الجـــــــــــــــــــــمـهـوريـــــة الجـــــــــــــزائـريـة الديمـــــــقـراطيـة

الـشـــــــــعبيـة

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التـــــــــــــــــــــــــعليـم العـــــــالي و البـــــــــحث العـــــلـمـــي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جـــــــــــــامعة أبي بكـــر بـلـقـايد – تـلمســـــــــــــان –

Université Aboubakr Belkaïd – Tlemcen –

Faculté de TECHNOLOGIE

MEMOIRE

Présenté pour l'obtention du **diplôme** de **MASTER**

En : Télécommunications

Spécialité : Réseaux et Télécommunications

Par : Diakité Oumar Moro et Sidibé Fatouma Chirfi

Sujet

Utilisation des techniques de reconnaissance pour sécuriser l'accès aux données et aux édifices

Soutenu en Septembre 2020, devant le jury composé de :

Mr. IRID Mohamed Hadj	MCB	Univ. Tlemcen	Président
Mr. Bouacha Abdellhafid	MCA	Univ. Tlemcen	Examinateur
Mr. Réda Bouabdellah	MAA	Univ. Tlemcen	Encadrant

Dédicaces

A mon magnifique père Mr. Salifou Diakite, homme merveilleux qui m'a inculqué les valeurs importantes de la vie et qui m'a soutenu dans chaque seconde de ma vie.

A ma très chère et tendre mère Mme. Diakite Oumou Namogo

A ma grande sœur Diakite Awa

A mes chers frères et sœurs

A toute ma famille, mes amis

A mon pays le Mali

A ce beau pays, l'Algerie qui m'a accueilli les bras ouverts

A tous mes camarades et à toute la communauté malienne et étrangère en Algerie.

(Diakité Oumar Moro)

Je dédie tout d'abord ce travail à mon adorable famille mes très chers parents source de vie et d'Amour inconditionnel.

Mon exemple, ma fierté mon papa Sidibé Modibo Kane

Ma source de motivation, ma tendre et douce mère Diakité Assétou

Mon grand-père le Pr. Aly Guindo

Mes frères et sœurs : Sory ; Moussa ; Aboubacar ; Maimouna ; Halima ; Safiatou ; Aissata Boité

Mon binôme et ami Diakité Oumar Moro

Mon très cher Daou Boubacar

Tous mes amis et toutes les personnes qui ont été présent de près ou de loin pour me soutenir et m'épauler durant tout mon cursus scolaire et qui ont cru en moi.

(Sidibé Fatouma Chirfi)

Remerciements

Nous remercions en 1er lieu le seigneur Tout-Puissant de nous avoir donné la force et le courage de mener à bien ce projet de fin d'études, par cette même présente nous lui témoignons notre reconnaissance pour ses bienfaits quotidiens.

Nous exprimons toute notre gratitude à Mr. Réda Bouabdallah de nous avoir dirigé et guidé tout le long de ce travail. Sa disponibilité, ses conseils et remarques constructifs furent pour nous très précieux dans la réalisation de ce travail.

Tous nos remerciements au président du jury Mr. IRID Mohamed Hadj, professeur à l'Université de Tlemcen d'avoir accepté de présider ce jury.

Nos remerciements également à Mr. Bouacha Abdelhafid, MCA à l'Université de Tlemcen, pour son enseignement en Communication Numérique et d'avoir accepté d'examiner ce modeste travail.

Nous remercions nos parents, nos frères et sœurs pour leurs incommensurables aides, soutiens et patience.

A tous nos proches, amis, à l'ensemble de nos professeurs, nous disons, merci d'avoir contribué à notre éducation, à notre épanouissement grâce à vous nous sommes à même de relever les défis de la société et de notre secteur d'activité

Résumé

La sécurité informatique demeure d'une importance capitale, dans une société aux ères technologiques en perpétuelle évolution.

Ce projet de fin d'études vise à sécuriser l'accès aux données et aux édifices, à travers l'utilisation d'une modalité biométrique, particulièrement le visage. Dans ce cadre, nous réaliserons un programme (3 sous-programmes complémentaires) effectuant la reconnaissance faciale (identification du visage). Pour se faire nous explorons les différentes méthodes et approches ainsi que les enjeux et verrous de cette technologie d'actualité.

Mots clés : Biométrie, reconnaissance faciale, identification, sécurité, Lbp, fonction Haar

Abstract

Computer security remains of paramount importance in a society of ever-changing technological ages.

This graduation project aims to secure access to data and buildings, through the use of a biometric modality, particularly the face. In this context, we will carry out a program (3 complementary subroutines) performing facial recognition (face identification). To do so, we explore the different methods and approaches as well as the challenges and obstacles of this topical technology.

Key words : biometric, facial recognition, identification, security, Lbp, Haar function

نبذة مختصرة

يظل أمن الكمبيوتر ذا أهمية قصوى

في مجتمع من العصور التكنولوجية المتغيرة باستمرار.

يهدف مشروع التخرج هذا إلى تأمين الوصول إلى البيانات والمباني ، من خلال استخدام طريقة القياس الحيوي ، وخاصة الوجه. في هذا السياق ، سنقوم بتنفيذ برنامج (3 إجراءات فرعية تكميلية) لأداء التعرف على الوجه (تحديد الوجه). للقيام بذلك ، نستكشف الطرق والأساليب المختلفة بالإضافة إلى تحديات وعقبات هذه التكنولوجيا الموضعية.

الكلمات الدالة: القياسات الحيوية ، التعرف على الوجه ، تحديد الهوية ، الأمن ، Lbp وظيفة Haar

Tables des matières

Table des matières

Chapitre 1: Biométrie

Chapitre 2: Spécifications et algorithmes

Chapitre 3: Conception et implémentation

Liste des abréviations

ADN : Acide Désoxyribonucléique

BSD: Berkeley Software Distribution

EBGM: Elastic Bunch Graph Matching

EBGM-PCA: Elastic Bunch Graph Matching- Principal Component Analysis

EER : Equals Error Rate

Etc : Et cetera

FAR: False-Acceptante Rate

FRR: False Rejection Rate

GNU: GNU's Not Unix

HMM: Hidden Markov Models

ICA: Independent Component Analysis

ID: Identifiant Informatique

LBP: Local Binary Pattern

LDA: Lattent Dirichlet Allocation

LED: Light-Emitting Diode

LFA: Local Feature Analysis

LG-PCA: Log Gabor- Principal Component Analysis

MacOS: Mac Operating System

NumPy: Numérique python

OS: Operating System

PCA: Principal Component Analysis

PFE : Projet de fin d'etudes

PIL : Python Imaging Library

OPEN CV: Open Source Computer Vision

RNA : Réseaux de Neurones

SVM : Support Vector Machine

2D : Deux dimensions

3D : Trois dimensions

Liste des figures

Chapitre1

Chapitre 2

Chapitre 3

Introduction générale

La sécurité informatique est devenue un domaine de recherche d'une très grande importance, car elle remédie aux problèmes d'insécurité de notre vie quotidienne et cela dans beaucoup de secteurs. L'être humain a longtemps utilisé des moyens de vérification d'identité pour des raisons de sécurité parmi ceux-ci nous pouvons citer : la carte d'identité, le passeport, le mot de passe. Malheureusement ces éléments peuvent être falsifiés, oubliés, oubien volés. Ces limitations, failles ont donné naissance au développement d'un autre moyen de sécurité, qui est la biométrie.

Une modalité biométrique doit être :

- ✓ universelle (présente chez chaque individu)
- ✓ unique (permettre de différencier les individus)
- ✓ permanente
- ✓ enregistrable
- ✓ mesurable

La biométrie vient donc remplacer ou renforcer les anciens moyens de vérification d'identité, elle utilise donc des caractéristiques physiologiques, comme le visage, l'iris, l'empreinte ou des caractéristiques comportementales comme la voix, la signature.

Les systèmes de reconnaissance biométriques, connaissent un essor remarquable et sont fréquemment présents dans notre vie de tous les jours.

Parmi les modalités biométriques, dans ce pfe nous nous intéresserons spécialement aux caractéristiques physiologiques du visage, étant le trait biométrique le plus utilisé par les humains et le plus accepté (car elle est non intrusive).

Notre travail consistera à mettre en place un système de reconnaissance faciale capable de sécuriser l'accès à des données ou à des édifices. Nous découvrirons derrière cette technologie semblant imprenable, de multiples failles et défis, raison pour laquelle elle a engendré autant de travaux de recherches ces 50 dernières années.

Le premier chapitre est consacré à la biométrie, sa définition, ses caractéristiques, ses différentes modalités, et l'architecture d'un système biométrique. Nous explorons aussi les différentes approches (globales, locales, hybrides)

Le deuxième chapitre présente les techniques de détection et de reconnaissance ; nous parlerons de certaines techniques dont (Eigeinface, la fonction Haar et le LBP) ; le choix des techniques et la programmation utilisée.

Le troisième chapitre, il sera question de matériel, de logiciel, d'implémentation de notre système de reconnaissance ainsi que la présentation de nos tests et résultats obtenus.

En conclusion, nous récapitulerons les principales contributions de ce mémoire et enfin exposerons les perspectives envisagées.

Chapitre 1 : Biométrie

I.1. Introduction

L'apparition de l'ordinateur ainsi que sa capacité à traiter et stocker les données permirent la création de systèmes biométriques informatisés, qui depuis quelques années sont de plus en plus utilisés. L'être humain possède plusieurs caractéristiques uniques et immuables d'où la diversité des systèmes appliquant la biométrie. Les appareils électroniques sont donc en mesure, de les mesurer, de les collecter et de les comparer avec une base de données afin d'identifier instantanément une personne.

Nous pouvons citer entre autres quelques données biométriques:

- L'empreinte digitale
- La géométrie de la main
- Le Visage
- L'iris
- La rétine …etc.

Nous verrons dans ce chapitre les principales technologies biométriques, puis nous allons nous focaliser sur les systèmes de reconnaissance faciale, leurs avantages ainsi que les problèmes liés à leurs applications.

I.2. La biométrie

I.2.1. Définition et principe

En technologie, la biométrie désigne la technique qui permet d'associer à une identité une personne voulant procéder à une action, grâce à la reconnaissance automatique d'une ou de plusieurs caractéristiques physiques et comportementales de cette personne préalablement enregistrées (empreintes digitales, visage, voix, ...) [1].

Une autre définition de la biométrie est toutes caractéristiques physiques ou traits personnels automatiquement mesurables, robustes et distinctifs qui peuvent être utilisés pour identifier un individu ou pour vérifier l'identité prétendue d'un individu [2].

La biométrie repose sur la mesure des caractéristiques morphologiques uniques d'un individu. Cette technologie de pointe est devenue en quelques années un moyen très utilisé d'identification d'une personne. Elle vient remplacer ou renforcer les dispositifs à clés ou à badges pouvant présenter des failles en matière de sécurité car contrairement à ce que l'on sait

ou ce que l'on possède la biométrie est basée sur ce que l'on est. Elle permet ainsi d'éviter la duplication, le vol, l'oubli, ou la perte. Les systèmes biométriques peuvent fournir deux modes de fonctionnement :

- L'identification consiste à déterminer l'identité d'une personne. Il s'agit de saisir une donnée biométrique de cette personne, en prenant par exemple une photo de son visage, en enregistrant sa voix, ou en captant l'image de son empreinte digitale. Ces données sont ensuite comparées aux données biométriques de plusieurs autres personnes qui figurent dans une base de données. C'est une comparaison du type un contre plusieurs. Dans ce mode, on pose une question simple : « qui êtes-vous ? ». Elle peut être utilisée par exemple pour sécuriser l'accès à un bâtiment où seules les personnes enregistrées dans la base de données seront reconnues autorisées à accéder au bâtiment.
- L'authentification, appelée également vérification, est le processus qui consiste à comparer les données caractéristiques provenant d'une personne, au modèle de référence biométrique de cette dernière, afin de déterminer la ressemblance. Le modèle de référence est préalablement enregistré et stocké dans une base de données. On vérifie ici que la personne présentée est bien la personne qu'elle prétend être. Il s'agit là d'une comparaison un à un. Dans ce mode, on pose la question : « êtes-vous bien Monsieur ou Madame X ? ». Elle est généralement employée dans des applications de contrôle d'accès et de paiement par authentification. A l'heure actuelle, les systèmes les plus utilisés sont les scanners d'empreintes digitales. Avec l'apparition de ces dispositifs sur les smartphones, cette technologie est devenue omniprésente. De même, selon une étude menée par Spiceworks, le scanning d'empreintes digitales est le système d'authentification biométrique le plus courant en entreprise, 57% d'entreprises l'utilisent. De même la reconnaissance faciale est de plus en plus couramment utilisée. Les smartphones récents, à l'instar de l'iPhone X d'Apple, peuvent être déverrouillés en montrant son visage à la caméra. Ce sont ici les contours faciaux de l'individu qui sont analysés et comparés pour mesurer les patterns uniques de son visage. Cette technologie est aussi utilisée par les systèmes de vidéosurveillance modernes [3].

I.3. Les Techniques Biométriques

On distingue trois catégories de mesures biométriques : les mesures morphologiques, comportementales et biologiques.

I.3.1. Les mesures morphologiques (physiologiques)

La biométrie morphologique se base sur les traits physiques particuliers qui, pour toutes personnes, sont permanents et uniques (empreinte digitale, visage, etc.).

I.3.2. Les mesures comportementales

La biométrie comportementale se base sur l'analyse de comportements d'un individu (manière de marcher, dynamique de frappe au clavier, etc.).

I.3.3. Les mesures biologiques

La biométrie biologique se base sur l'analyse des données biologiques liées à l'individu (salive, ADN, etc.). Les différentes techniques utilisées font l'objet de recherches régulières, de développements et bien entendu, d'améliorations constantes. Toutefois, les différentes sortes de mesures n'ont pas le même niveau de fiabilité. On estime que les mesures physiologiques ont l'avantage d'être plus stables dans la vie d'un individu. Par exemple, elles ne subissent pas autant les effets du stress, contrairement à l'identification par mesure comportementale. La biométrie permet l'identification et l'authentification d'une personne à partir de données reconnaissables et vérifiables, qui lui sont propres et qui sont uniques [4].

I.4. Présentation de quelques technologies biométriques

I.4.1. L'empreinte digitale

L'empreinte digitale est le modèle de relief cutané des doigts. L'identification par cette caractéristique est la technique la plus anciennement utilisée. En fait, c'était toujours le choix biométrique évident pour les services de police depuis plus de 100 ans. C'est pour cela qu'elle est généralement mal acceptée par les utilisateurs en raison de l'alignement fort avec la criminologie. Il existe plusieurs types de système de capture d'empreinte digitale : optique, thermique, électro magnétique et ultrasons [5].

Figure 1: Empreinte digitale

I.4.2. Le visage

Le visage est le moyen le plus naturel pour identifier des personnes, ce qui explique pourquoi cette caractéristique est bien acceptée par les utilisateurs. L'être humain a une capacité naturelle à reconnaitre les visages et d'identifier les personnes en un coup d'œil, ce qui n'est point le cas de la machine. Il faut donc apprendre artificiellement la machine à reconnaitre le visage humain. Pour se faire, le système intelligent autonome de reconnaissance du visage tire son exemple de l'homme. L'homme tout au long de sa vie, voit de nombreux visages, et conserve naturellement en mémoire ces visages formant ainsi une sorte de base de données. L'identification des visages par ordinateur nécessite également une base de données de visages. Les systèmes actuels d'identification du visage disposent d'un module d'acquisition d'images avec une caméra. La 1ère étape est donc la détection de visages dans l'image acquise, la 2ème étape consiste à appliquer des prétraitements d'images sur l'image de visage détectée afin de faciliter le repérage de traits caractéristiques, la 3ème étape est réalisée par le module de reconnaissance qui à l'aide d'algorithmes extrait une signature du visage. Cette signature est par la suite comparée aux signatures préalablement enregistrées dans la base de données locales, ainsi la machine procède pour identifier un individu. Plusieurs recherches furent effectuées pendant 25 ans pour améliorer la performance de ce genre de système, cependant de nombreux problèmes se posent.

Figure 2: Visage humain

I.4.3. La géométrie de la main

Cette modalité consiste à analyser la forme de la main sa longueur, sa largeur, sa hauteur, la courbure des doigts etc. Cette technique est récente, simple et bien acceptée par les utilisateurs qui suivent des guides des capteurs (LEDs infrarouge, des appareils photo numériques) pour qu'ils puissent bien positionner leurs doigts, ce qui rend ainsi la détection / la segmentation plus aisée, cependant ce genre de système peut être trompé par de vrais jumeaux ou même par des personnes ayant des formes proches de la main [6].

Figure 3: Forme de la main

I.4.4. L'iris

L'iris est la région annulaire située entre la pupille et le blanc de l'œil. La biométrie par ce trait est la plus récente, et la plus fiable, selon les estimations de Daugmann.

La probabilité de trouver 2 iris suffisamment identiques est 1 sur 10^{72} environ. L'image de l'iris est capturée par une caméra standard contraignante (exemple la distance entre la caméra et l'iris ne dépasse pas un mètre), ce qui limite l'utilisation de cette modalité. [6]

Figure 4: Iris [6]

I.4.5. La rétine

La rétine est la couche sensorielle de l'œil qui permet la vision, cette zone est parcourue par des vaisseaux sanguins dont les positions sont inchangeables durant toute la vie de la personne. L'identification de la rétine n'est pas récente, elle remonte aux années 30. Cette technologie est mal acceptée par les utilisateurs à cause des contraintes de l'acquisition. [6]

Figure 5: Rétine [6]

I.4.6. La voix

La reconnaissance de la voix est une biométrie comportementale. Elle n'exige aucun contact physique avec le lecteur de système. En 1962 Lawrence Kersta a prouvé que la voix de chaque personne est unique et qu'il est possible de la présenter graphiquement. Il existe deux principales méthodes de traitement de ce trait biométrique, la première dépend du texte prononcé, et la deuxième (la plus difficile) est indépendante du texte. Bien que cette modalité ne nécessite pas de matériel cher (microphone par exemple), cependant le bruit ambiant et les propriétés acoustiques telles que la réflexivité et l'absorption, influence la vérification de la voix, réduisant ainsi son utilisation [6].

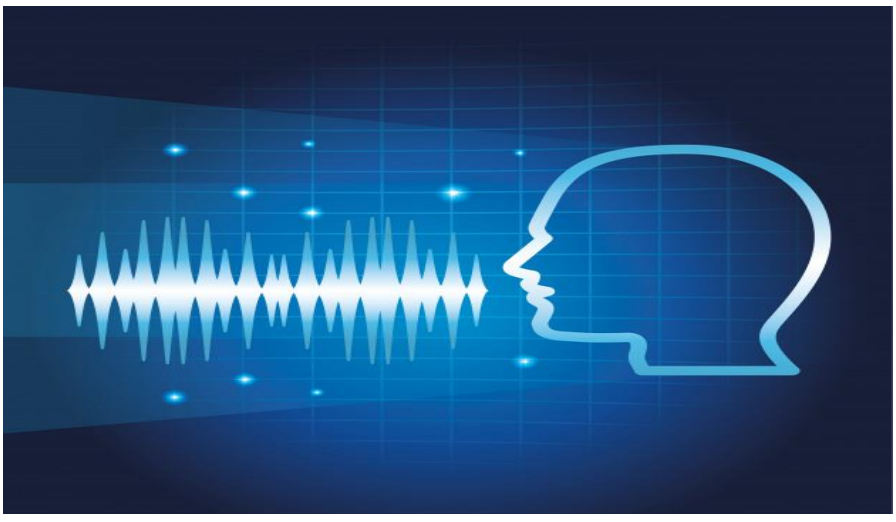

Figure 6: Voix

I.4.7. La signature

La vérification par la signature est l'une des premières méthodes utilisées dans le domaine de la biométrie. Les systèmes de reconnaissance de l'écriture analysent soit la géométrie de la signature (mode statique), soit ses caractéristiques spécifiques comme la vitesse, la pression sur le crayon, ce mode qui s'appelle le mode dynamique est le plus discriminant.

La capture se fait à l'aide d'une tablette graphique. Bien que la signature soit bien acceptée par les utilisateurs, sa variabilité (à cause de l'état de santé ou l'état émotionnel de l'individu) pose un grand problème.

Figure 7: Signature

I.4.8. La dynamique de frappe

C'est une autre technique primitive dans laquelle un énorme apport en temps et en effort a été investi, notamment par quelques grandes compagnies de technologie de l'information. L'idée d'identifier un individu par sa dynamique particulière de frappe était clairement attrayante [6].

Figure 8: Dynamique de frappe

I.5. Architecture d'un système biométrique

Un système biométrique comporte en général 3 modules, à travers ses 3 modules (apprentissage, reconnaissance, base de données) le système peut réaliser soit une authentification soit une identification. Il existe aussi un quatrième module facultatif qui est le module d'adaptation.

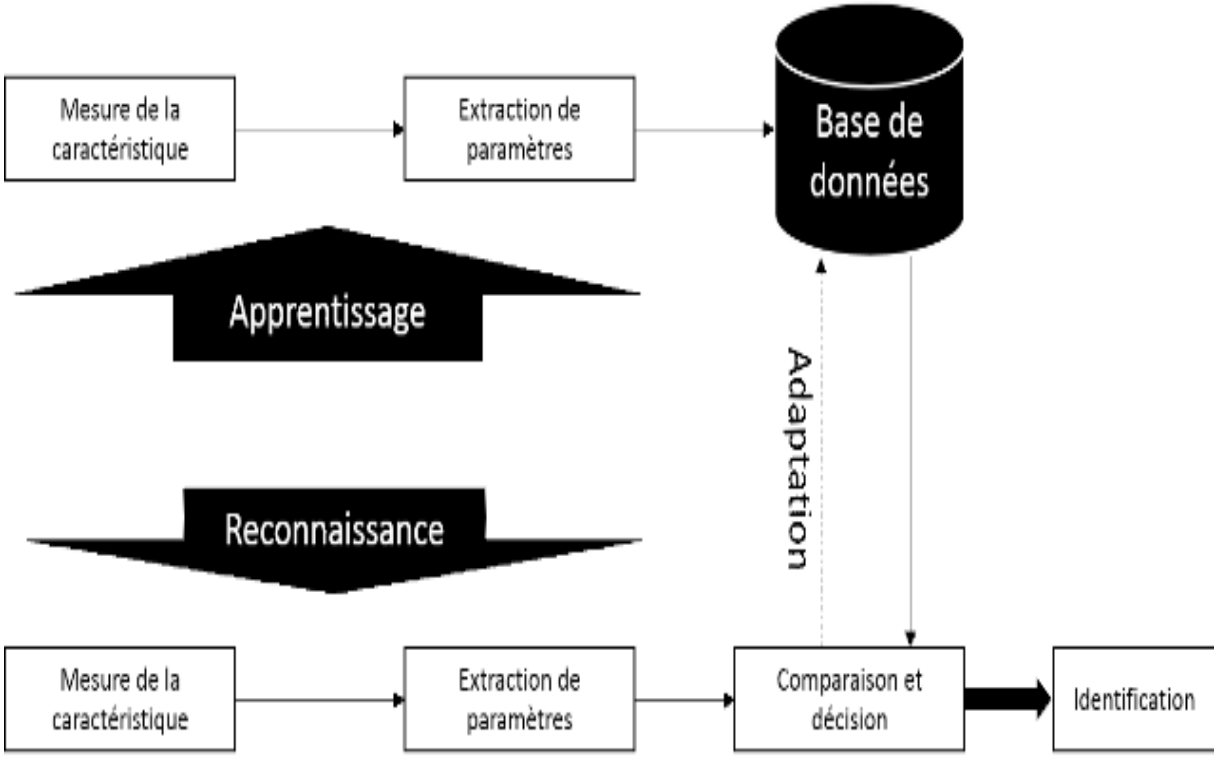

Figure 9: Représentation d'une architecture d'un système biométrique

I.5.1. Module d'apprentissage

C'est la première phase de tout système biométrique, il s'agit de l'étape pendant laquelle un utilisateur est enregistré dans le système pour la première fois et où une ou plusieurs modalités biométriques sont capturées et enregistrées dans une base de données à travers un ou plusieurs capteurs biométriques. Les données capturées ne sont pas enregistrées directement, le signal contient de l'information inutile et seuls les paramètres pertinents sont extraits et constituent ainsi un vecteur descripteur qui est moins volumineux que la capture brute mais qui est suffisamment discriminant. Cet enregistrement peut s'accompagner par l'ajout d'information biographique dans la base de données.

I.5.2. Module de base de données

Le module de base de données sert à stocker le modèle biométrique de la personne.

I.5.3. Module de reconnaissance

Le dernier module qui sert pour la phase de reconnaissance est celui de la mise en correspondance et de la prise de décision, il fait la comparaison entre la capture qui lui a été soumise et le modèle stocké dans la base de données en se basant sur le résultat de cette comparaison, le système juge de l'authenticité ou de l'imposture de la personne.

I.5.4. Module d'adaptation

Pendant la phase d'apprentissage, le système biométrique ne capture souvent que quelques instances d'un même attribut afin de limiter la gêne pour l'utilisateur. Il est donc difficile de construire un modèle assez général capable de décrire toutes les variations de cet attribut. De plus, les caractéristiques de cette biométrie ainsi que ses conditions d'acquisition peuvent varier. L'adaptation est donc nécessaire pour maintenir voir améliorer la performance d'un système utilisation après utilisation. L'adaptation peut se faire en mode supervisé ou non - supervisée mais le second mode est de loin le plus utile en pratique. Si un utilisateur est identifié par le module de reconnaissance, les paramètres extraits du signal serviront alors à réestimer son modèle. En général, le taux d'adaptation dépend du degré de confiance du module de reconnaissance dans l'identité de l'utilisateur. L'adaptation est quasi indispensable pour les caractéristiques non permanentes comme la voix. [2]

I.6. La Multi modalité

La biométrie multi modale désigne l'utilisation de plus d'une source d'information pour la reconnaissance d'un individu. Elle permet de combiner plusieurs modalités biométriques afin de diminuer les restrictions du système de reconnaissance monomodal. L'architecture d'un système multimodal est composée au minimum de deux systèmes monomodaux présentables sous diverses configurations :

- Plusieurs programmes traitant une même donnée biométrique acquise : multi-algorithme
- L'acquisition d'une même donnée par plusieurs capteurs : multi-capteur (exemple : empreinte digitale optique et thermique)
- L'acquisition indépendante de plusieurs données biométriques d'une même personne : multi-biométrie (exemple : visage et géométrie de la main)

Si l'acquisition et le traitement sont réalisés successivement, on parle d'architecture en série (voir figure), sinon d'architecture en parallèle s'ils sont réalisés simultanément. [7]

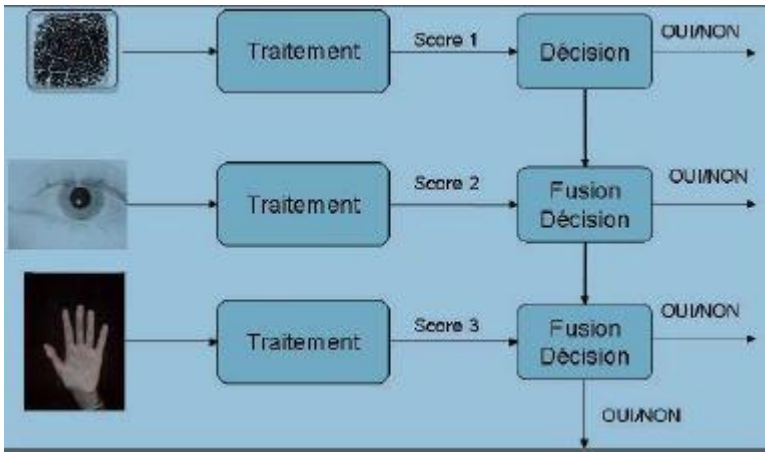

Figure 10: Architecture de fusion en série [7]

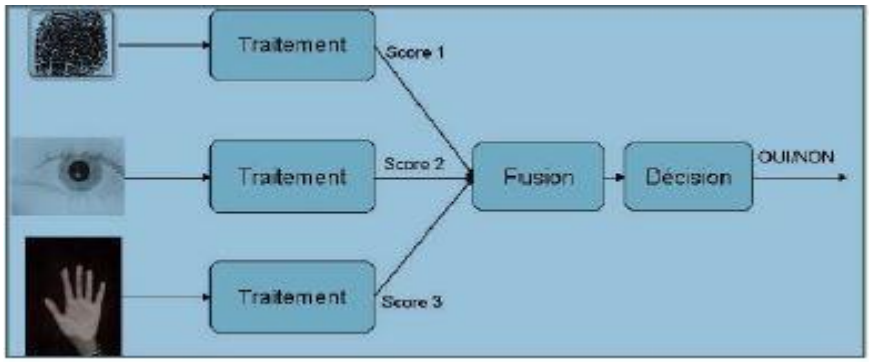

Figure 11: Architecture de fusion en parallèle [7]

I.7. Reconnaissance du visage

La reconnaissance faciale a fait l'objet de plusieurs travaux ces deux dernières décennies, 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).

Les méthodes appliquées dans les modules formant le système dépendent essentiellement de la technique d'acquisition d'images de visage. En effet, s'il s'agit d'un scanner tridimensionnel ou d'un système d'acquisition stéréoscopique, la détection de visages et l'extraction de signatures reposent sur des techniques de traitement 3D. Cette thématique est actuellement en pleine expansion. Elle met en évidence l'information de profondeur qui enrichit les données utilisées aussi bien dans la phase d'apprentissage que dans la phase d'identification. Toutefois, elle est limitée par le prix élevé de l'appareil d'acquisition (s'il s'agit d'un scanner 3D) et la difficulté d'installation (s'il s'agit d'un système stéréoscopique). Cette limite laisse le traitement 2D des images de visages plus accessibles et encore très explorés, tant dans le

domaine académique que dans le milieu industriel. Dans la suite de notre travail nous utiliserons la reconnaissance 2D [8].

Nous pouvons classer les méthodes de reconnaissance 2D sous 3 approches :

I.7.1. Approches globales

Le principe de ces approches est d'utiliser toute la surface du visage comme source d'information sans tenir compte des caractéristiques locales comme les yeux, la bouche, etc. L'une des méthodes la plus largement utilisée pour la représentation du visage dans son ensemble est le PCA. Les algorithmes globaux s'appuient sur des propriétés statistiques bien connues et utilisent l'algèbre linéaire. Ils sont relativement rapides à mettre en œuvre, mais sont sensibles aux variations d'illumination, de pose et d'expression faciale.

Parmi les approches les plus importantes réunies au sein de cette classe on trouve:

- L'Analyse en Composantes Principales (PCA ou Eigenfaces),
- L'Analyse Discriminante Linéaire (LDA),
- Machine à Vecteurs de Support (SVM),
- Les Réseaux de Neurones (RNA),
- Independent component Analysis (ICA) [8].

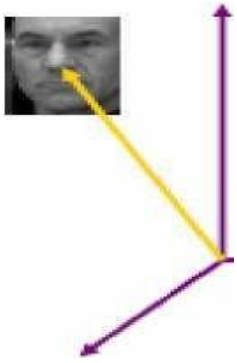

Figure 12: Principe des approches globales

I.7.2. Approches locales

On les appelle aussi les méthodes à traits, géométriques, à caractéristiques locales, ou analytiques. Ce type consiste à appliquer des transformations en des endroits spécifiques de l'image, le plus souvent autour des points caractéristiques (coins des yeux, de la bouche, le nez, etc.), l'énergie sera accordée aux petits détails locaux évitant le bruit engendré par les cheveux, les lunettes, les chapeaux, la barbe, etc. Mais leur difficulté se présente lorsqu'il s'agit de prendre en considération plusieurs vues du visage ainsi que le manque de précision dans la phase d'extraction des points constitue leur inconvénient majeur. Précisément, ces méthodes extraient les caractéristiques locales de visage comme les yeux, le nez et la bouche, puis utilisent leur géométrie et/ou l'apparence comme donnée d'entrée du classifieur. On peut distinguer deux pratiques différentes :

 La première repose sur l'extraction de régions entières du visage, elle est souvent implémentée avec une approche globale de reconnaissance de visage.

 La deuxième extrait des points particuliers des différentes régions caractéristiques du visage, tels que les coins des yeux, de la bouche et du nez.

Parmi ces approches on peut citer :

 Modèles de Markov Cachés (Hidden Markov Models (HMM)),

 L'Algorithme Elastic Bunch Graph Matching (EBGM),

 L'appariement de gabarits [8].

I.7.3. Approches hybrides

La robustesse d'un système de reconnaissance peut être augmentée par la fusion de plusieurs méthodes qui est appelée méthode hybride. Il est par ailleurs possible d'utiliser une combinaison de classifieurs basés sur des techniques variées dans le but d'unir les forces de chacun et ainsi pallier à leurs faiblesses. Les techniques hybrides combinent les deux méthodes précédentes pour une meilleure caractérisation des images de visages [8].

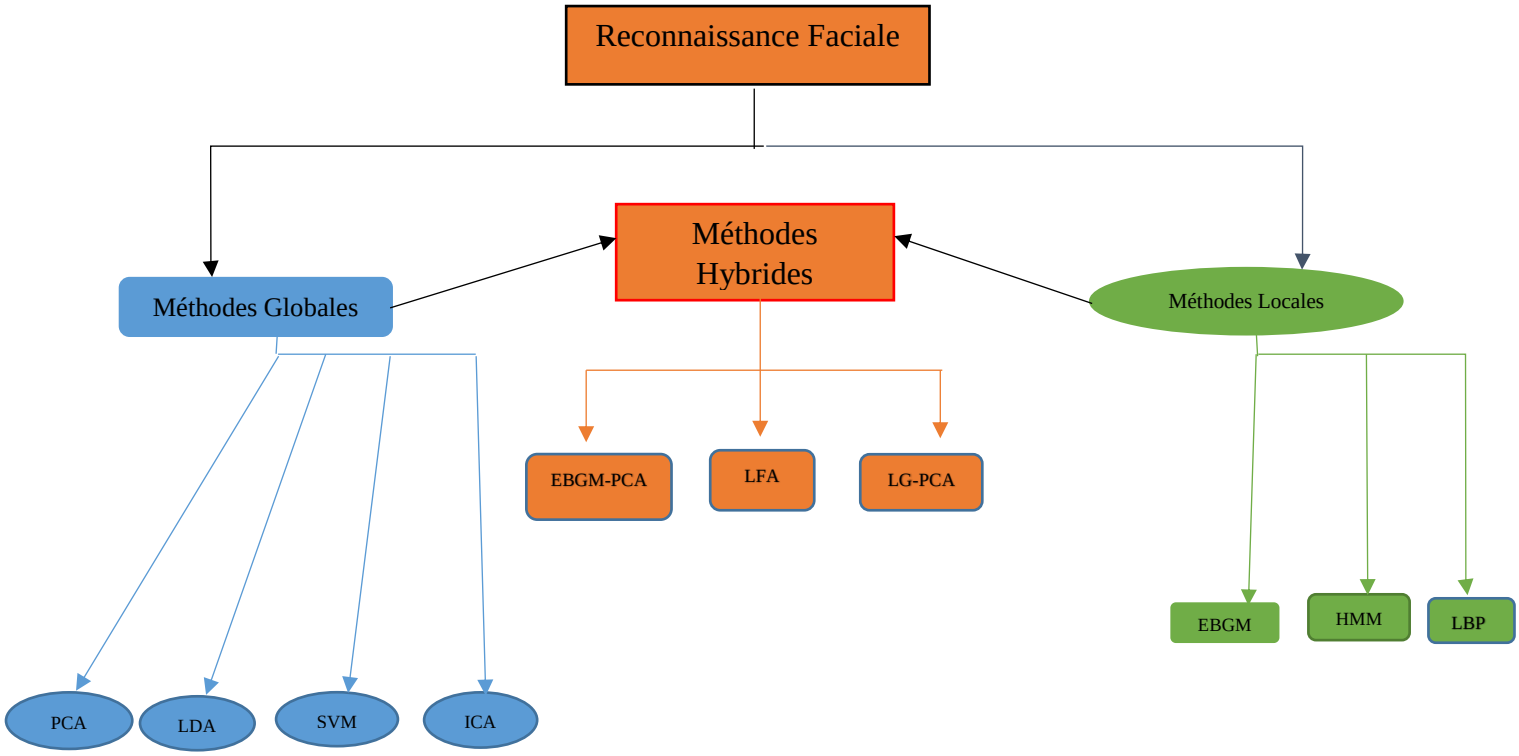

Figure 13: Arbre Récapitulatif

I.8. Principales difficultés de la reconnaissance de visage

Pour le cerveau humain, le processus de la reconnaissance de visages est une tâche visuelle de haut niveau. Bien que les êtres humains puissent détecter et identifier des visages dans une scène sans beaucoup de peine, construire un système automatique qui accomplit de telles tâches représente un sérieux défi. Ce défi est d'autant plus grand lorsque les conditions d'acquisition des images sont très variables. Il existe deux types de variations associées aux images de visages : inter et intra sujet. La variation inter-sujet est limitée à cause de la ressemblance physique entre les individus. Par contre la variation intra-sujet est plus vaste. Elle peut être attribuée à plusieurs facteurs que nous analysons ci-dessous.

I.8.1. Changement d'illumination

Les variations d'éclairage rendent la tâche de reconnaissance de visage très difficile. En effet, le changement d'apparence d'un visage dû à l'illumination, se révèle parfois plus critique que la différence physique entre les individus, et peut entrainer une mauvaise classification des images d'entrée [9].

Figure 14: Exemple de variation d'éclairage

I.8.2. Variations de pose

Le taux de reconnaissance de visage baisse considérablement quand des variations de pose sont présentes dans les images. La variation de pose est considérée comme un problème majeur pour les systèmes de reconnaissance faciale. Quand le visage est de profil dans le plan image (orientation < 30°), il peut être normalisé en détectant au moins deux traits faciaux (passant par les yeux). Cependant, lorsque la rotation est supérieure à 30°, la normalisation géométrique n'est plus possible [9].

Figure 15: Exemple de variation de pose

I.8.3. Expressions faciales

La déformation du visage qui est due aux expressions faciales est localisée principalement sur la partie inférieure du visage. L'information faciale se situant dans la partie supérieure du visage reste quasi invariable. Elle est généralement suffisante pour effectuer une identification. Toutefois, étant donné que l'expression faciale modifie l'aspect du visage, elle entraine forcément une diminution du taux de reconnaissance. L'identification de visage avec expression faciale est un problème difficile qui est toujours d'actualité et qui reste non résolu [9].

Figure 16: Exemple de variations d'expression faciale

I.8.4. Les vrais jumeaux

Qui ont le même indicatif d'ADN, peuvent tromper les personnes qui ne les connaissent pas (les personnes familières avec les jumeaux ont reçu une grande quantité d'informations sur ces derniers et sont donc beaucoup plus qualifiées à distinguer les jumeaux.). Il est peu probable que la vérification automatique de visage, ne pourra jamais détecter les différences très subtiles qui existent entre les jumeaux [9].

I.8.5. Occultations partielles

L'idéal pour un système de reconnaissance de visage étant d'être non intrusif, cependant l'occultation partielle change complétement l'apparence d'une partie du visage. Un visage peut être partiellement masqué par des objets dans la scène, ou par le port d'accessoire tels que les lunettes, écharpe, chapeau, les cheveux longs etc. Ceci ne provoque pas uniquement une dégradation des performances en reconnaissance faciale, mais peut aussi avoir des conséquences en termes de sécurité. Gross et Al ont étudié l'impact du port de lunettes de soleil et du cache-nez occultant la partie inférieure du visage sur la reconnaissance faciale .Leurs résultats expérimentaux semblent que, dans ces conditions, les performances des algorithmes de reconnaissance restent faibles.

I.9. Evaluation de performance

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 »[7].

$$T = \frac{\text{nombre d'identifications correctes}}{\text{nombre total de test effectué}} \qquad \text{(I.1)}$$

T : Taux de reconnaissance

I.9.1. 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.

I.9.1.1. 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.

I.9.1.2. Le taux de fausses acceptations :

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

$$P(I,VC) = \begin{cases} accepté\ si\ S(VI,VC) \geq \alpha \\ réfusé\ sinon \end{cases} \qquad \text{(I.2)}$$

I : Identité de la personne P

VC : Vecteur caractéristique de la personne P

VI : Vecteur caractéristique déjà enregistré

S : Score de similitude

α : Seuil du taux d'exactitude croisé [7].

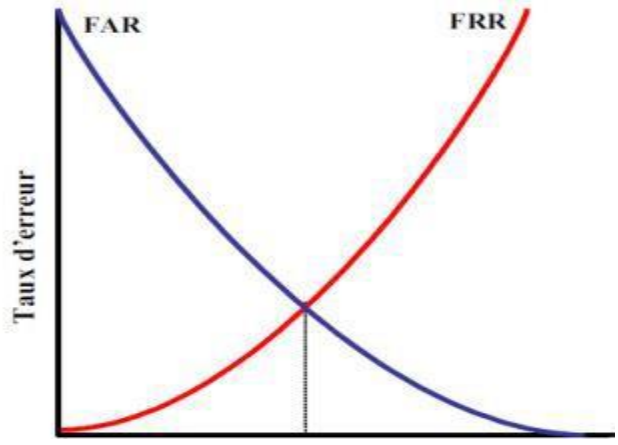

Figure 17: Courbe du point d'équivalence des erreurs dans un système biométrique

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.

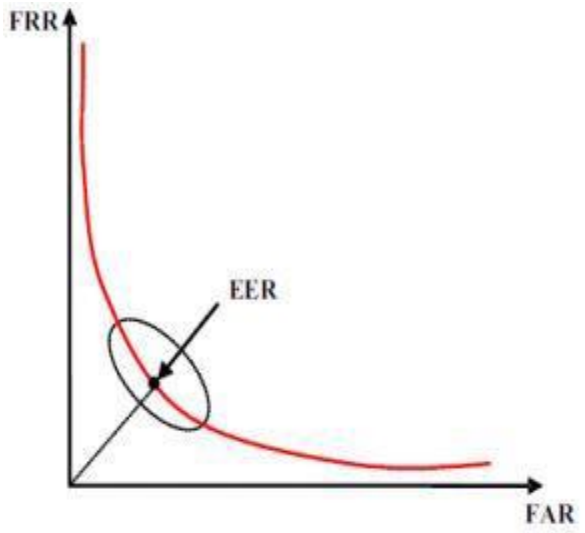

Figure 18: Courbe FRR en fonction du FAR

I.9.2. 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 [7].

$$P(I,VC) = \begin{cases} Ik\ si \max S(VIk,VC) \geq \alpha \\ IN+1\ sinon \end{cases} \quad k = 1 \dots N \qquad \text{(I.3)}$$

I : identité de la personne P

K : le nombre de personne enregistrée

VC : Vecteur caractéristique de la personne P

VI : Vecteur caractéristique déjà enregistré

S : Score de similitude

α : seuil du taux d'exactitude croisé

I1...IN : identités enregistrées

IN+1 : Identité rejetée [7].

Conclusion

Dans ce chapitre, nous avons présenté les technologies utilisées dans les systèmes biométriques pour l'identification de personnes. Nous avons aussi donné un aperçu sur les techniques de

mesure de leurs performances. Cette étude nous a permis de constater que la reconnaissance de visage suscite de plus en plus l'intérêt de la communauté scientifique, car elle présente plusieurs challenges et verrous technologiques. Enfin, nous avons mis en évidence les différentes difficultés inhérentes à la reconnaissance automatique de visages, ce qui nous a permis de bien définir les problématiques traitées dans ce mémoire.

Chapitre 2 : Spécifications et Algorithme

II.1. Introduction

Au jour d'aujourd'hui la reconnaissance du visage s'est présentée très performante, n'atteignant pas encore le système visuel humain. Cependant de nombreuses techniques ont été développées ces dernières années, dans ce chapitre nous allons passer en revue quelques-uns des plus populaires notamment l'Eigenface, le LBP et la méthode de Viola et Jones (fonction Haar). Nous préciserons aussi le langage utilisé puis enfin en conclusion nous choisirons la méthode à utiliser pour la détection et la reconnaissance.

II.2. Spécification de la technique et du langage utilisé

L'objectif de notre travail consiste à mettre en place un système de reconnaissance faciale capable d'assurer le contrôle d'accès.

II.2.1. Techniques de détection et de reconnaissance

La détection de visages dans l'image est un passage indispensable et crucial avant la phase de reconnaissance. En effet sans une détection de visages efficace, le processus de reconnaissance de visages ne serait jamais intégralement automatique.

- La détection de visages consiste à détecter des visages humains dans une image numérique. Il s'agit alors de repérer la présence dans l'image de caractéristiques de visages humains tels que les yeux, le nez, la bouche, réunis sur une surface suffisamment petite pour qu'on puisse considérer qu'ils appartiennent à la même personne. Ce problème est rendu d'autant plus délicat qu'un grand nombre de facteurs viennent compliquer sa solution, par exemple la couleur, la forme, la présence de lunettes ou d'une moustache, l'orientation (face ou profil), l'expression faciale modifiant la géométrie du visage (rire, peur, colère), etc. Cependant les algorithmes de détection de visages ont fait de notables progrès en efficacité et en rapidité au point, qu'à l'heure actuelle, presque tous les appareils photo numériques offrent la fonction de mise au point sur les visages [10].
- La reconnaissance de visages consiste à identifier une personne sur la base de son visage. Il faut alors mettre en œuvre des méthodes plus sophistiquées que pour une simple détection. En effet la présence des yeux, d'un nez et d'une bouche ne suffit pas à l'identification d'un individu. Il faut tenir compte des caractéristiques spécifiques à une personne. C'est ainsi que l'on mesure les positions relatives d'une série de points

caractéristiques disposés sur un visage (les points représentent les positions des yeux, du nez, de la bouche, etc et des lignes sont tracées entre ces points, ce qui permet d'obtenir une géométrie du visage). Il est alors possible de comparer ces mesures avec celles faites sur une image (ou une série d'images) de référence pour laquelle on dispose d'un identificateur. La comparaison fournit alors un indice de ressemblance qui permettra de dire si le visage à reconnaître est celui de la personne connue [10].

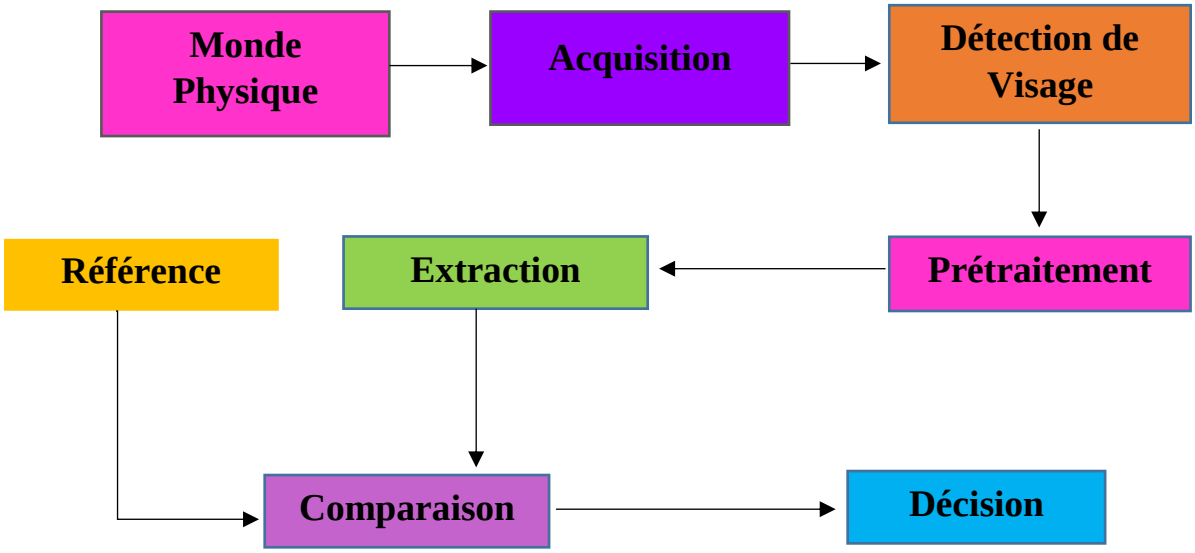

Figure 19: Schéma générale d'un système de reconnaissance de visages

II.2.1.1. Détection de visages à l'aide de la fonction Haar

La fonction Haar 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 20).

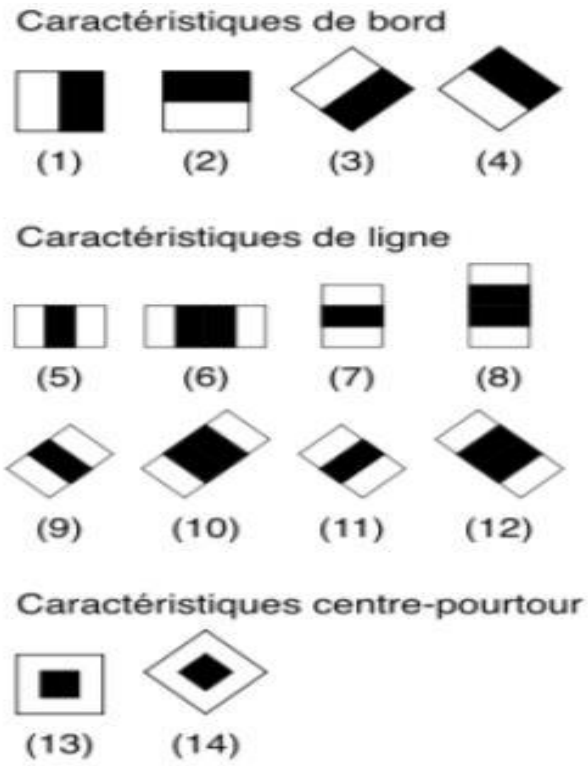

Figure 20: Caractéristiques Pseudo-Haar [11]

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 21 et 22).

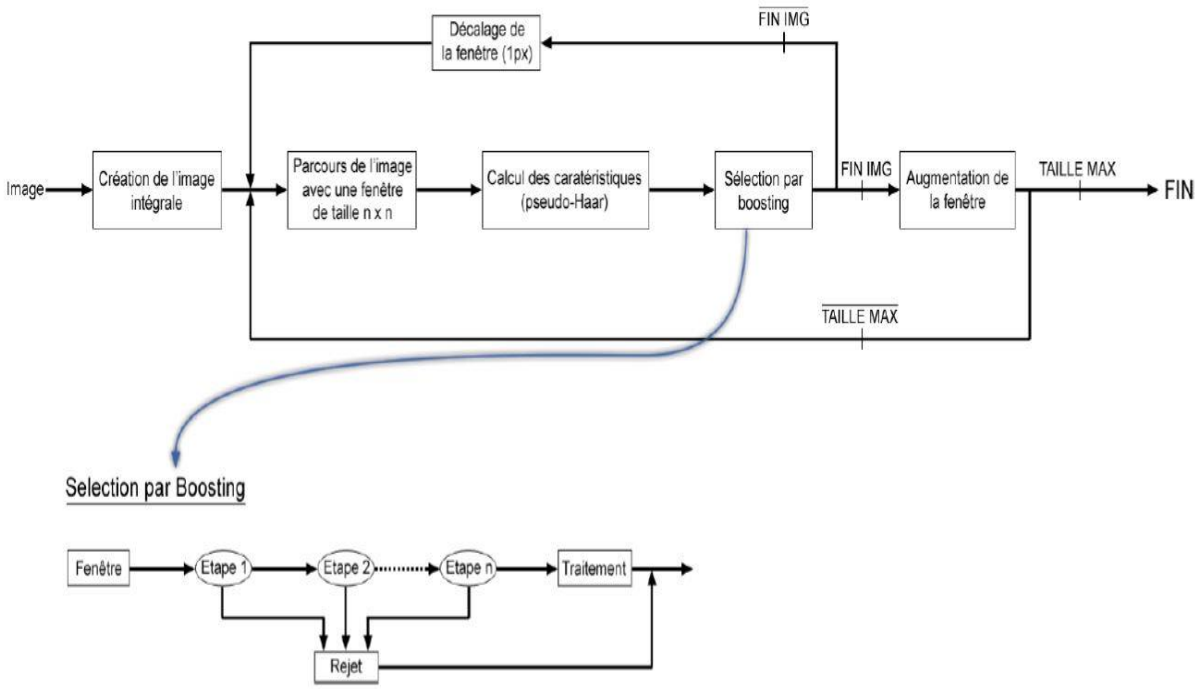

Figure 21: Etapes de détection d'un visage [11]

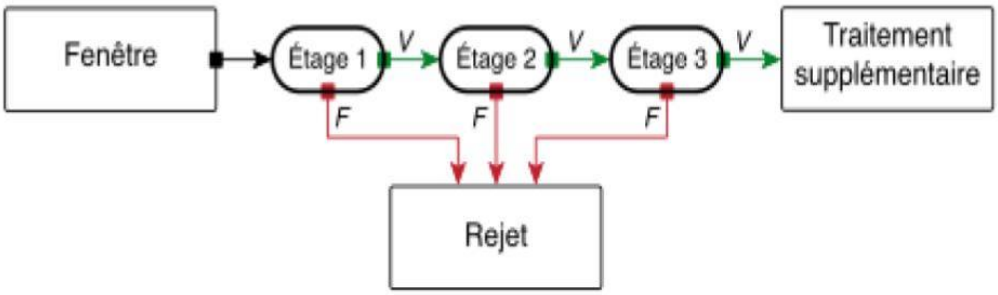

Figure 22: Cascade de Viola et Jones [11]

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.2.1.2. Détection et reconnaissance à l'aide de LBP

Les motifs binaires locaux (LBP) sont des descripteurs de texture qui peuvent également être utilisés pour représenter des visages, puisqu'une image de visage peut être vue comme une composition de motifs de micro-texture.

LBP présente un espace caractéristique discriminant qui peut être appliqué à des problèmes de détection des visages et de reconnaissance [12].

En bref, la procédure consiste à diviser une image faciale en plusieurs régions où les caractéristiques LBP sont extraites et concaténées dans un vecteur de caractéristiques qui sera plus tard utilisé comme descripteur facial.

L'opérateur LBP d'origine, présenté par Ojala et al, est un puissant moyen de description de la texture. L'opérateur étiquette les pixels d'une image par seuillage du voisinage 3x3 de chaque pixel avec la valeur centrale et enregistre le résultat comme un nombre binaire. Ensuite, l'histogramme des étiquettes peut être utilisé comme descripteur de texture [13]. (Voir la figure 23 pour une illustration de l'opérateur LBP de base)

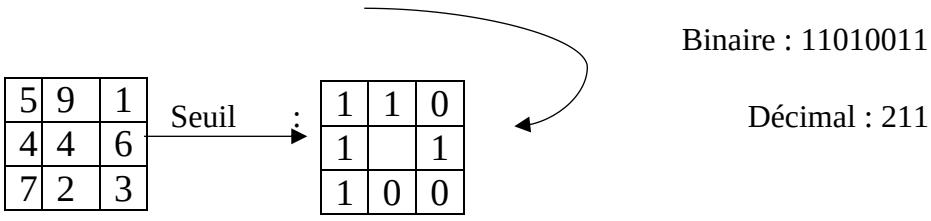

Figure 23: operateur LBP de base [13]

En d'autres termes, étant donné une position de pixel (xc, yc), LBP est défini comme un ensemble ordonné de comparaisons binaires d'intensités de pixels entre le pixel central et ses pixels environnants. La valeur d'étiquette décimale résultante du mot de 8 bits est calculée. Dans différentes publications, les valeurs résultantes circulaires 0 et 1 sont lues soit dans le sens des aiguilles d'une montre, soit dans le sens inverse des aiguilles d'une montre. Dans ce pfe, le résultat binaire sera obtenu en lisant les valeurs dans le sens des aiguilles d'une montre, à partir du voisin supérieur gauche, comme on peut le voir sur la figure 23.

Afin de traiter les textures à différentes échelles, l'opérateur LBP a été étendu pour utiliser des quartiers de tailles différentes. En utilisant des voisins circulaires et une interpolation bilinéaire des valeurs de pixels, n'importe quel rayon et nombre d'échantillons dans le voisinage peuvent être manipulés. Par conséquent, la notation suivante est définie:

(P, R) ce qui signifie P points d'échantillonnage sur un cercle de rayon R [13].

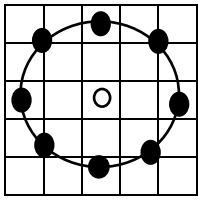

Figure 24: Le quartier circulaire (8,2). Les valeurs des pixels sont interpolées bilinéairement chaque fois que le point d'échantillonnage n'est pas au centre d'un pixel.

Lors du calcul des opérations de pixel prenant en compte les voisinages NxN à la limite d'une image, une partie du masque NxN se trouve hors du bord de l'image. Dans de telles situations, différentes techniques de remplissage sont généralement utilisées, telles que le remplissage à zéro, la répétition d'éléments de bordure ou l'application d'une réflexion miroir pour définir les bordures d'image. Cependant, dans le cas de l'opérateur LBP, la limite critique, définie par le rayon R de l'opération circulaire, n'est pas résolue en utilisant une technique de remplissage, plutôt que cela, l'opération est démarrée au pixel d'image (R, R) [13]. L'avantage est que l'histogramme final des étiquettes LBP ne sera pas affecté par les bordures car une petite zone sur les bordures de l'image n'est pas utilisée. Par exemple pour une image de N x M, le vecteur caractéristique est construit en calculant le code LBP pour chaque pixel (xc, yc) avec xc € {R+1,…, N-R} et yc € {R+1,…, M-R}.

Les images suivantes montrent le résultat de l'application de modèles binaires locaux LBP sur une image. Dans les images résultantes, on peut clairement observer pourquoi les LBP sont appelés descripteur de texture.

Figure 25: Résultat de l'application du LBP sur l'image a [12]

Comme le montre la figure 25, les contours des traits du visage (yeux, bouche, nez, parenté, sourcils ...) sont clairement remarqués. Dans l'image des étiquettes de premier niveau, les contours sont constamment mis en évidence, ce qui donne une vue d'ensemble générale de la netteté de l'image, ce qui est utile pour distinguer les images sans visage dans la première phase rapide. D'un autre côté, dans l'image des étiquettes du deuxième étage, les informations de texture locales sont plus détaillées, ce qui est utile pour la décision finale où le facteur de l'image est évalué localement [13].

Une autre extension à l'opérateur d'origine est le LBP uniforme. Un code LBP est uniforme s'il contient au plus deux transitions de bits de 0 à 1 ou vice-versa, lorsque la chaîne binaire est considérée circulaire. Par exemple 00011110 et 10000011 sont des codes uniformes. L'utilisation d'un code LBP uniforme présente deux avantages. Le premier est le gain en mémoire et en temps de calcul. Le deuxième est qu'il permet de détecter uniquement les textures locales importantes, comme les spots, les fins de ligne, les bords et les coins [14]. Ojala a constaté que seuls 58 des 256 patterns LBP sont uniformes mais expérimentalement, il a été constaté que 90% des patterns rencontrés dans les images sont uniformes [15].

Le descripteur de modèles binaires locaux n'est pas seulement une fonction discriminante efficace pour le visage, mais aussi pour la tâche de reconnaissance faciale.

Une fois le code LBP calculé pour tous les pixels de l'image, on calcule l'histogramme de cette image LBP pour former un vecteur de caractéristiques représentant l'image faciale. En réalité, afin d'incorporer plus d'informations spatiales au vecteur représentant le visage, on divise tout d'abord l'image codée par l'opérateur LBP en petites régions et l'histogramme est construit pour chaque région. Finalement, on concatène tous les histogrammes des régions afin de former un grand histogramme représentant l'image des caractéristiques faciales (voir figure 26).

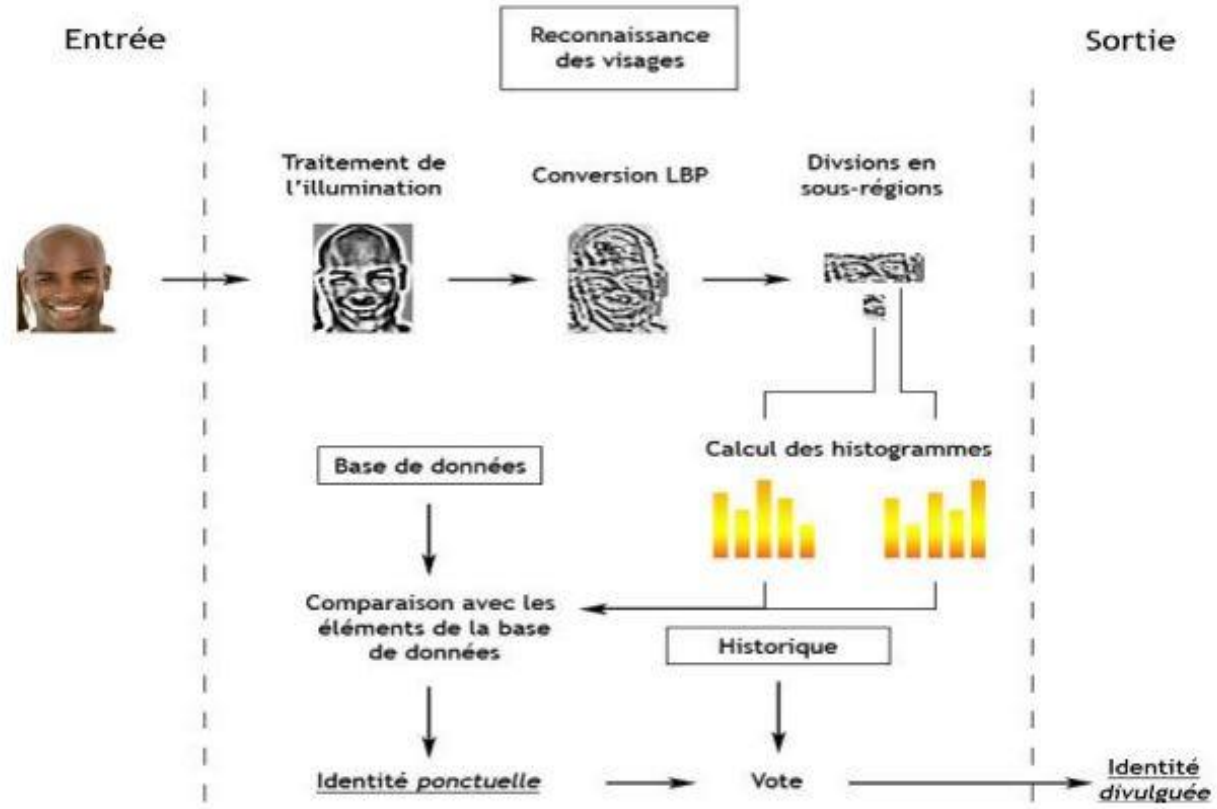

Figure 26: Illustration du stage de reconnaissance [15]

Etant donnés deux histogrammes de LBP H1 et H2 de deux visages, l'étape suivante est d'utiliser une métrique pour calculer la similarité entre ces deux histogrammes. En testant les trois métriques x^2, Histogrammes intersection et Log likelihood statistic, Ahonen et al ont observé que la première métrique permet d'obtenir les meilleurs résultats :

$$x^2(H^1, H^2) = \sum i \frac{(H_i^1 - H_i^2)^2}{(H_i^1 + H_i^2)^2} \qquad \text{(II.1)}$$

Le descripteur LBP est robuste comme méthode, il présente :

- une faible complexité de calcul.
- Il peut être appliqué à la fois pour la détection et la reconnaissance.
- Robuste aux changements d'éclairage.

II.2.1.3. Détection et reconnaissance à l'aide d'Eigenface

L'algorithme ACP, PCA en anglais (Principal Component Analysis) est né des travaux de MA. Turk et AP. Pentland au MIT Media Lab, en 1991. Il est aussi connu sous le nom d'Eigenface car il utilise des vecteurs propres et des valeurs propres. Cet algorithme s'appuie sur des

propriétés statistiques bien connues et utilise l'algèbre linéaire [16]. Il est à la base de nombreux algorithmes globaux actuels.

L'idée principale consiste à exprimer les N images d'apprentissage selon une base de vecteurs orthogonaux particuliers, contenant des informations indépendantes d'un vecteur à l'autre. Ces nouvelles données sont donc exprimées d'une manière plus appropriée à la reconnaissance du visage. Nous voulons extraire l'information caractéristique d'une image de visage, pour l'encoder aussi efficacement que possible afin de la comparer à une base de données de modèles encodés de manière similaire. En termes mathématiques, cela revient à trouver les vecteurs propres de la matrice de covariance formée par les différentes images de notre base d'apprentissage [17].

- **Le vecteur d'image :**

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

$X_i = \begin{bmatrix} a1,1 & \cdots & a1,m \\ \vdots & \vdots & \vdots \\ an,1 & \cdots & an,m \end{bmatrix}$ Donne le vecteur d'image Vi=$\begin{bmatrix} a1,1 \\ \vdots \\ a1,m \\ \vdots \\ an,m \end{bmatrix}$ qui forme la grande matrice :

$$V=[V1\ V2\ \dots VN] = \begin{bmatrix} a1,1 & b1,1 & z1,1 \\ \vdots & \vdots & \vdots \\ a1,m & b1,m & z1,m \\ \vdots & \vdots & \vdots \\ an,m & bn,m & zn,m \end{bmatrix}$$ avec N le nombre d'images (II.2)

- **L'image moyenne :**

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

$$M = \frac{1}{N}\sum_{i=1}^{N} Vi \qquad \text{(II.3)}$$

- **Ajustement des images par rapport à la moyenne :**

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 ai cette matrice :

ai = Vi – M 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 :

$$C = \sum_{i=1}^{N}. a_1 * a_1^T = AA^T \ , \quad A=[a_1 a_2 \dots a_N] \qquad \text{(II.4)}$$

La prochaine étape consiste à calculer les vecteurs propres et les valeurs propres de cette matrice de taille (DxD) c'est-à-dire de l'ordre de résolution d'une image. Le problème est que cela peut parfois être difficile et très long ; en effet si D > N (si la résolution est supérieure au nombre d'images) par exemple pour 50 images d'une taille d'image de 128 × 128, il faudrait calculer la matrice de dimensions 16384 × 16384 et trouver 16384 vecteurs propres. Ce n'est pas très efficace car nous n'avons pas besoin de la plupart de ces vecteurs. Le rang de la matrice de covariance est limité par le nombre d'images dans l'ensemble d'apprentissage si nous avons N images, nous aurons N – 1 (50-1) vecteurs propres qui contiendront de l'information (les vecteurs propres restants auront des valeurs propres associées nulles). Donc au lieu de résoudre une matrice de 16384 x 16384 nous pourrions résoudre une matrice de 50 x 50, nous ferions donc un gain de temps de calcul et nous passerions d'une complexité de l'ordre du nombre de pixels dans une image à celle de l'ordre du nombre d'images.

Un des théorèmes de l'algèbre linéaire stipule que les vecteurs propres di et les valeurs propres λi de la matrice de covariance peuvent être obtenus en trouvant les vecteurs propres et les valeurs propres de la matrice E= $A^T A$ (dimensions N × N).

- **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 N×N, dont nous trouverons les vecteurs propres ei.

$$Eei = (A^T)Aei = \lambda_i ei \qquad \text{(II.5)}$$

Avec « λi » la valeur propre associée au vecteur propre « ei ».

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

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

Donc les vecteurs propres di de la matrice de covariance C est égal aux vecteurs propres ei de la matrice E multiplié par A.

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

Pour ei vecteur propre de la matrice E associé à la valeur propre λi, nous avons un vecteur propre di de la matrice C associé à la même valeur propre λi.

$$di = Aei \qquad \text{(II.6)}$$

Ensuite il ne faut sélectionner que les k meilleurs vecteurs propres (ceux avec les k de plus grandes valeurs propres). Pour le choisir, les chercheurs ont adopté différentes solutions:

- ✓ Pour un ensemble de 115 images, Sirovitch et Kirby ont trouvé que 40 vecteurs propres sont suffisants pour représenter efficacement cet ensemble,
- ✓ Turk et Pentland l'ont choisi heuristiquement. Pour leurs tests, sur une base de 16 individus, 7 vecteurs propres ont été retenus,
- ✓ Moghaddam a préservé, pour comparer différentes approches de reconnaissance de visages 20 vecteurs propres en justifiant son choix par une erreur de reconstruction raisonnable (0.0012) et un taux de reconnaissance 80% obtenu par eigenfaces sur une base de 1829 images,
- ✓ Zhao et al ont retenu 300 vecteurs propres pour une base de 1038 images après avoir observé que pour un nombre très élevé, les eigenfaces ne représentent pas des visages, donc leur choix était basé sur l'allure des eigenfaces au lieu des valeurs propres [2].

Donc, les k premiers vecteurs propres correspondant aux k plus grandes valeurs propres sont un paramètre critique sur lequel dépend la performance du système de reconnaissance de visages (temps de calcul et taux de reconnaissance). Son choix dépend des contraintes d'application liées au temps de calcul et au nombre des images de visages disponibles et aussi de la qualité des images de visages reconstruites.

- Détermination du poids des images d'entrée :

On définit un espace vectoriel engendré par ces k vecteurs propres que l'on appelle l'espace des visages E_v où seront projetées nos images de départ. Une image Vi est alors transformée en ses composantes eigenfaces (vecteurs propres) par une simple opération de projection vectorielle.

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

$$P_k = d_k^T Ai \qquad \text{avec k allant de 1 à } l \qquad \text{(II.7)}$$

- L'espace visage :

Les vecteurs P_k forment un vecteur qui décrit la contribution de chaque eigenface dans la représentation de l'image d'entrée. Nous avons un vecteur πi où i représente le $ieme$ visage :

$$\pi_i = \begin{bmatrix} p1 \\ p2 \\ pl \end{bmatrix} \qquad \text{(II.8)}$$

Maintenant passons à la phase d'identification. Cette phase consiste tout d'abord à soustraire le visage moyen du vecteur image à identifier et obtenir les caractéristiques propres à ce visage :

a = V – M V étant le vecteur image de l'image à identifier

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

$$P_k = d_k^T a \qquad \text{(II.9)}$$

L'espace visage sera donné alors par :

$$\pi = \begin{bmatrix} p1 \\ p2 \\ pl \end{bmatrix} \qquad \text{(II.10)}$$

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

$m = min|\pi - \pi_i|$: Distance minium entre les points du visage patron (II.11)

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 [7].

L'eigenface est une méthode globale assez utilisée dans la reconnaissance faciale, il est :

- Rapide à mettre en œuvre
- Sensible aux variations d'éclairage, de pose et d'expression faciale.

II.2.2. Choix des méthodes

Après avoir étudié les différentes méthodes, nous avons décidé d'utiliser le descripteur LPB, pour la reconnaissance faciale en raison de sa robustesse aux changements d'éclairage, ce qui rendra le programme plus adapté à différents environnements et contextes.

Notre choix pour la détection fut la fonction de Haar développée par Paul Viola et Michael Jones, en raison de sa rapidité estimée à environ 15 fois plus rapide que toute autre approche [18].

II.2.3. Choix du langage de programmation

Nous avons opté pour le langage python pour la réalisation de notre travail. Ce choix se justifie par la puissance de ce langage qui se veut performant, simple et facile d'implémentation. Il compte beaucoup 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++ etc.

Conclusion

Au cours de ce chapitre nous avons décrit différentes méthodes notamment le descripteur LBP la fonction Haar, et l'Eigenface, puis précisé les méthodes à utiliser pour la suite de notre travail, notamment le LPB pour la reconnaissance de visage et la fonction Haar pour la détection du visage. En fin nous choisîmes Python comme langage de programmation.

Chapitre 3 : Conception et implémentation

III.1. Introduction

Dans ce chapitre, il sera question de matériel, de logiciel, d'implémentation de notre système de reconnaissance ainsi que la présentation de nos tests et résultats obtenus.

III.2. Environnement de travail

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

III.2.1.1. Webcam

Nous avons utilisé la caméra interne du pc de qualité photo 0,9 mégapixels, format 16:9 avec une résolution de 1280x720. En qualité vidéo, notre webcam a une résolution de 720p avec une fréquence d'images qui équivaut à 30fps.

III.2.1.2. L'ordinateur portable

L'ordinateur portable a les caractéristiques suivantes : 6 giga de mémoire Ram, processeur Intel core i7 HD graphics family, quatrième génération.

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, NumPy, PIL, OS et DLIB.

III.2.2.1. Open cv

Open cv (**Open** Source **C**omputer **V**ision) est une bibliothèque proposant un ensemble de plus de 2500 algorithmes de vision par ordinateur, accessibles au travers d'API pour les langages C, C++, et Python. Elle est distribuée sous une licence BSD (libre) pour les plateformes Windows, GNU/Linux, Android et MacOS.

Elle est utilisée par une communauté de plus de 40 000 membres actifs. C'est **la** bibliothèque de référence pour la vision par ordinateur, aussi bien dans le monde de la recherche que celui de l'industrie [19]. Ces algorithmes peuvent être utilisés, pour détecter et reconnaître des visages, identifier des objets, classer des actions humaines, suivre des mouvements de caméra, suivre des objets en mouvement, extraire des modèles 3D d'objets, trouver des images similaires à partir d'une base de données etc.

Parmi les caractéristiques d'Open CV on peut citer :

• Analyse du mouvement (flot optique, segmentation de mouvement, suivi).

• Reconnaissance d'objets (Eigen-méthodes, HMM).

• Diverses structures de données dynamiques (listes, les files d'attente, ensembles, arbres, graphes).

• Traitement de l'image de base (filtrage, détection de contour, détection d'angle, l'échantillonnage et l'interpolation, de conversion des couleurs, des opérations morphologiques, histogrammes, pyramides d'images).

III.2.2.2. NumPy

NumPy (Numérique python) est une bibliothèque du langage de programmation Python, destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux.

III.2.2.3. PIL

PIL (Python Imaging Library) est une bibliothèque de traitement d'images pour le langage de programmation Python. Elle permet d'ouvrir, de manipuler, et de sauvegarder différents formats de fichiers graphiques. La bibliothèque supporte plusieurs formats de fichier, parmi lesquels PNG, JPEG, GIF, TIFF, et BMP. Il est aussi possible d'ajouter son propre décodeur de fichiers pour étendre le nombre de formats disponibles.

III.2.2.4. Os

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

III.2.2.5. Dlib

Dlib est une collection d'algorithmes divers dans l'apprentissage automatique, la vision par ordinateur, le traitement d'image et l'algèbre linéaire. 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.

III.3. Description du programme

Pour la réalisation de notre programme de reconnaissance faciale, nous avons subdivisé le programme en 3 sous-programmes complémentaires réalisant respectivement, la tâche de détection et d'enregistrement de visages, la tâche d'apprentissage et enfin la tâche de reconnaissance.

III.3.1. Sous-programme de détection et enregistrement de visages

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 de la webcam, 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 établit, nous nous intéresserons à une zone particulière, le visage humain. Pour ça, nous importons un fichier xml aussi appelé haarcascade_frontalface_alt2 permettant de détecter un visage humain dans une image.

```
face_cascade = cv2.CascadeClassifier("./haarcascade_frontalface_alt2.xml")
face = face_cascade.detectMultiScale(gray, scaleFactor=1.2, minNeighbors=4, minSize=(d.min_size, d.min_size)
```

Figure 27: Code import du classifieur xml et fonction de détection de visages

III.3.1.3. Capture du visage

Une fois un visage détecté, il est enregistré dans la base de données.

III.3.1.4. Base de données

D'une part nous avons les photos de visages capturées lors de la détection, qui seront stockées dans un dossier (Dataset) et d'autre part un fichier texte, comportant la liste des identifiants ainsi que les noms et prénoms des personnes enregistrées. L'enregistrement du nom, prénom et l'attribution d'un numéro ID se fait au lancement du sous-programme de détection et d'enregistrement de visages à travers l'appel de la fonction ‘’AjoutNom()’’.

III.3.1.5. Prétraitements

Le prétraitement est l'ensemble des techniques utilisées afin d'améliorer la fiabilité et la robustesse du système, en réduisant au maximum les variations pouvant rendre plus complexe que nécessaire la reconnaissance ; tels que : la couleur, la lumière, les bruits.

Dans le sous-programme de détection et d'enregistrements de visages, nous avons appliqué quelques opérations de traitements d'image sur les images de visages détectées, avant de les stocker dans le Dataset.

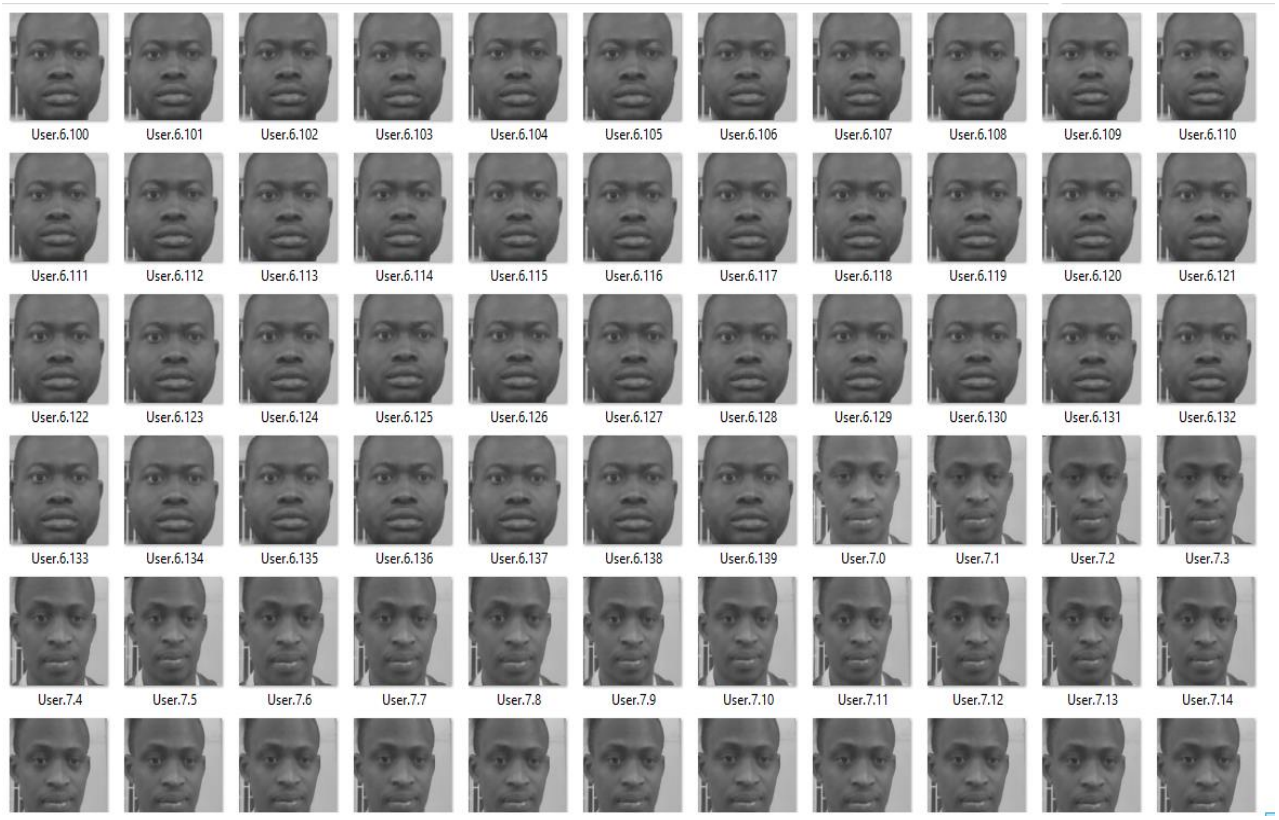

Figure 28: images de visages stockées dans le Dataset

Les prétraitements qui ont optimisé nos résultats sont :

- **La conversion en niveaux de gris et son recadrage (110x110 Pixel)**

Figure 29: image de visage convertit en niveaux de gris (110x110)

- **Mettre les yeux sur un même axe**

  Le fichier xml appelé haarcascade_eye permettant la détection des yeux est utilisé en ce sens.

Figure 30: Alignement des yeux sur le même axe

III.3.2. Sous-programme d'apprentissage

La base de données étant mise en place, ce sous-programme nous permet de créer un classificateur, issue de l'extraction des caractéristiques des images de visage de notre base de données, à l'aide du classifieur LBP.

L'exécution de ce sous-programme nous fournit donc le classificateur en format xml, que nous avons nommé '' classificateurLBPH'.

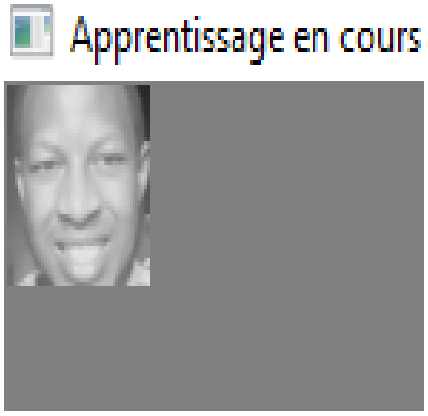

Figure 31: exécution du sous-programme d'apprentissage

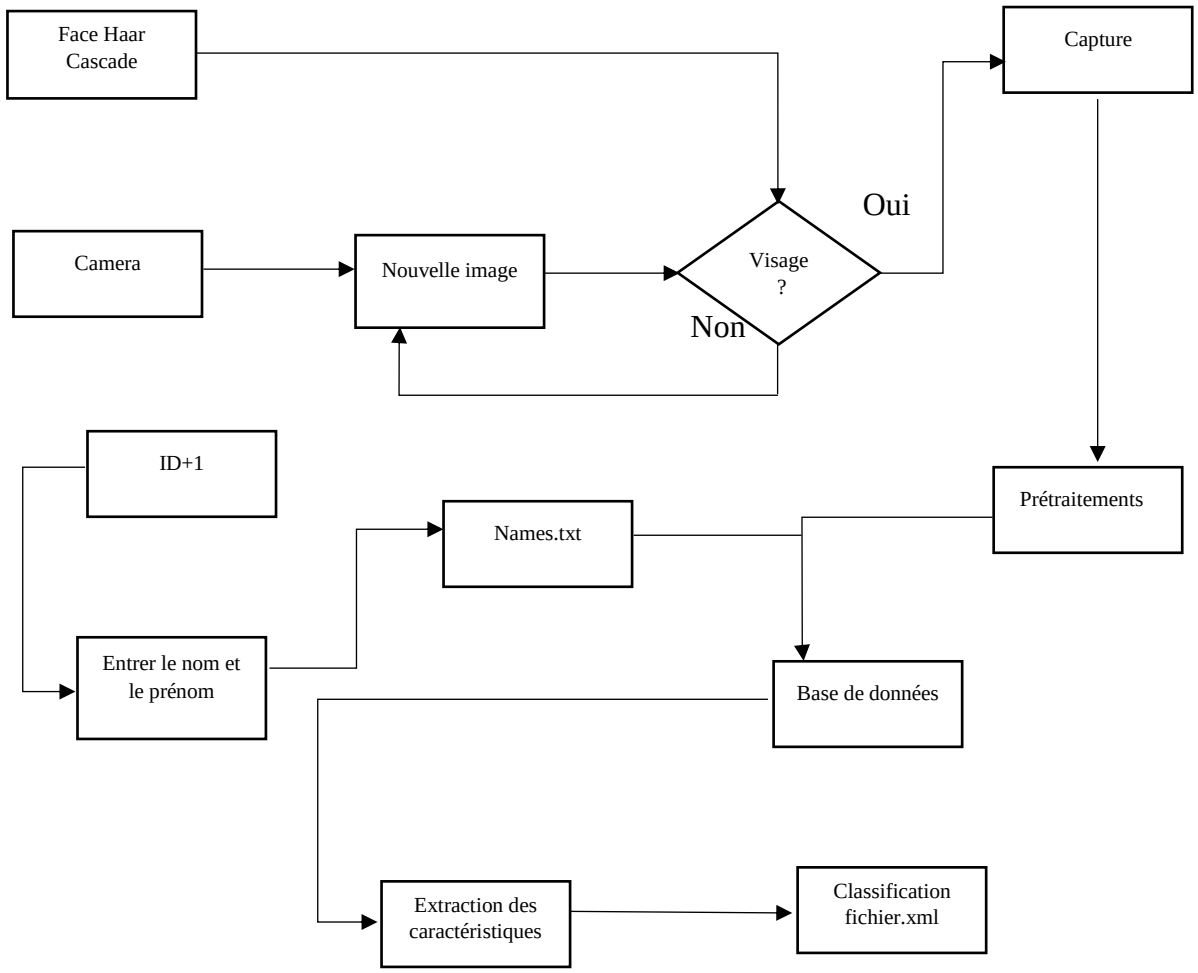

Figure 32: Organigramme phase de détection, enregistrements de visages et apprentissage

III.3.3. Sous-programme de reconnaissance

Ce sous-programme, est à peu près similaire à celui de la détection et d'enregistrements de visages, sauf qu'au lieu d'enregistrer les visages, une comparaison (identification) est effectuée avec la base de données au travers du fichier xml d'apprentissage, qui contient les caractéristiques des visages déjà enregistrées.

Une fois exécuté, ce sous-programme vous demande d'entrer 0 pour identifier les visages sur une photo ou un autre chiffre différent de 0 pour procéder à l'identification en temps réel ou en vidéo.

Le chemin d'accès de la photo ou de la vidéo devra être copié manuellement dans le sous-programme de reconnaissance.

L'une des fonctions principales de ce sous-programme est la fonction Predict, qui retourne un indice de confiance et un ID, en fonction du visage à identifier. Cet indice de confiance est utilisé comme seuil pour reconnaitre ou pas un individu à travers une simple condition if. L'iD retourné nous permet tout simplement de retrouver le nom correspondant à la personne. Si la condition régissant l'indice de confiance n'est pas satisfaite, le nom retourné est 'Inconnu' dans le cas contraire L'ID retourné est utilisé pour retrouver le nom correspondant dans le fichier texte.

```
reco = cv2.face.LBPHFaceRecognizer_create(2,7,7,7)
reco.read("classificateurLBPH.xml")
ID, conf = reco.predict(gris_face)
if conf<=45:

    couleur=couleur_ok
    NOM =Noms[ID-1]
else:
    couleur=couleur_ko
    NOM="Inconnu"
```

Figure 33: Création objet de reconnaissance LBP, lecture du fichier xml et utilisation de la fonction predict

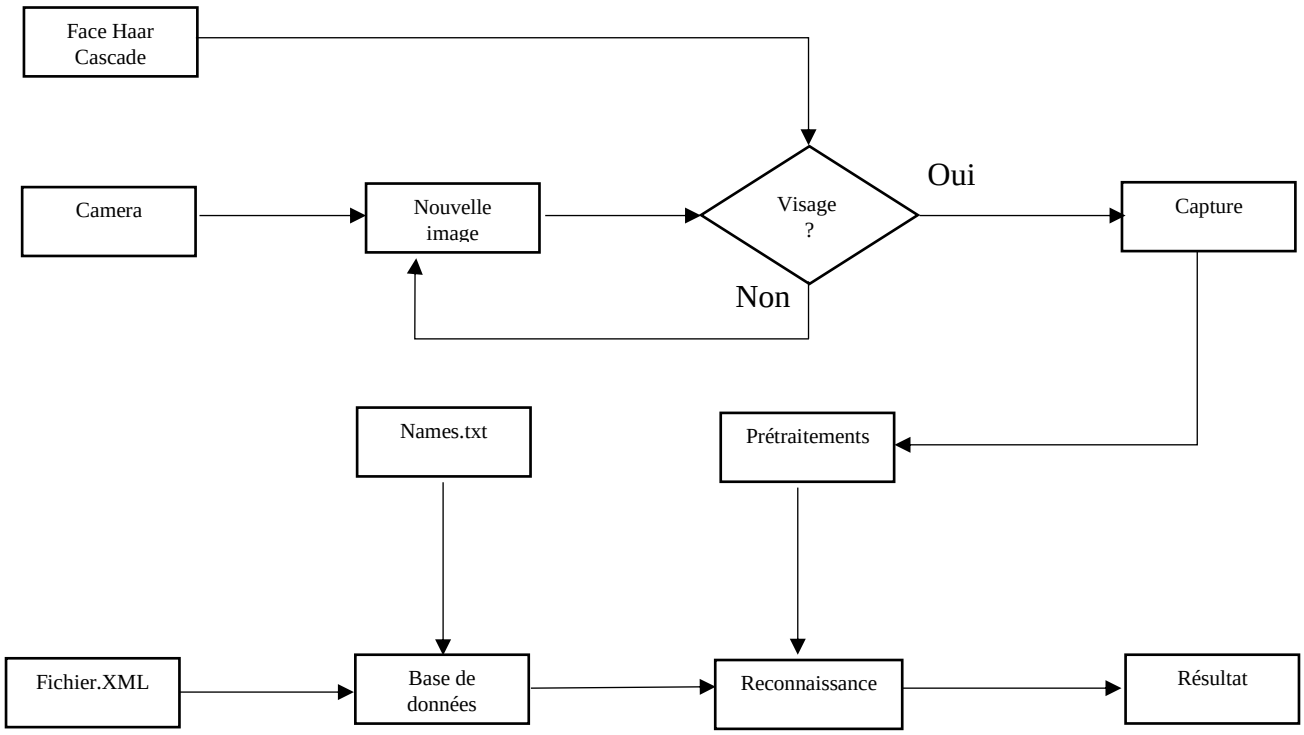

Figure 34: Organigramme de reconnaissance

III.4. Fonctionnalités du programme :

Notre programme effectue :

- **la détection de visages**

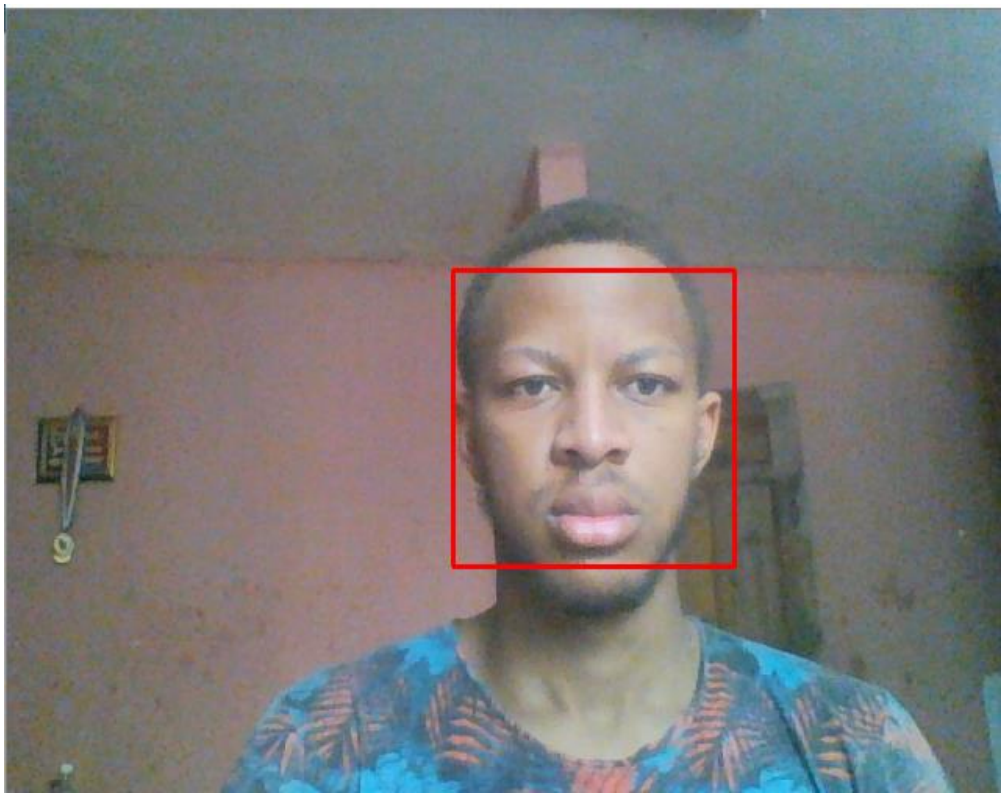

Figure 35: Détection de visages

- **L'identification de visages en temps réel**

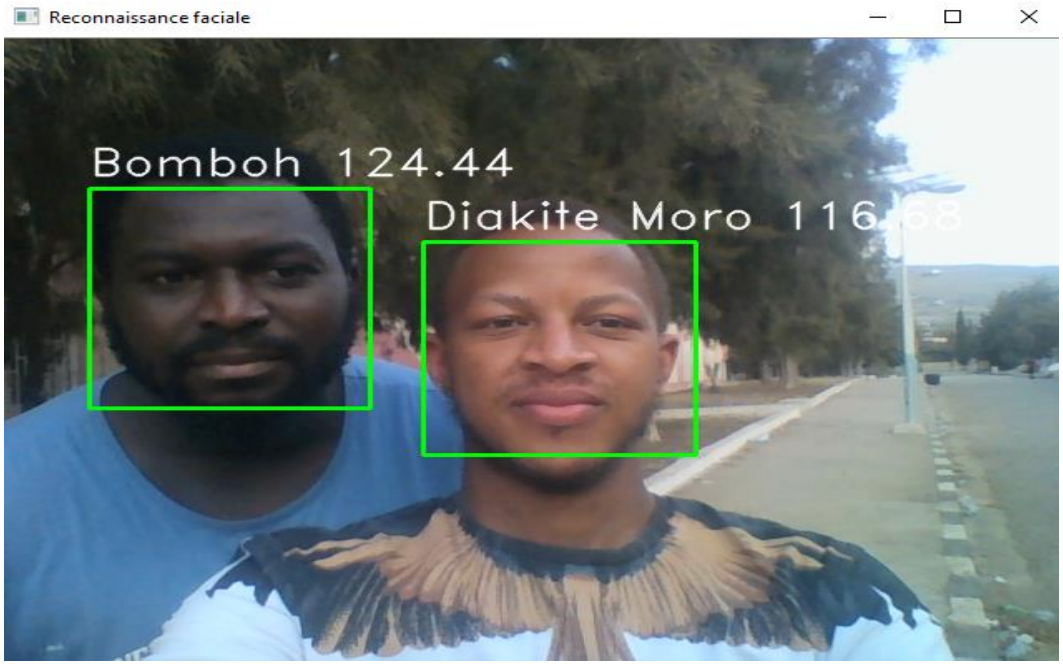

Figure 36: identification de visages en temps réel

- **L'identification de visages en vidéo**

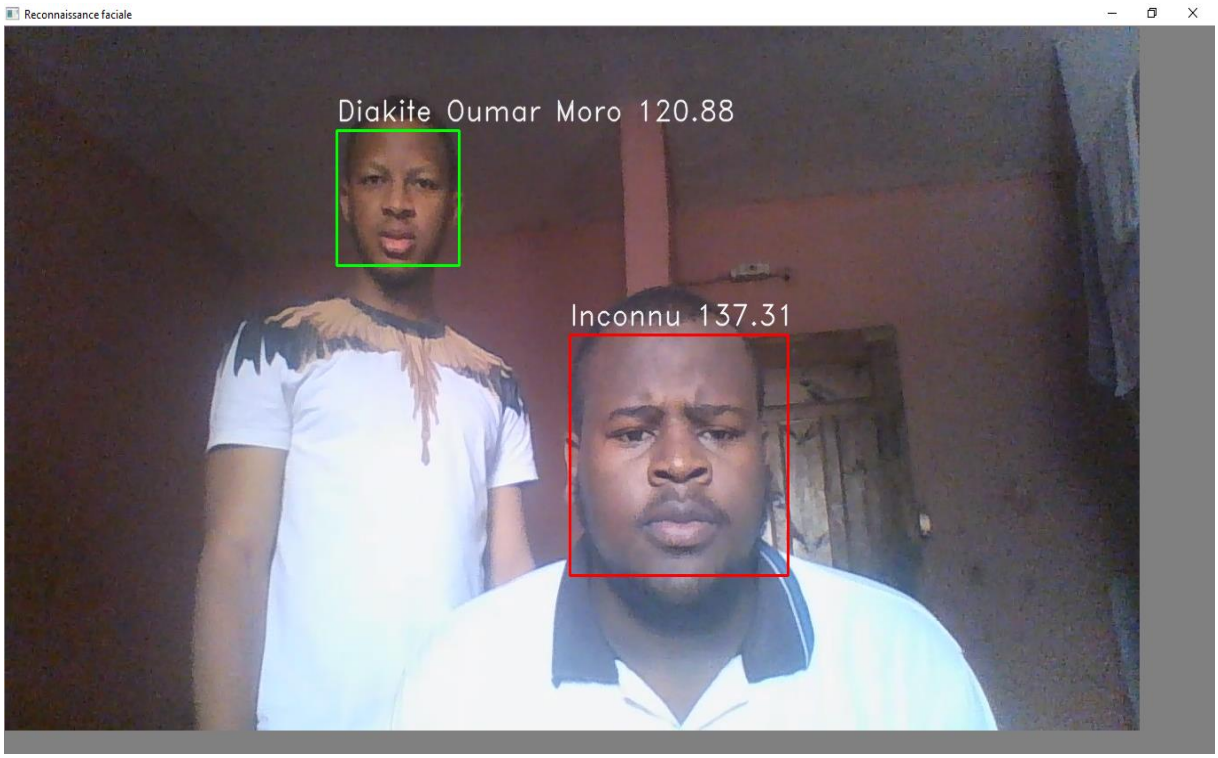

Figure 37: Identification de visages en vidéo

- **L'identification en photos**

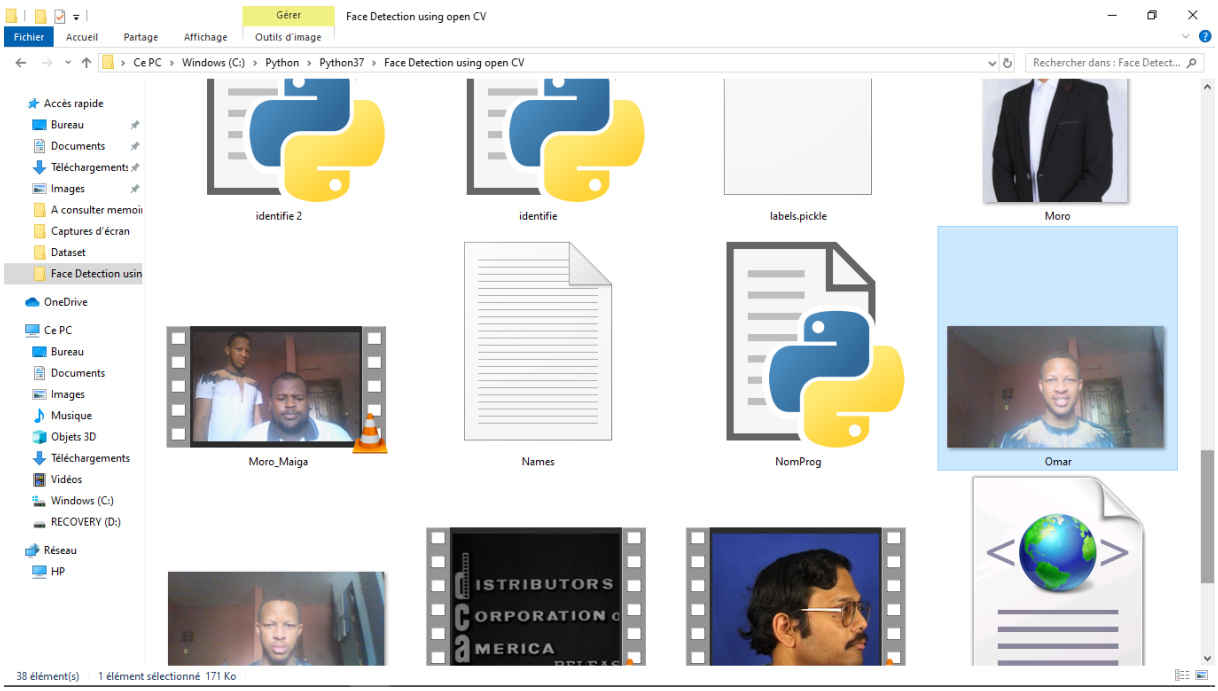

Figure 38: chemin d'accès de la photo à copier dans le sous-programme de reconnaissance

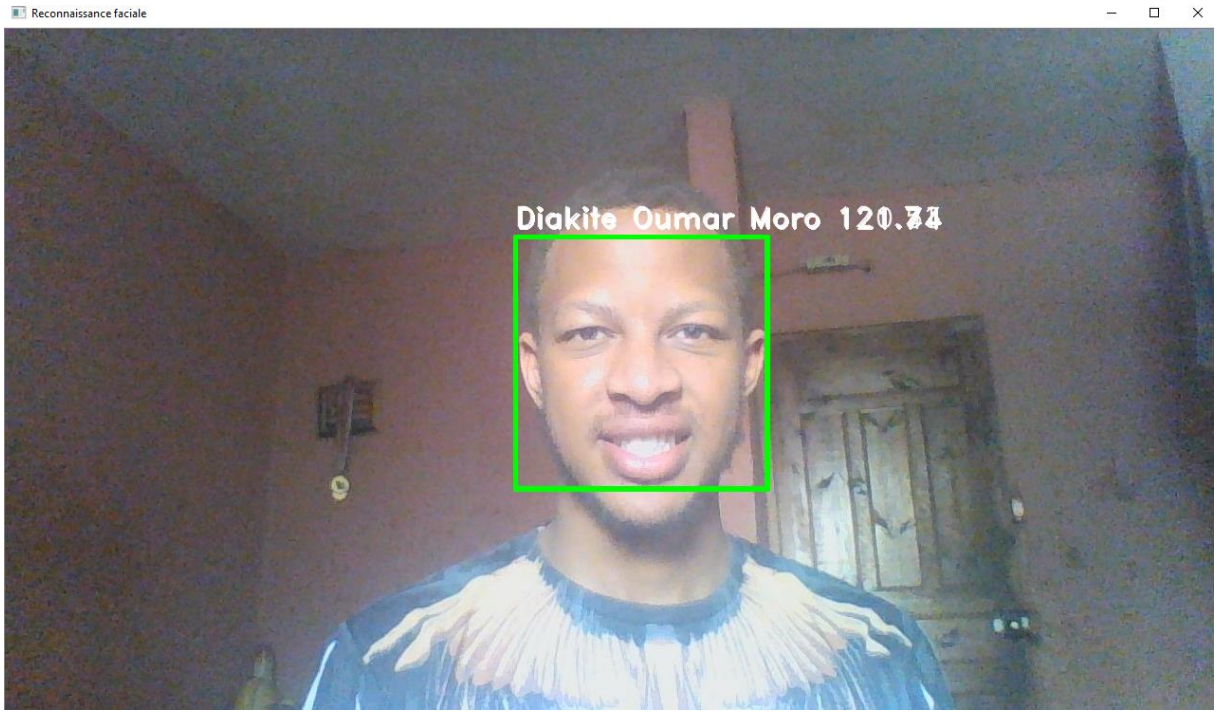

Figure 39: Identification de visages sur une photo

III.5. Tests et résultats :

Nos différents tests, nous ont permis de fixer notre seuil de reconnaissance finalement à 45, ce seuil nous a permis de réduire très considérablement le taux de FAR et FRR.

III.5.1. Performance du système :

Nous avons réalisé à ce jour un total de 28 tests, nous comptons 25 tests corrects et 3 tests incorrects (2 FRR, 1 FAR). Le taux de reconnaissance T du système est donc :

$$T=\frac{25}{28} = 0{,}8928 = 89{,}28\%$$

III.6. Discussion sur les résultats obtenus

La méthode utilisée pour la reconnaissance (LBP) est l'une des plus robustes aux problèmes d'éclairage mais malgré cette robustesse, ce problème persiste toujours. Notre classifieur LBP a nécessité plusieurs images afin de pouvoir bien entrainer le classificateur. Nous estimons que l'acquisition d'image influence considérablement le taux de reconnaissance de notre système, la qualité de la webcam utilisée étant assez basse. Il est aussi à noter la qualité de la base de données, si la base de données d'images contient des images floues ceci entraine une augmentation du taux d'erreur. Avec une performance de 89,28% notre système est assez performant.

a. Points faibles :

Le mauvais éclairage et les grandes variations de pose affectent considérablement l'extraction des caractéristiques du visage. A travers nos tests, nous avons remarqué que les images très petites (visage loin) affectent très négativement sur l'extraction des caractéristiques du visage, ce qui influe tout le système. Aussi l'apprentissage du classificateur ne se fait qu'avec les images issues du sous-programme de détection et d'enregistrement de visages donc de la webcam.

b. Points forts :

La capacité de détection de visages de notre système est très bonne même avec éclairage faible, ou présence de barbe, moustache, lunette. Le système peut faire l'identification à travers une photo, une vidéo, ou en temps réel. Le système est aussi entièrement automatique dans l'extraction des caractéristiques du visage.

Conclusion

Dans ce chapitre nous avons présenté en détail la conception et l'implémentation de notre système. Nous avons aussi procédé à la présentation et discussion des résultats obtenus notamment le taux de reconnaissance du système qui est de 89,28%, un taux qui est améliorable, en effet l'amélioration de l'acquisition d'images et l'utilisation éventuelle d'une approche hybride pour la reconnaissance pourraient améliorer ce taux.

Conclusion générale et perspectives

Au terme de ce travail, nous avons traité une problématique majeure et d'actualité dans le domaine de la sécurité informatique, qui est la mise en œuvre d'un système de reconnaissance faciale, que nous avons réussi à réaliser à travers 3 sous-programmes complémentaires, effectuant respectivement la tache de détection et enregistrements du visage, l'apprentissage du classificateur et la reconnaissance. Un système de reconnaissance automatique passe obligatoirement par ces 3 étapes.

Notre système permet d'identifier une personne avec un taux de reconnaissance assez élevé, et la détection de visages est très satisfaisante, malgré la robustesse à la lumière de l'approche locale utilisée (LBP) pour la reconnaissance, les problèmes d'éclairage demeurent. Néanmoins un apprentissage avec des images nettes, prises sous différentes luminosités corrige un peu ce problème.

Nous insistons sur la qualité des images d'entrainement, car elle influence beaucoup le taux de reconnaissance. Les images d'entrainement sont converties en niveaux de gris rendant la détection et l'extraction de caractéristiques plus aisée.

Les perspectives de ce travail sont nombreuses : nous souhaitons d'abord utiliser une meilleure caméra pour l'acquisition d'images, en vue d'optimiser le taux de reconnaissance. Essayer des approches hybrides afin de résoudre le problème d'éclairage. Améliorer encore plus la robustesse du système, à travers la multi-modalité donc l'association d'une autre donnée biométrique, comme l'empreinte digitale ou géométrie de la main, et enfin l'élaboration d'une interface graphique adaptée.

Notre système peut être utilisé pour sécuriser l'accès à des données et même des édifices. Seules les personnes autorisées, auront accès à la pièce sécurisée ou aux données sécurisées. Il peut aussi servir à la vidéosurveillance à condition d'avoir un appareil d'acquisition d'images d'une grande qualité.

Le perfectionnement étant toujours possible, les systèmes de sécurité s'améliorant de jour en jour, nous restons dans cette optique pour une éventuelle suite de ce travail.

Annexe

Annexe 1 : Installation de python et des différentes bibliothèques

Python 3.7 :

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. 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.

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)

Open cv :

python -m pip install opencv-python

python -m pip install opencv-contrib-python

Numpy :

python -m pip install numpy

PIL :

python -m pip install pil

DLIB :

python -m pip install dlib

Annexe 2 : Quelques sous-programmes utilisés

Sous-programme de détection et d'enregistrement de visages

```
import cv2
import math
import numpy as np
from PIL import Image
Count = 0
min_size=110
face_cascade = cv2.CascadeClassifier("./haarcascade_frontalface_alt2.xml")
glass_cas = cv2.CascadeClassifier('./haarcascade_eye.xml')
def AjoutNom():
    Nom = input('Veuillez entrer votre nom et prenom: ')
    Info = open("Names.txt", "r+")
    ID = ((sum(1 for line in Info))+1)
    Info.write(str(ID) + "," + Nom + "\n")
    print ("Nom enregistrer sous l'ID: " + str(ID))
    Info.close()
    return ID
def DetectEyes(Image):
    Theta = 0
    rows, cols = Image.shape
    glass = glass_cas.detectMultiScale(Image)
    for (sx, sy, sw, sh) in glass:
        if glass.shape[0] == 2:
```

```
        if glass[1][0] > glass[0][0]:
            DY = ((glass[1][1] + glass[1][3] / 2) - (glass[0][1] + glass[0][3] / 2))
             DX = ((glass[1][0] + glass[1][2] / 2) - glass[0][0] + (glass[0][2] / 2))
         else:
            DY = (-(glass[1][1] + glass[1][3] / 2) + (glass[0][1] + glass[0][3] / 2))
            DX = (-(glass[1][0] + glass[1][2] / 2) + glass[0][0] + (glass[0][2] / 2))
        if (DX != 0.0) and (DY != 0.0):
            Theta = math.degrees(math.atan(round(float(DY) / float(DX), 2)))
            M = cv2.getRotationMatrix2D((cols / 2, rows / 2), Theta, 1)
            Image = cv2.warpAffine(Image, M, (cols, rows))
    return Image
ID = AjoutNom()
cap = cv2.VideoCapture(0)
while Count<50:
    ret, frame = cap.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    if np.average(gray)>110:
        face = face_cascade.detectMultiScale(gray, scaleFactor=1.2, minNeighbors=4,
minSize=(min_size, min_size))
        for (x, y, w, h) in face:
            FaceImage = gray[y - int(h / 2): y + int(h * 1.5), x - int(x / 2): x + int(w * 1.5)]
            Img = DetectEyes(FaceImage)
            if Img is  None:
                fd = frame
            else:
                fm= gray[y: y+h, x: x+w]
```

```
            cv2.imwrite("dataSet/User." + str(ID) + "." + str(Count) + ".png", fm)
            cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 0, 255), 2)
            Count = Count + 1
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
    cv2.imshow("Video", frame)
    cv2.waitKey(1)
    for cpt in range(4):
        ret, frame=cap.read()
print ('Capture du visage terminée!')
cap.release()
cv2.destroyAllWindows()
```

Sous-programme d'apprentissage

```
import os
import cv2
import numpy as np
from PIL import Image
min_size=110
LBPHFace = cv2.face.LBPHFaceRecognizer_create(2,7,7,7)
path = 'dataSet'
def ImageId (path):
    imagePaths = [os.path.join(path, f) for f in os.listdir(path)]
    FaceList = []
    IDs = []
    for imagePath in imagePaths:
        faceImage = Image.open(imagePath)
        faceImage = faceImage.resize((min_size, min_size))
        faceNP = np.array(faceImage, 'uint8')
        ID = int(os.path.split(imagePath)[-1].split('.')[1])
        FaceList.append(faceNP)
        IDs.append(ID)
        cv2.imshow('Apprentissage en cours', faceNP)
        cv2.waitKey(1)
    return np.array(IDs), FaceList
IDs, FaceList = ImageId(path)
print('Apprentissage en cours.....')
LBPHFace.train(FaceList, IDs)
```

```
print('Apprentissage du Classifieur complet...')

LBPHFace.save('classificateurLBPH.xml')

print ('Fichier XML enregistré...')

cv2.destroyAllWindows()
```

Sous-programme d'identification

```
import cv2
import math
import numpy as np
from PIL import Image
couleur_ko=(0, 0, 255)
couleur_ok=(0, 255, 0)
couleur_info=(255, 255, 255)
min_size=110
face_cascade = cv2.CascadeClassifier("./haarcascade_frontalface_alt2.xml")
eye_cascade = cv2.CascadeClassifier('./haarcascade_eye.xml')
reco = cv2.face.LBPHFaceRecognizer_create(2,7,7,7)
reco.read("classificateurLBPH.xml")
def LectureFichier():
    Info = open("Names.txt", "r")
    NOM = []
    while (True):
        Ligne = Info.readline()
        if Ligne == '':
            break
        NOM.append (Ligne.split(",")[1].rstrip())
    return NOM
Noms = LectureFichier()
a=input("Entrez 0 pour identifier à partiir d'une photo, 1 pour une video ou temps reel:")
a=int(a)
```

```
if a==0:
    photo= cv2.imread("User.8.69.png")
    gris = cv2.cvtColor(photo, cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gris, scaleFactor=1.2,minNeighbors=4, minSize=(min_size, min_size))
    for (x, y, w, h) in faces:
        gris_face = gris[y: y+h, x: x+w]
        eyes = eye_cascade.detectMultiScale(gris_face)
        for (ex, ey, ew, eh) in eyes:
            ID, conf = reco.predict(gris_face)
            if conf<=45:
                couleur=couleur_ok
                NOM =Noms[ID-1]
            else:
                couleur=couleur_ko
                NOM="Inconnu"
            label=NOM+" "+'{:5.2f}'.format(conf)
            cv2.putText(photo, label, (x, y-10), cv2.FONT_HERSHEY_DUPLEX, 1, couleur_info, 1, cv2.LINE_AA)
            cv2.rectangle(photo, (x, y), (x+w, y+h), couleur, 2)
    cv2.imshow('Reconnaissance faciale', photo)
    cv2.waitKey(0)
else:
    cap = cv2.VideoCapture(0)
    # Pour une vidéo remplacer le 0 dans la fonction par le chemin d'accès de la video ainsi que son type cv2.Videocapture("M2RT.mp4")
    while True:
```

```
    ret, img = cap.read()
    gris = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gris, scaleFactor=1.2,minNeighbors=4, minSize=(min_size, min_size))
    for (x, y, w, h) in faces:
        gris_face = gris[y: y+h, x: x+w]
        eyes = eye_cascade.detectMultiScale(gris_face)
        for (ex, ey, ew, eh) in eyes:
            ID, conf = reco.predict(gris_face)
            if conf<=45:
                couleur=couleur_ok
                NOM =Noms[ID-1]
            else:
                couleur=couleur_ko
                NOM="Inconnu"
            label=NOM+" "+'{:5.2f}'.format(conf)
            cv2.putText(img, label, (x, y-10), cv2.FONT_HERSHEY_DUPLEX, 1, couleur_info, 1, cv2.LINE_AA)
            cv2.rectangle(img, (x, y), (x+w, y+h), couleur, 2)
    cv2.imshow('Reconnaissance faciale', img)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        cap.release()
        break
cv2.destroyAllWindows()
```

Bibliographie

[1] https://www.larousse.fr/encyclopedie/divers/biom%C3%A9trie/27110 09/06/2020

[2] Boudjellal Sofiane «Détection et identification de personne par méthode biométrique », Mémoire de Magister, Université de Tizi-ouzou, 2010

[3] https://www.lebigdata.fr/donnees-biometriques-definition-securite 09/06/2020

[4] Mohamad El-Abed « Évaluation de système biométrique», Thèse de Doctorat, Université de Caen, 2011

[5] Peter Meenen et Reza Adhami, « Fingerprinting for Security », IEEE potentials, Vol. 20, No. 3, pp. 33-38, 2001.

[6] Benabdi Mouad « IDENTIFICATION DES PERSONNES PAR LES EMPREINTES D'ARTICULATION DES DOIGTS ET LE DEEP LEARNING », Mémoire de Master, Université de M'sila, 2019

[7] Karabenta Alpha Aboubacar Sidiki et Maiga Hamidou, « Conception d'un logiciel d'authentification des individus par reconnaissance faciale », Mémoire de Master, Université de Mostaganem, 2019

[8] Mébarka Belahcene «Authentification et identification en biométrie», Thèse de Doctorat, Université de Biskra, 2013

[9] Meramria Nabila «Reconnaissance de visages par Analyse Discriminante Linéaire (LDA) », Mémoire de Master, Université de Mostaganem, 2019

[10] https://www.rts.ch/decouverte/sciences-et-environnement/technologies/4641430-comment-fonctionnent-les-logiciels-de-reconnaissance-faciale-.html 20/06/2020

[11] https://fr.wikipedia.org/wiki/M%C3%A9thode_de_Viola_et_Jones 20/06/2020

[12] Laura Sánchez López « Local Binary Patterns applied to Face Detection and Recognition», Mémoire de Master, Université Polytechnique de Catalogne, 2010

[13] Boukerrouche Youssouf et Zerriouh Ahmed, «Mise au point d'une application de détection et reconnaissance faciale », Mémoire de Master, Université de Tlemcen, 2018

[14] Timo Ahonen, Abdenour Hadid et Matti Pietikäinen, «Face Recognition with Local Binary Patterns», ECCV, pp. 469–481, 2004

[15] Phung Van Doanh « RECONNAISSANCE DE VISAGES EN UTILISANT LE DESCRIPTEUR POEM (Patterns of Oriented Edge Magnitudes)», Mémoire de Fin d'Etudes, Institut National Polytechnique de Grenoble, 2010.

[16] Mathieu Van Wambeke « Reconnaissance et suivi de visages et implémentation en robotique temps-réel », Mémoire de Master, Université de Louvain, 2010.

[17] Matthew Turk et Alex Pentland, «Eigenfaces for recognition», Journal of Cognitive Neuroscience, Vol. 3, No. 1, pp. 71–86, 1991.

[18] Paul Viola et Michael Jones, « Rapid Object Detection using a Boosted Cascade of Simple Features », IEEE CVPR, 2001

[19] https://openclassrooms.com/fr/courses/1490316-introduction-a-la-vision-par-ordinateur/1490412-avant-de-commencer 02/07/2020