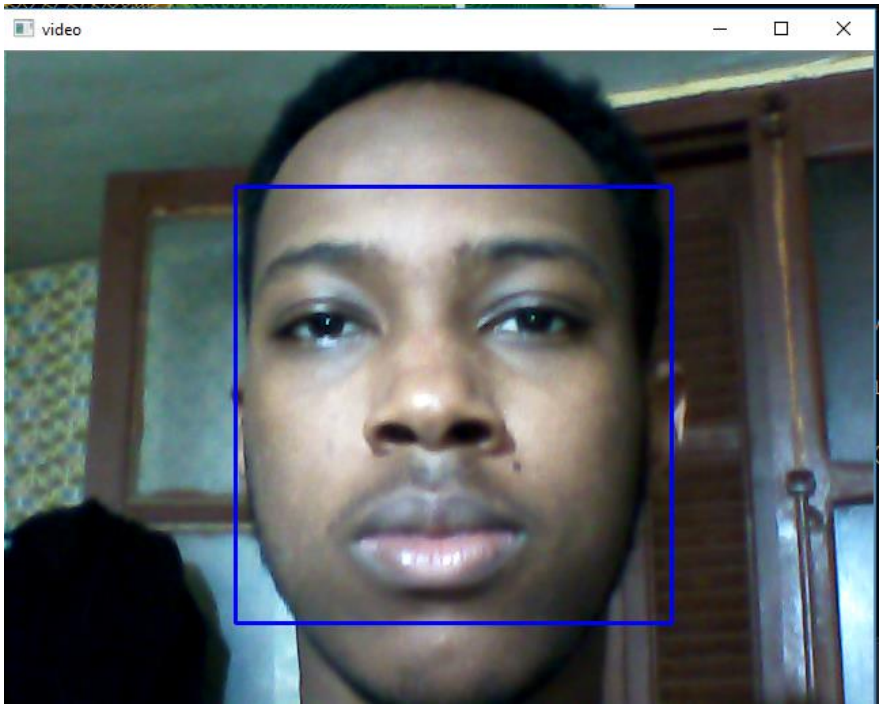

Figure 28 : Détection de visage en temps réel

IV.2.1.4. Reconnaissance

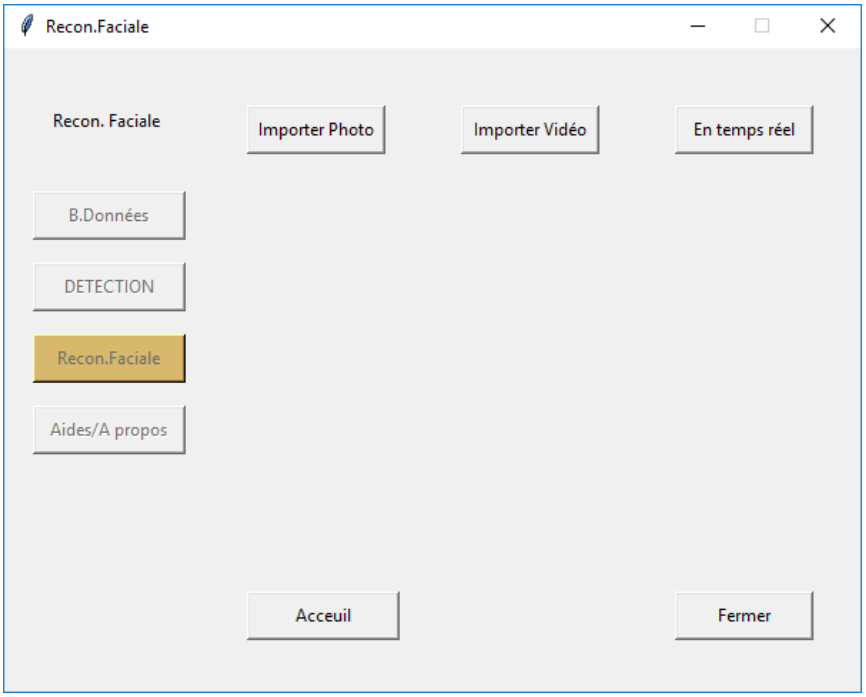

Figure 29 : Fenêtre de la reconnaissance faciale

Il s'agit du même principe que la détection, sauf que cette fois, le logiciel va reconnaitre l'individus ou les individus sur la photo ou la vidéo, mais cela peut aussi bien se faire en temps réel (voir figure 29 ci-dessus).

IV.2.1.4.(a) Détection sur une photo

Après avoir importé la photo dans le logiciel, il donne automatique sur l'image le résultat du traitement (voir figure 30 et 31).

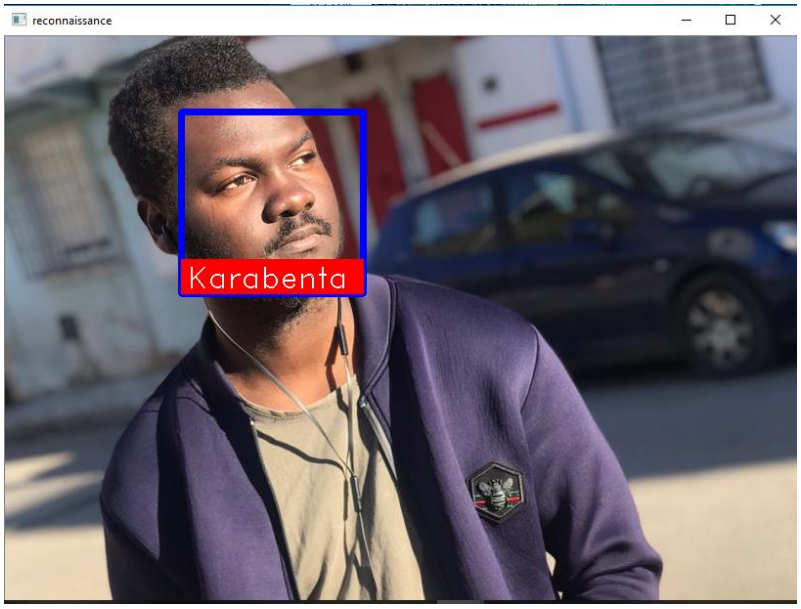

Figure 30 : Reconnaissance sur une photo

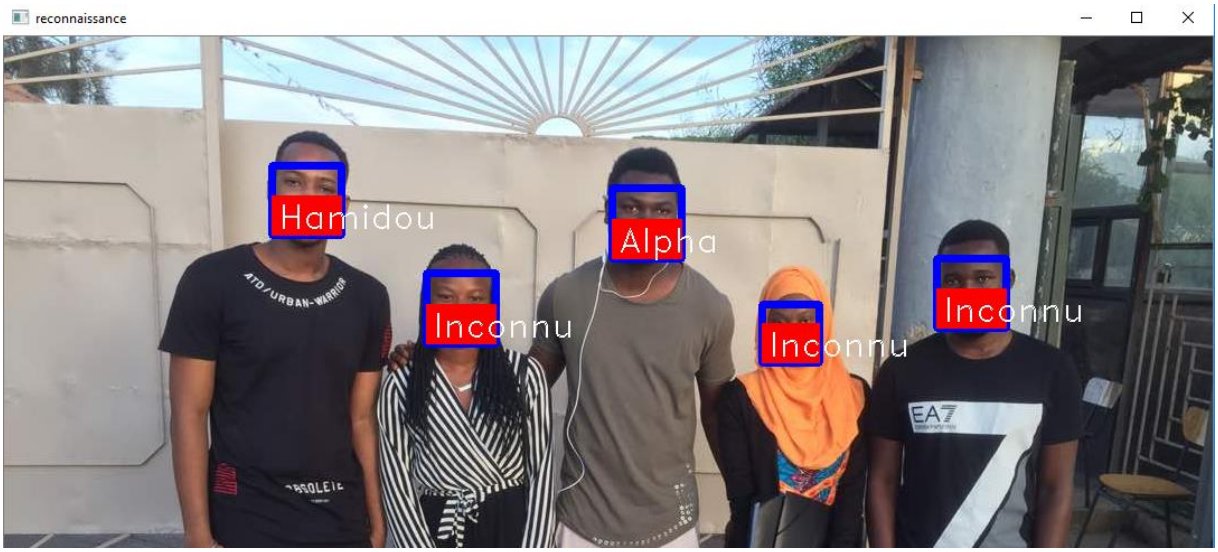

Figure 31 : Reconnaissance multiple sur une photo

IV.2.1.4.(b) Détection sur une vidéo

Le même principe pour la vidéo (voir figure 32).

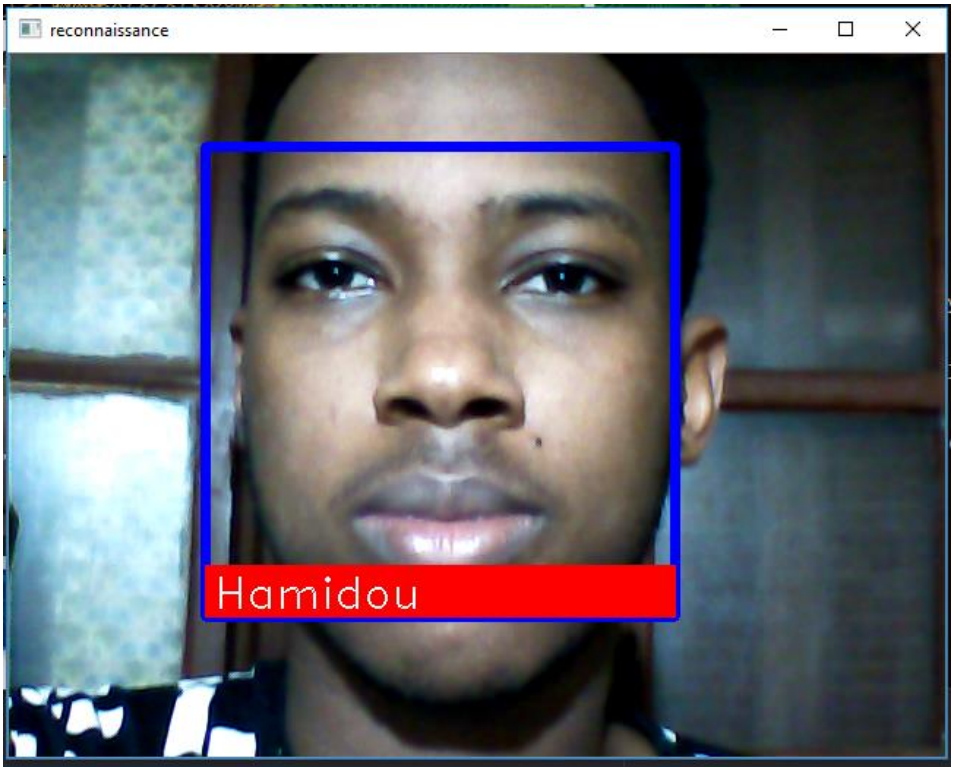

Figure 32 : Reconnaissance sur une vidéo importée

IV.2.1.4.(c) Détection en temps réel

Il en est de même pour la reconnaissance en temps réel (voir figure 33).

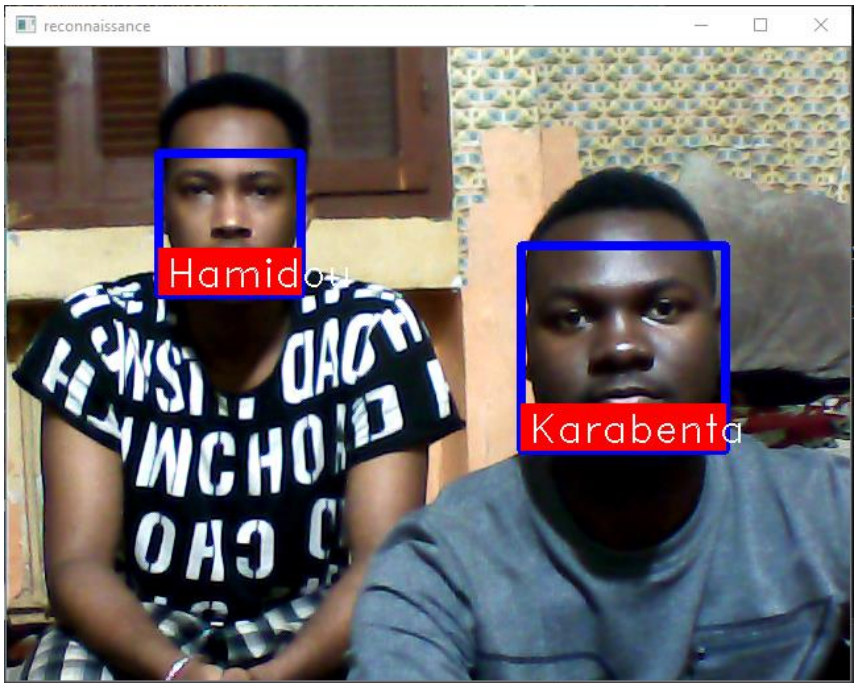

Figure 33 : Reconnaissance en temps réel

IV.2.1.5. A propos

Cette option apporte des informations sur la performance du système, sur sa version ainsi que sur ses concepteurs. Il affiche : « FARES_MAK (Face Recognition System of Maïga And Karabenta) est un logiciel de reconnaissance faciale créer par les étudiants Maïga Hamidou et Karabenta Alpha Aboubacar Sidiki au terme de leur étude universitaire cycle master… Date de dernière mise à jour : 30/06/209. »

IV.2.2. Performance du logiciel

Ce système a un taux de reconnaissance très élevé étant donnée la méthode utilisée (l'Analyse en Composante Principale très sensible à la lumière, la pose, les bruits, les expressions faciales …), il est capable de fonctionner normalement même avec une seule image en acquisition. Nous comptons à ce jour 28 tests corrects sur un total de 30 tests effectués.

IV.2.2.1. Calcul du taux d'erreur :

Pour chaque personne, les points caractéristiques du visage sont répartis différemment., et selon le temps, l'environnement et d'autres facteurs, le visage peut un peu changer, sachant cela nous choisissons un seuil ou plus explicitement une distance limite au-delà duquel le point à comparer au point du modèle établi sera qualifié à une position insuffisante pour dire que les deux sont à une même position sur le visage présenté et le visage enregistré dans la base de données. Etant donné que la position des yeux ne change pas avec le temps, il y a donc pas besoin de continuer la comparaison si ces derniers ne sont pas à la même position. Pour obtenir une très petite marge d'erreur, il faut donc un taux de probabilité très faible (entre 0 et 1% maximum).

Pour un total de 22 points, nous estimons que si 21 points sur 22 ne sont pas validés lors de la comparaison des points (supérieur au seuil de la distance euclidienne), le visage est rejeté.

Soit e l'erreur tel que e = 1- 21/22 = 1/22 =0.045. Le taux d'erreur E = e*100 = 4.5%

La fiabilité du système est donc de 100% - 4.2% = 95.5%

IV.3. Conclusion

Le logiciel implémenté du nom de FARES_MAK a une fiabilité assez élevée : 95.8%. Ce qui veut dire que seulement 4 sur 100 résultats donnés par le système peut être erroné de point de vue mathématique, cette erreur est certes très faible mais non négligeable. Pour déterminer la performance du système, nous avons effectué des tests sur 30 sujets et avons obtenu le résultat suivant : 28 test corrects sur 30, la performance est donc de :

$$P = \frac{28}{30} = 0.9333 = 93.33\%$$

Il est néanmoins possible d'améliorer ce résultat en utilisant par exemple la SVM pour amoindrir les difficultés de l'ACP fasse à la pose, l'éclairement et d'autres paramètre affectant la reconnaissance.

V. Conclusion générale

V.1. Résumé

La reconnaissance faciale est la technique biométrique la plus utilisée au monde. Elle passe en générale par deux principaux phases de traitement : l'apprentissage et la reconnaissance.

L'apprentissage machine consiste :

- Dans un premier temps à créer une base de données avec lesquelles l'ordinateur sera entrainé à reconnaitre un individu en particulier parmi tant d'autre sur une image statique ou dynamique,
- Enfin à extraire les caractéristiques pour une classification.

La reconnaissance est l'opération qui consiste à vérifier ou proclamer l'identité d'un individu.

Plusieurs méthodes peuvent être utilisées pour entrainer un ordinateur à reconnaitre une personne, elles se regroupent en trois grands catégories :

- Les méthodes globales sensibles : à la pose, le bruit, la lumière, les expressions faciales… (tel que l'analyse en composante principale mais qui est néanmoins capable de reconnaitre une personne même avec une seule image d'apprentissage),
- Les méthodes locales beaucoup plus fluide mais relativement complexe et
- Et les méthodes hybrides, une combinaison de méthode globale et locale. Le modèle de Markov et les réseaux neurones peuvent êtes cités bien qu'ils soient des méthodes n'entrant dans aucun de ces catégories.

Au terme de nos recherches, le logiciel que nous avons implémenté du nom de FARES_MAK permet d'authentifier une personne avec une performance très élévée. Pour témoigner de sa grande robustesse, nous nous sommes permis de prendre une seule image en acquisition (de l'individu à authentifier) dans notre base de données et de procéder aux tests suivants : changement de pose, expressions faciale, variation de lumière, camera de différentes résolutions, le port de lunette et de casquette pour voir à quel point sa capacité de reconnaissance serait affectée. Nous avons constaté que la résolution de la caméra lors de la reconnaissance ainsi que la qualité de l'image d'acquisition influençaient beaucoup plus le taux de

reconnaissance que la variation de pose, de lumière, le port de lunette ou de caquette. Cela est dû au fait que nous avons utilisé une technique multi algorithme (en traitant la même image par plusieurs programmes lors du prétraitement : conversion en niveau de gris, égalisations de l'histogramme, même axe pour les yeux, même résolution et dimension pour les images acquises et à comparer…).

V.2. Perspectives

Nous vous présentons dans cette partie le projet dans son intégralité. Tout d'abords la description du système qui sera mis en place :

Un système d'authentification destiné à sécuriser un établissement sensible, par exemple :

- Les salles à accès restreints : Une chambre munie d'un détecteur de présence est aménagée pour permettre à une seule personne de faire le test d'authentification faciale (au-delà de ce nombre le système se verrouille automatiquement). Lors du test la chambre est complètement verrouillée, un bouton permet d'ouvrir la porte et de revenir sur ces pas en cas d'échec de l'authentification, mais en cas de réussite, l'accès est autorisé à la personne.
- Le signalement automatique d'un personnel non autorisé ainsi que des intrus : grâce aux caméras placées à des points stratégiques, tout intrus détectée serra signaler aux personnelles de sécurité grâce à l'application FARES_MAK avec comme contenu du message d'alerte : la photo du présumé intrus, la date, l'heure et l'endroit de la prise, un temps de réaction de cinq minuté leur est donnée au-delà duquel une alarme se déclenche.
- La capacité de reconnaitre les armes à feu et les armes de destruction massive, de les localiser grâce à un drone ou un satellite et enfin faire un rapport détaillé.
- Pour finir, afin que notre système puisse accomplir toutes ces tâches, nous effectuerons des améliorations sur notre application afin de la rendre plus interactive, puis nous en feront une application web pour permettre les contrôles à distances et bien d'autres fonctionnalités intéressent qu'offre les technologies du moment.

VI. Bibliographie

[1] Internaute. (s.d.). *http://www.linternaute.com/science/biologie/dossiers/06/0607-biometrie/visage.shtml*.

[2] actualitéhitech.com. (s.d.). *https://actualitehightech.com/2018/07/27/histoire-de-la-biometrie-regard-sur-les-technologies-biometriques-du-passe-au-present/*.

[3] Wikipedia.org. (s.d.). *https://fr.wikipedia.org/wiki/M%C3%A9thode_de_Viola_et_Jones*.

[4] Wikihow.com. (s.d.). *https://fr.wikihow.com/calculer-la-covariance*.

[5] Doanh, P. V. (2010). *RECONNAISSANCE DE VISAGES EN UTILISANT LE DESCRIPTEUR POEM (Patterns of Oriented Edge Magnitudes).* Institute de la francophonie pour l'informatique.

[6] « Mise au point d'une application de reconnaissance faciale ». Préparer par : Khefif Bouchra, Novembre 2013 Université Abou Bakr Belkaid – Tlemcen Département d'Informatique Option : Réseaux et systèmes distribués (R.S.D.), page 18-23.

[7] S. Mika, G. Ratsch, J.Weston, B. Schölkopf, and K.-R. Müller. "Fisher Discriminant Analysis With Kernels". In : *Neural Networks for Signal Processing IX*, pp. 41–48, 1999

[8] Laurent Zwald, These de l'universite Paris XI UFR Sientifique d'Orsay. *Perfermence statistique d'apprentissage Kernel projection machine,2006*.

[**9]** P. Sinha, B. B. (2006). *Face Recognition by Humans.*

Annexe 1 : Installation de python et des différentes bibliothèques

Python 3.7 :

Commençons par une bonne nouvelle : sous Ubuntu, Python est déjà préinstallé. Pour Windows et Mac OS X, la première étape consiste à installer l'implémentation officielle *CPython*. Mac OS X est livré en standard avec Python. Malheureusement, celui-ci n'est mis à jour qu'à chaque sortie d'une nouvelle version du système, soit environ tous les deux ans. On se retrouve souvent avec une version de Python largement dépassée. Il est donc indispensable d'installer la version qui nous intéresse à côté de celle existante et qui cohabitera en très bon voisinage avec cette dernière.

Une fois le paquet téléchargé sur le site de python, il suffit de l'exécuter et de suivre les étapes de l'assistant d'installation. À la première étape, l'assistant vous demande si vous désirez installer Python pour tous les utilisateurs de l'ordinateur ou juste pour l'utilisateur courant ; il est recommandé de choisir *Install for all users*. L'assistant demande ensuite de choisir un emplacement pour l'installation. Afin de respecter les standards Windows, nous recommandons d'installer Python dans Program Files et non à la racine du disque système, comme proposé par défaut par l'installeur.

Python –m pip install –upgrade pip (pour la mise à jour)

Anaconda :

Anaconda est un IDLE de python accessible via le site wwwcontinuum.io, après l'avoir téléchargé, il est possible de le metre à jour :

Conda update conda

Conda list

Conda search mpi4py

Conda install mpi4py

Open cv :

python -m pip install opencv-python

python -m pip install opencv-contrib-python

Numpy :

python -m pip install numpy

DLIB :

python -m pip install dlib

MATPLOTLIB :

python -m pip install matplotlib

PIL :

python -m pip install pil

Annexe 2 : Quelques sous-programmes utilisés

Sous-programme de détection du visage

```python
import cv2    # importer la bibliothèque openCv
dossier_visage= "Cascades\\haarcascade_frontalface_default.xml"    #  direction du classifieur de detection du visage(fichier.xml)
dossier_yeux="Cascades\\haarcascade_eye.xml"    #  direction du classifieur de detection des yeux(fichier.xml)

classifieur_visage= cv2.CascadeClassifier(dossier_visage)   #  chargement du classifieur de visage
classifieur_yeux= cv2.CascadeClassifier(dossier_yeux)   #  chargement du classifieur des yeux

cam=cv2.VideoCapture(0)   #  Activation de la caméra
if cam.isOpened():        # test si la webcam a été bien démarer
    ret,video=cam.read()

while ret:

    ret,video=cam.read()     #  lecture de la vidéo recu par la webcam
    gray=cv2.cvtColor(video,cv2.COLOR_BGR2GRAY)  #  converssion en niveau de gris
    visages=classifieur_visage.detectMultiScale(gray,1.3,5)  #  detection du visage

    for (x,y,w,h) in visages:

        image_visage = image[y:y+h,x:x+w]   #  stocker la region qui contient le visage
        yeux=classifieur_yeux.detectMultiScale(image)   #  detection du visage
        for (ex,ey,ew,eh) in yeux:
            cv2.rectangle(video,(x,y),(x+w,y+h),(255,0,0),2)       #  dessiner un rectangle autour du/des visages detecter

    cv2.imshow("video",video) #  afficher la video sur une nouvelle fenetre de visualisation de titre video
    if cv2.waitKey(1)== 27:   #  sortir de la boucle dès appui sur la touche esc(code = 27)
        break
cam.release()    # eteindre la webcam
cv2.destroyAllWindows()    # fermer toute les fenetres de visualisation
```

Sous-programme de la création de base de données : (Acquisition de l'acquisition à la classification)

```python
import cv2    # importer la bibliothèque openCv
dossier_visage= "Cascades\\haarcascade_frontalface_default.xml"    #  direction du classifieur de detection du visage(fichier.xml)
dossier_yeux="Cascades\\haarcascade_eye.xml"    #  direction du classifieur de detection des yeux(fichier.xml)

classifieur_visage= cv2.CascadeClassifier(dossier_visage)   #  chargement du classifieur de visage
classifieur_yeux= cv2.CascadeClassifier(dossier_yeux)   #  chargement du classifieur des yeux

cam=cv2.VideoCapture(0)   #  Activation de la caméra
if cam.isOpened():        # test si la webcam a été bien démarer
    ret,video=cam.read()

while ret:

    ret,video=cam.read()     #  lecture de la vidéo recu par la webcam
    gray=cv2.cvtColor(video,cv2.COLOR_BGR2GRAY)  #  converssion en niveau de gris
    visages=classifieur_visage.detectMultiScale(gray,1.3,5)  #  detection du visage

    for (x,y,w,h) in visages:

        image_visage = image[y:y+h,x:x+w]   #  stocker la region qui contient le visage
        yeux=classifieur_yeux.detectMultiScale(image)   #  detection du visage
        for (ex,ey,ew,eh) in yeux:
            cv2.rectangle(video,(x,y),(x+w,y+h),(255,0,0),2)       #  dessiner un rectangle autour du/des visages detecter

    cv2.imshow("video",video) #  afficher la video sur une nouvelle fenetre de visualisation de titre video
    if cv2.waitKey(1)== 27:   #  sortir de la boucle dès appui sur la touche esc(code = 27)
        break
cam.release()    # eteindre la webcam
cv2.destroyAllWindows()    # fermer toute les fenetres de visualisation
```

```python
import cv2    # importer la bibliothèque openCv
import math   # importer bibliothèque math pour les calculs mathematiques


compteur=0  # initialisation du nombre de pose prise
#--------------------------------Charger les classificateurs-----------------------------------------------------------

dossier_visages= "Cascades\\haarcascade_frontalface_default.xml"   #  direction du classificateur (fichier.xml)
dossier_yeux="Cascades\\haarcascade_eye.xml"
visage_cascade=cv2.CascadeClassifier(dossier_visages)  #  chargement du classificateur visage
yeux_cascade=cv2.CascadeClassifier(dossier_yeux)       #  chargement du classificateur des yeux

#-------------------------------------Fonction d'enregistrement du nom/Id dans fichier texte----------------------

def enrgNom(id,nom,prenom,da_Naissance,li_Nais,nationnalite):
    id=int(id)   # transformer id en entier
    if id==1:    # test si c'est le premier id a enregistrer

        fichier = open("Noms.txt","w")  # ouvrir le fichier texte en mode ecriture (w=write)
        texte=str(id)+"."+prenom+"."+nom+"."+da_Naissance+"."+li_Nais+"."+nationnalite+"\n"  # construire les informations a enregistrer
        fichier.write(texte)  # enregistrer l'information dans le fichier texte
        fichier.close()  # fermer le fichier texte
    else:
        fichier= open("Noms.txt","a")  # ouvrir le fichier texte en mode ajout(a= append)
        texte=str(id)+"."+prenom+"."+nom+"."+da_Naissance+"."+li_Nais+"."+nationnalite+"\n"  # construire les informations a enregistrer
        fichier.write(texte)  # enregistrer l'information dans le fichier texte
        fichier.close()   # fermer le fichier texte
```

```python
def pretraitement(Image):  # Fonction de pretraitement
    Theta = 0
    rows, cols = Image.shape  # recuperer la dimenssion de l'image
    glass = yeux_cascade.detectMultiScale(Image)    # detecter les yeux
    for (sx, sy, sw, sh) in glass:
        if glass.shape[0] == 2:    # verifier si on a 2 yeux visibles
            if glass[1][0] > glass[0][0]:
                DY = ((glass[1][1] + glass[1][3] / 2) - (glass[0][1] + glass[0][3] / 2))  # calcul de la diference entre les hauteurs
                DX = ((glass[1][0] + glass[1][2] / 2) - glass[0][0] + (glass[0][2] / 2))  # calcul de la difference entre les largeurs
            else:
                DY = (-(glass[1][1] + glass[1][3] / 2) + (glass[0][1] + glass[0][3] / 2))  # calcul de la diference entre les hauteurs
                DX = (-(glass[1][0] + glass[1][2] / 2) + glass[0][0] + (glass[0][2] / 2))  # calcul de la diference entre les largeurs

            if (DX != 0.0) and (DY != 0.0):  # Test si la differnce entre les yeux sont differents
                Theta = math.degrees(math.atan(round(float(DY) / float(DX), 2)))   # chercher l'angle


                M = cv2.getRotationMatrix2D((cols / 2, rows / 2), Theta, 1)  # chercher la matrice de rotation
                Image = cv2.warpAffine(Image, M, (cols, rows))    # roter l'image pour que les yeux soit sur le meme axe


    Image=cv2.equalizeHist(Image)    #  Equalisation de l'histogram


    return Image    # retourner l'image pretraiter
```

```
55
56  print("**********Creation de la bank d'image des individus**********")
57
58  id = input('\n Enter id ')    # l'identifiant
59  nombreDePrise =input("Combien de photo voulez-vous prendre par personne ?")      # nombre de prise
60  nombreDePrise= int(nombreDePrise)  # trenformation du nombre de prise en entier
61
62  #  recuperation des informations de la personne
63  nom=input("\n Entrer le nom de la personne")
64  prenom=input("\n Entrer le prenom de la personne")
65  da_Naissance=input("\n Entrer la date de naissance de la personne")
66  li_Nais=input("\n Entrer le lieu de naissance de la personne")
67  nationnalite=input("\n Entrer la nationnalité de la personne")
68
69  fonction.enrgNom(id,nom,prenom,da_Naissance,li_Nais) #   enregistrer les informations dans le fichier texte
70  print("\n [INFO] INITIALISATION DE LA CAMERA \ TENEZ VOUS FACE A LA CAMERA.....")
71
72  cam=cv2.VideoCapture(0)   #  Activation de la caméra
73
74  if cam.isOpened():        # test si la webcam a été bien démarer
75      ret,video=cam.read()
76
77  while ret:
78
79      ret,video=cam.read()       #  lecture de la vidéo recu par la webcam
80      gray=cv2.cvtColor(video,cv2.COLOR_BGR2GRAY)   #  converssion en niveau de gris
81      visages=visage_cascade.detectMultiScale(gray,1.3,5)   #  detection du visage
```

```
84      for (x,y,w,h) in visages:
85
86
87          image=pretraitement(gray)  #  Appelle de la fonction de pretraitement
88          image_visage = image[y:y+h,x:x+w]    #  stocker la partie du visage
89          yeux=yeux_cascade.detectMultiScale(image)  #  detection du visage
90          for (ex,ey,ew,eh) in yeux:
91              fonction.dessin_box(video,x,y,w,h)   #  dessiner un rectangle au niveau des visages
92
93          if cv2.waitKey(1)==32:
94
95              image=cv2.resize(image_visage,(100,100))  #  mettre l' image a la dimenssion (100,100)
96              cv2.imwrite("data//User." + str(id) + '.' + str(compteur) + ".jpg", image)  #  stocker l'image dans la base de donnée d'image
97              print("compteur = {}".format(compteur))
98              compteur+=1   #  incrementer le nombre de prise
99              directive(compteur)
100         cv2.imshow("visage",image_visage)  #  afficher le visage detcter sur la fenetre
101     cv2.imshow("video",video)   #  afficher la video sur la fenetre video
102     if cv2.waitKey(1)== 27:     #  sortir de la boucle dès appui sur la touche esc(code = 27)
103         break
104     elif compteur >= nombreDePrise:
105         break
106
107 print("Fin de la prise.....")
108 cam.release()     # fermetture de la webcam
109 cv2.destroyAllWindows()   # fermer toute les fenetres de visualisation
```