

الجمهورية الجزائرية الديمقراطية الشعبية

République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la recherche scientifique



Université Mohamed Khider Biskra
Faculté des sciences et de la Technologie
Département de Génie Electrique
Filière : électronique
Option : communication

Réf:.....

Mémoire de Fin d'Etudes
En vue de Préparation de diplôme:
D'Ingénieur d'état en Electronique

Thème :

***Reconnaissance de visage
par méthode
hybride PCA2D ADL2D***

Présenté par :

BEN TOUMI Adel

BENBOUZID AbdElHak

Proposé et Dirigé Par:

Mme **BENATIA BELAHCENE** Mébarka.

Promotion : 2009 /2010

الجمهورية الجزائرية الديمقراطية الشعبية
République Algérienne Démocratique et Populaire
Ministère de l'enseignement Supérieur et de la recherche scientifique



Université Mohamed Khider Biskra
Faculté des sciences et de la Technologie
Département de Génie Electrique
Filière : électronique
Option : communication

Mémoire de Fin d'Etudes
En vue de Préparation de diplôme:
D'Ingénieur d'état en Electronique

Thème :

**Reconnaissance de visage par méthode hybride
PCA2D ADL2D**

Présenté par :

.....
.....
.....

Avis favorable du président du Jury

Nom et prénom

.....
.....

Signature

.....

Cachet et signature



**Résumé du mémoire de fin d'étude
En vue de Préparation de diplôme:**

D'Ingénieur d'état en Electronique

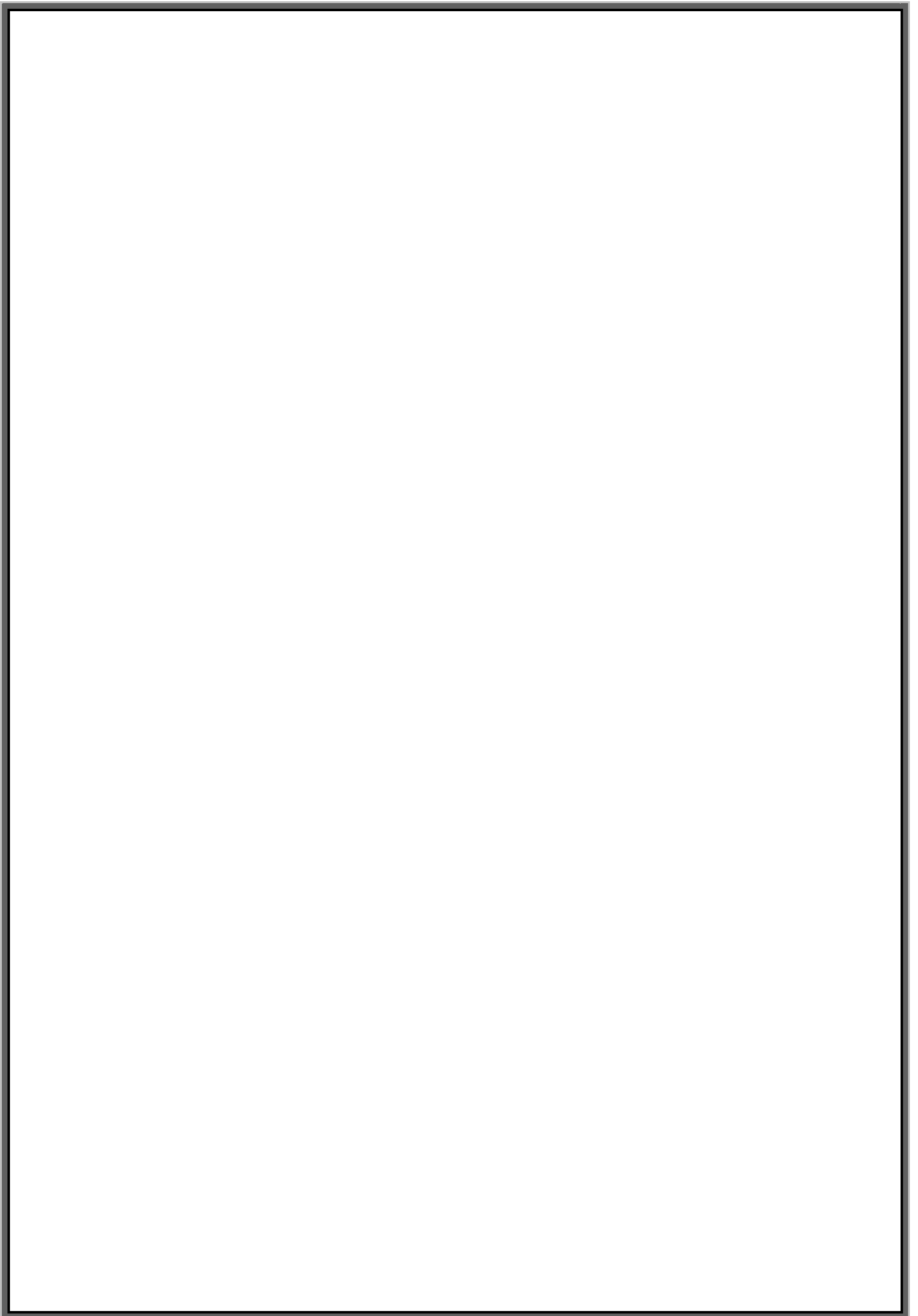
Option : communication

Thème :

**Reconnaissance de visage par méthode hybride
PCA2D ADL2D**

Proposé par : Mme BENATIA BELAHCENE Mébarka.

Dirigé par : Mme BENATIA BELAHCENE Mébarka.



Dédicace

A tous ceux que j'aime

A tous ceux qui m'aiment

Et surtout à tous ceux qui le méritent

SOMMAIRE

Introduction générale.....	I
-----------------------------------	----------

CHAPITRE I RECONAISSANCE DE VISAGE

I .1 Introduction.....	1
-------------------------------	----------

I .2 La biométrie.....	1
-------------------------------	----------

I.2.1 Définition	1
------------------------	---

I .2.2 Présentation des modalités biométriques.....	2
-----------------------------------------------------	---

I.3 La reconnaissance de visage	4
----------------------------------------------	----------

I.4 Identifier ou authentifier ?	5
-----------------------------------------------	----------

I.5 Le processus de reconnaissance de visages.....	6
-----------------------------------------------------------	----------

I.6 Mesure de qualité.....	8
-----------------------------------	----------

I.7 Les classes des techniques de reconnaissance de visages	9
--------------------------------------------------------------------------	----------

I.7.1 Les Méthodes locales (géométriques).....	9
------------------------------------------------	---

I.7.2 Les Méthodes Globales (holistiques).....	11
------------------------------------------------	----

I.7.3 Méthodes hybrides.....	13
------------------------------	----

I.8 Conclusion	14
-----------------------------	-----------

CHAPITRE II

LES METHODES DE RECONAISSANCE DE VISAGE

II.1 Introduction.....	15
-------------------------------	-----------

II.2 Les méthodes de reconnaissance de visage.....	15
-----------------------------------------------------------	-----------

II.2.1 Le réseau de neurones (LVQ).....	15
-----------------------------------------	----

II.2.2 Analyse en Composantes Principales (PCA).....	16
------------------------------------------------------	----

II.2.3 Analyse en Composantes Indépendantes (ACI).....	22
--------------------------------------------------------	----

II.2.4 Analyse Discriminante Linéaire (LDA).....	25
--------------------------------------------------	----

II.3 Conclusion.....	29
-----------------------------	-----------

CHAPITRE III	CONCEPTION DU MODELE
III.1 Introduction	30
III.2 Prétraitement	30
III.2.1 La transformée de Hough.....	31
III.3 Méthodes bidimensionnelles d'extraction de caractéristiques ...	32
III.3.1 L'analyse en composantes principales bidimensionnelle (ACP2D).....	32
III 3.1.1 Principe.....	32
III 3.1.2 Choix du nombre de vecteurs propres.....	34
III.3.2 L'analyse discriminante linéaire bidimensionnelle (ADL2D).....	35
III 3.2.1 Principe.....	35
III 3.2.2 L'analyse discriminante linéaire bidimensionnelle orientée en lignes (ADL2DoL).....	35
III 3.2.3 L'analyse discriminante linéaire bidimensionnelle orientée en colonnes (ADL2DoC).....	36
III.4 Supériorité de l'ACP2D et de l'ADL2D orientée	37
III.5 Inconvénients de l'ACP2D et de l'ADL2D orientée	38
III.6 Conclusion	39

CHPITRE IV	ALGORITHMES ET RESULTATS
IV.1 Introduction	40
IV.2 Généralité sur les bases de données existantes	40
IV.2.1 La base XM2VTS.....	42
IV.2.1.1 Le protocole de Lausanne	43
IV.3 Architecture structurelle d'un système d'authentification	46
IV.3.1 Prétraitement d'image	47
a. Découpage des images et conversion Couleur→ NG.....	47
b. Filtrage.....	48
IV .3.2 Les descripteurs de l'image (Extraction de paramètres).....	48
IV.3.2.1 Transformée de Hough.....	49
IV.3.2.2 Algorithmes de la transformée de Hough.....	49

IV.3.2.3 Photo normalisation.....	51
IV.4 Création de la matrice de données d'apprentissage.....	52
IV.4.1. L'analyse en composantes principales bidimensionnelle (ACP2D).....	52
IV.4.2 L'analyse discriminante linéaire bidimensionnelle orientée en colonnes (ADL2DoC).....	53
IV.4.3 Projection des images.....	55
IV.4.4 Calcul des distances.....	55
IV.4.5 Classification	56
IV.4.6 Décision.....	58
IV.5 Résultats.....	59
IV.5.1 Extraction des paramètres avec l'ACP2D ET ADL2DOC.....	59
Choix du nombre de paramétriser et de la distance.....	60
1) Cas $N_p(\text{ACP2D})=60$	60
2) Cas $N_p(\text{ACP2D})=80$	64
3) Cas $N_p(\text{ACP2D})=100$	66
4) Cas $N_p(\text{ACP2D})=200$	68
5) Récapitulation.....	70
IV.5.2 Le test du système de reconnaissance de visage.....	72
IV.5.3 Taux de réduction de capacité de BDD.....	73
IV.5.3.1 Taux de réduction de capacité pour LDA 1 seul dimension.....	73
IV.5.3.2 Taux de réduction de capacité pour PCA2D LDA2D	74
IV .6 Conclusion.....	74
Conclusion générale.....	II
Bibliographie	III
Annexe.....	IV

Liste des figures

Figure I.1 Les étapes de reconnaissance de visage.....	5
Figure I. 2: L'identification.....	5
Figure I.3 : L'authentification.....	5
Figure I.4 : Le schéma général d'un système de reconnaissance de visages.....	6
Figure I.5 : Les TFA, TFR et TEE en fonction du laxisme du système.....	8
Figure I.6 : La position de TEE de TFR en fonction du TFA.....	9
Figure I.7 : Harmonie faciale.....	11
Figure II.1: Passage d'une image vers un vecteur dans un espace vectoriel de grande dimension.....	17
Figure II.2: Image moyenne et les 15èmes eigenfaces.....	20
Figure II.3 Une version simplifiée de E_v illustrant les quatre résultats de la projection d'une image sur E_v	21
Figure II.4: Le modèle de synthèse d'image de l'architecture 1 d'ICA.....	23
Figure II.5: La représentation d'ICA = (b_1, b_2, \dots, b_n)	23
Figure II.6: Les deux architectures d'ICA	24
Figure II.7: Le modèle de synthèse d'image de l'architecture 2 d'ICA	24
Figure II.8: La représentation factorielle d'ICA = (u_1, u_2, \dots, u_n)	25
Figure II.9: Illustration du principe de séparation optimale des classes par le LDA..	26
Figure III.1: Transformée de Hough (a) Plan cartésien (x, y) (b) Plan des paramètres (a, b).....	31
Figure III.2: Une représentation d'une droite dans l'espace de Hough.....	32
Figure III.3: Comparaison entre les images reconstruites par : eigenfaces (1 ^{ère} ligne), ADL2DoC (2 ^{ème} ligne) et ACP2D (3 ^{ème} ligne)	37
Figure III.4: Décroissance des valeurs propres pour l'ACP2D et l'ADL2DoC.....	38
Figure IV.1: Exemple d'images extraites de la base XM2VTS.....	45
Figure IV.2: Exemples des images d'une personne de la base de données XM2VTS.....	45

Figure IV.3: Organigramme du Système de Vérification d'identité à l'aide d'images de Visages.....	46
Figure IV.4 : Exemple de découpage d'une image.....	47
Figure IV.5 : L'effet de seuil dans chaque filtre.....	48
Figure IV.6 : Organigramme de la Transformée de Hough	50
Figure IV.7 : Taux d'égale erreur dans l'ensemble d'évaluation pour différentes fonctions de mesure de score Cas $N_p(ACP2D)=60$	63
Figure IV.8 : Taux d'égale erreur dans l'ensemble d'évaluation pour différentes fonctions de mesure de score Cas $N_p(ACP2D)=80$	65
Figure IV.9 : Taux d'égale erreur dans l'ensemble d'évaluation pour différentes fonctions de mesure de score Cas $N_p(ACP2D)=100$	67
Figure IV.10 : Taux d'égale erreur dans l'ensemble d'évaluation pour différentes fonctions de mesure de score Cas $N_p(ACP2D)=200$	69
Figure IV.11 : Taux d'égale erreur dans l'ensemble d'évaluation pour différentes fonctions de mesure de score ACP2D varie et ADL2DOC varie.....	70
Figure IV.12 : Taux d'égale erreur dans l'ensemble d'évaluation et l'ensemble de test.....	73

Liste des algorithmes

Algorithme IV.1 : L'algorithme de la fonction de prétraitement pour la TH.....	51
Algorithme IV.2 : Algorithme de la fonction ACP2D	53
Algorithme IV.3 : Algorithme de la fonction ADL2DoC ()	54
Algorithme IV.4 : Algorithme de calcul de TFA, TFR et seuillage.....	57
Algorithme IV.5 : Algorithme de test (décision).....	58

Liste des tableaux

Tableau I.1 : Comparaison des méthodes basées sur les caractéristiques locales ou globales.....	13
Tableau II.1 :Les quatre possibilités qui apparaissent lors de la phase de reconnaissance.....	22
Tableau IV.1 : Principales caractéristiques des bases de visages.....	41
Tableau IV.2 : Répartition des images de la base de données selon la Configuration I.....	43
Tableau IV.3 : Répartition des photos dans les différents ensembles.....	44
Tableau IV.4 : Nombre de comparaisons possibles.....	44
Tableau IV.5 : le tableau des paramètres ACP2D et ADL2DOC	60
Tableau IV.6 : Le tableau total de ACP2D=60 et ADL2DOC= [10 20 30 40 50 60]	61
Tableau IV.7 : Le tableau total de ACP2D=80 et ADL2DOC= [10 30 40 50 60 70]	64
Tableau IV.8 : Le tableau total de ACP2D=100 et ADL2DOC= [10 20 30 50 80 90]	66
Tableau IV.9 : Le tableau total de ACP2D=200 et ADL2DOC= [10 30 50 80 120 150]	68
Tableau IV.10 : comparaison entre l'ensemble d'évaluation et l'ensemble de test.....	72

List de symbole

FR : (false rejection) Rejeter faussement un client.

FA : (false acceptation) Accepter faussement un imposteur.

TFA :Taux rejeter faussement un client.

TFR : Taux accepter faussement un imposteur.

TEE : Taux total.

ρ_k : Pas de quantification de dimensions de ρ .

θ_k : Pas de quantification de dimensions de θ .

std (x) : l'écart type de la variable **x**.

Y : Image d'entrée (à traiter).

X : Vecteurs images de l'ensemble d'apprentissage.

G_t : La matrice de covariance.

Vprop : Valeurs propres Vprop.

U_{ACP2D} : la base orthonormée.

\bar{X}_c : Matrice moyenne de la classe **c**.

\bar{X} : Matrice moyenne totale (de toutes les matrices de la base d'apprentissage).

X_j : La $j^{\text{ème}}$ image de la base d'apprentissage.

S_b : Matrice de covariance inter-classes.

S_w : Matrice de covariance intra-classes données respectivement.

σ_A : L'écart type de **A**.

μ_A : Le moyenne de **A**.

σ_B : L'écart type de **B**.

μ_B : Le moyenne de **B**.

imtest : Image de test (image de l'utilisateur).

DIS : Distance entre **X** et **Y**. (projetées dans le même sous espace) en utilisant l'une des normes de mesure de similarité.

Np : Nombre de paramètre.

BDD : Base de donne.

ICA: Analyse en Composantes Indépendantes.

LVQ: Le réseau de neurone.

PCA: Analyse en Composantes Principales.

LDA: Analyse Discriminante Linéaire.

XM2VTS : base de données outil.

ACP2D : Analyse en Composantes Principales bidimensionnelle

ADL2D: Analyse Discriminante Linéaire bidimensionnelle.

RESUME

La reconnaissance de visages est une technologie biométrique en vogue, elle est très utilisée dans les applications de contrôle d'accès. Dans la littérature, on trouve plusieurs méthodes globales, locales et hybrides de reconnaissance de visage. Le but de ce projet est d'introduire et d'expliquer l'approche hybride de reconnaissance à savoir : **ACP2D ADL2D**. L'originalité de notre méthode réside dans le fait qu'elle soit hybride. Nous avons tenté d'associer deux méthodes de natures différentes la transformation de Hough (méthodes de morphologie mathématique) et l'ACP LDA (méthodes statistiques). Notre approche assure une reconnaissance satisfaisante et offre une représentation de visages très compacte vis-à-vis l'ACP1D et l'ADL1D. Des tests ont été effectués sur une base de données universelle la **XM2VTS** et les résultats obtenus sont très encourageants. Le système de reconnaissance de visage a été testé et est bien stable.

Mots clés : Biométrie, Transformation de Hough, **ACP2D, ADL2D**, Similarité, **TFA, TFR, TR, TEE**

Introduction générale

La reconnaissance de visages humains est considérée actuellement comme l'une des tâches les plus importantes pour relever le défi en reconnaissance des formes. La facilité et la précision avec lesquelles nous pouvons identifier nos amis et nos ennemis, même dans des conditions défavorables, sont des capacités des plus étonnantes du système visuel humain, le but de la reconnaissance de visages est de concevoir des systèmes informatiques capables d'égaliser les êtres humains dans ce domaine.

S'investir dans le domaine de la reconnaissance de visages est sans doute motivé par la multiplicité et la variété des champs d'applications de celui-ci (télésurveillance et vérification d'identités, chirurgie plastique, langage des signes...). L'intérêt pour ce domaine s'explique aussi par le fait que par rapport aux autres filières de la reconnaissance des formes (reconnaissance de l'écriture, de la voix, etc...), la reconnaissance de visages n'a pas atteint un niveau aussi avancé que ces autres filières. Ceci s'explique avant tout par la complexité de la forme en question (le visage) comparée avec les autres formes.

Dans le cadre de notre travail nous présentons une nouvelle méthode pour la reconnaissance de visage est l'hybridation entre l'Analyse en Composantes Principales Bidimensionnelle (**ACP2D**) et l'Analyse Discriminante Linéaire Bidimensionnelle(**ADL2D**).la partie de prétraitement de cette méthode est basée sur la transformation de Hough. Notre approche est basée sur l'approche Eigens Faces (EF) ou les visages propres.

Nous avons choisi de présenter notre étude autour de **quatre chapitres** principaux:

➤ Le **premier chapitre** est consacré à la définition de la biométrie et la présentation des différentes modalités biométriques comme les empreintes digitales, la main, l'œil, la voix, la signature et la reconnaissance de visage. Nous présentons aussi le processus de reconnaissance de visages et la différence entre l'identification et l'authentification. Nous terminons le chapitre par les performances du système de vérification biométrique.

➤ Dans le **deuxième chapitre** nous présentons les algorithmes classiques de reconnaissance de visage comme le réseau de neurones (**LVQ**), l'Analyse en Composantes Principales (**ACP**), l'Analyse en Composantes Indépendantes (**ACI**) et l'Analyse Discriminante Linéaire (**ALD**).

➤ L'étude de notre modèle est présenté dans le **troisième chapitre** où une

brève présentation de la transformation de **Hough** est effectuée. Celle-ci est utilisée dans la phase de prétraitement pour la détection de contours du visage et l'élimination des basses fréquences. Ensuite, nous détaillons les différentes approches de reconnaissance faciale que nous avons élaborées, à savoir : l'approche basée sur ACP bidimensionnelle (**ACP2D**) et l'approche basée sur ALD bidimensionnelle (**ADL2D**) ayant pour but d'assurer une bonne séparation entre classes pour améliorer la performance d'un système de reconnaissance de visage.

➤ Dans le **quatrième chapitre** nous présentons l'implémentation du système de reconnaissance étudié, la base de données universelle la **XM2VTS** est définie. Tous les organigrammes des différentes étapes du système d'identification basé sur l'Approche Bidimensionnelle Hybride **ACP2D** et **ADL2D** sont réalisés. Nous présentons d'abord la partie **prétraitement** basée sur la méthode de transformation de Hough , puis nous développons la partie d'extraction de paramètres par L'**ACP2D** associée à la **LDA2D** suivie d'une classification par une mesure de similarité en utilisant diverses **distances métriques**. Les résultats obtenus pour les différents filtres et distances sont présentés à la fin du chapitre IV. Et, finalement de le meilleur score est retenu pour un filtre et une distance bien choisis.

Notre mémoire se termine par une conclusion générale où l'essentiel de nos résultats est présenté et commenté, ainsi que l'évolution future du travail que nous jugeons la meilleure.

I.1 Introduction

Dans la vie de tous les jours une personne doit être identifiée dans de multiples situations : entrée d'un immeuble, accès au travail, retrait d'argent, paiement par cartes bancaires, connexion à son ordinateur et à sa messagerie, entraînant ainsi la mémorisation obligatoire de nombreux codes. C'est pourquoi aujourd'hui de nombreuses recherches visent à prouver l'identité d'une personne en utilisant des caractéristiques biologiques ou comportementales qui lui sont propres : iris, empreinte digitale, forme de la main, frappe sur un clavier, vitesse de signature...

I.2 La biométrie

En 1775, le pasteur suisse, Jean-Gaspard Lavater, rédige un essai sur la physiognomonie. Il propose d'identifier le caractère d'un individu à partir de l'observation de la mobilité de son visage. Quelques années plus tard, le médecin allemand, François-Joseph Gall, s'oppose à Lavater. Ce n'est plus le visage qu'il faut observer mais le crâne qu'il faut tâter pour déterminer la personnalité d'un individu. En effet, ce crâne serait, selon lui, modelé par la forme du cerveau en fonction de la personnalité. La fameuse "bosse des maths" est née. C'est la phrénologie. Dans la même lignée, le médecin italien, Cesare Lombroso, propose en 1885 sa théorie du criminel-né. Selon lui, le crime est le fait d'individus constitutionnellement voués à cela, que l'on peut repérer par des stigmates physiques ou morphologiques dont il suffit d'établir l'inventaire. Dès lors, des millions d'observations chiffrées seront recueillies sur les criminels dans toute l'Europe [CSI05].

I.2.1 Définition

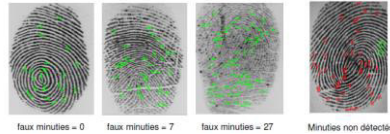
La biométrie est la science qui regroupe l'ensemble de ces travaux et dispositifs, c'est le système d'identification et de protection le plus fort, loin devant les mots de passe et les clés matérielles - du type carte à puce. Si l'on combine plusieurs solutions de biométrie de haut de gamme, on arrive même à un niveau de protection inédit et difficile à déjouer. La biométrie n'est jamais aussi efficace que lorsqu'elle fonctionne de façon redondante - un système de reconnaissance d'empreintes couplé à la reconnaissance d'iris par exemple. C'est pourquoi nous nous intéressons aux différentes technologies existantes, les nouvelles technologies de la surveillance.

I.2.2 Présentation des modalités biométriques

a. Les empreintes digitales

Analyse des détails caractéristiques des sillons cutanés, ou « minuties » : terminaison des sillons, croisements, bifurcations, etc.

Reconnaissance des empreintes digitales est de 50 % et c'est la plus **ancienne** et la plus répandue des technologies d'identification utilisée par la CIA depuis les **années 60** [MAR03].



b. La main

On analyse la longueur, largeur, forme des phalanges, des articulations, des lignes de la main, etc. La reconnaissance de la main représente les 10 % de la reconnaissance biométrique. Elle s'appuie sur une image en 3 dimensions : quelques traits caractéristiques sont gardés en mémoire - la taille et la largeur des doigts, l'espace entre les différentes parties de la main. Cette technologie présente l'avantage d'être moins capricieuse que la reconnaissance d'empreintes digitales : la saleté et les petites coupures n'empêcheront pas le système de fonctionner [MAR03].



c. L'œil (rétine ou iris)

C'est l'analyse de la disposition des muscles circulaires et radiaux qui ouvrent et ferment la pupille. La reconnaissance de l'iris est de 6 % c'est le 'fleuron' de la biométrie : notre iris est unique et extrêmement complexe. On peut distinguer jusqu'à 244 points de comparaison, et le taux d'erreur des produits disponibles sur le marché est proche de 0 % [MAR03].



Ici aussi, il faut opter de préférence pour un produit haut de gamme : les systèmes de reconnaissance d'iris les moins évolués peuvent être trompés par une image ou une lentille reproduisant le dessin de l'iris d'une personne. Les systèmes les

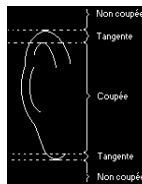
plus évolués disposent de techniques qui leur permettent de déjouer ces reproductions notamment en contrôlant que l'iris change bien de taille avec l'intensité de la lumière.

d. La voix

Analyse des caractéristiques quantitatives : fréquences, harmoniques, puissance sonore, etc.

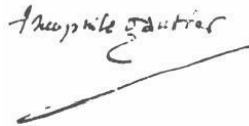


e. L'oreille C'est l'analyse de la forme de l'oreille.



f. La signature (reconnaissance statique ou dynamique)

On réalise l'analyse de la forme (statique) et/ou de la vitesse et de la trajectoire de la signature (dynamique).



g. L'ADN

La méthode la plus fiable pour identifier une personne, mais actuellement pas adaptée à la reconnaissance en temps réel.

h. Reconnaissance faciale

La reconnaissance faciale représente les 15 % du marché mais elle est classée la deuxième après les empreintes ; c'est la technologie qui monte. Elle s'appuie sur les caractéristiques principales du visage - l'éloignement des yeux, la taille de la bouche, etc... pour construire une carte du faciès. La reconnaissance faciale est capable de déjouer le port d'une barbe ou la teinte des cheveux : seule une opération chirurgicale intervenant sur le cartilage est en mesure de la tromper à coup sûr. On distingue deux technologies : la reconnaissance de visage fixe et mobile. L'identification d'un sujet fixe dans de bonnes conditions de luminosité peut-être considérée comme mûre. Pour ce qui est de la reconnaissance d'un visage sur une vidéo, on doit encore déplorer un taux d'échec assez élevé. Un défaut qui n'empêche pas le succès de cette technologie :

c'est pour l'instant la seule solution que l'on peut **utiliser sans** obtenir le **consentement de la personne identifiée** [MAR03].

I.3 La reconnaissance de visage

C'est l'analyse de la géométrie du visage de face et profil, ou en trois dimensions: forme des yeux, de la bouche, du nez, position des pommettes etc.... à partir d'une photographie numérique ou d'une caméra infrarouge (thermographie pour utilisation dans le noir), celle-ci permet l'analyse des vaisseaux sanguins du visage. [MOHO2].

La reconnaissance automatique de visage humain basée sur le traitement des images 2D s'est bien développée ces dernières années, et plusieurs techniques ont été proposées. Malgré les résultats obtenus dans ce domaine, la reconnaissance robuste de visage reste un problème très difficile. Les méthodes actuelles sont efficaces lorsque les conditions de prise de vue des images tests sont similaires à celles des images d'apprentissage. Cependant, la grande variabilité générée par le changement de luminosité et le changement de prise de vue causent de sérieux problèmes pour de nombreux systèmes de reconnaissance existants.

Pourquoi utilisons-nous la technologie de reconnaissance de visage?

Cette technologie nous aide à éviter qu'une personne puisse avoir deux cartes d'identité ou usurper l'identité d'une autre personne.

Comment fonctionne la technologie de reconnaissance de visage?

Lorsque vous demandez n'importe quelle sorte de permis de conduire ou de carte d'identité. La technologie de reconnaissance faciale est ensuite utilisée pour mesurer les caractéristiques particulières de votre visage afin de créer une équation mathématique. Le système compare ensuite cette équation aux équations stockées dans la base de données. Notre équipe de gestion de l'identité peut effectuer des comparaisons faciales complémentaires pour les visages dont les caractéristiques sont trop proches pour que le logiciel puisse les distinguer.

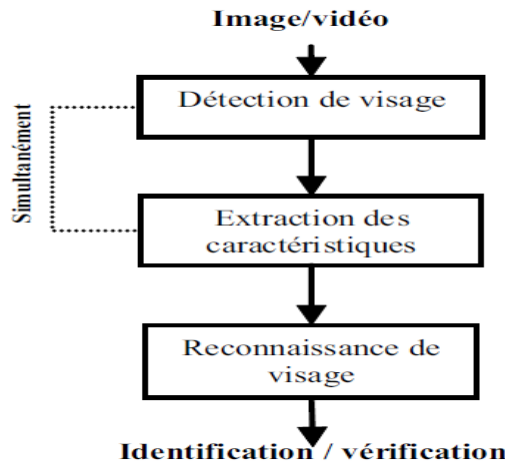


Figure I.1 : Les étapes de reconnaissance de visage

I.4 Identifier ou authentifier ?

Un système biométrique peut servir à identifier une personne : retrouver une donnée biométrique parmi celles d'un grand nombre de personnes enregistrées dans une base de données. Il s'agit alors d'une reconnaissance de 1 parmi n, que l'on symbolise souvent par 1:n [OUA_MEH 09]

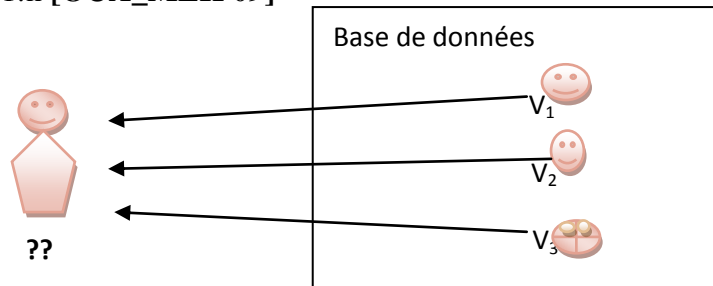


Figure I. 2: L'identification.

Un système biométrique peut servir à authentifier aussi une personne vérifier qu'une donnée biométrique enregistrée dans une carte à puce est bien la même que celle de la personne qui détient cette carte. Il s'agit alors d'une reconnaissance de 1 parmi 1, symbolisée par 1:1. Dans ce cas, les risques de détournement des données biométriques sont bien moins grands que pour l'identification puisqu'il n'y a pas de base de données centralisée susceptible d'être utilisée abusivement. [OUA_MEH 09].

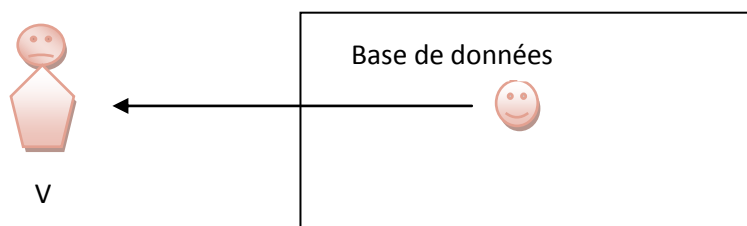


Figure I.3 : L'authentification.

I.5 Le processus de reconnaissance de visages

Tout processus automatique de reconnaissance de visages doit prendre en compte plusieurs facteurs qui contribuent à la complexité de sa tâche, car le visage est une entité dynamique qui change constamment sous l'influence de plusieurs facteurs.

Ce qui conduit au schéma général de reconnaissance des visages [MOHO2].

de la figure. I.4.

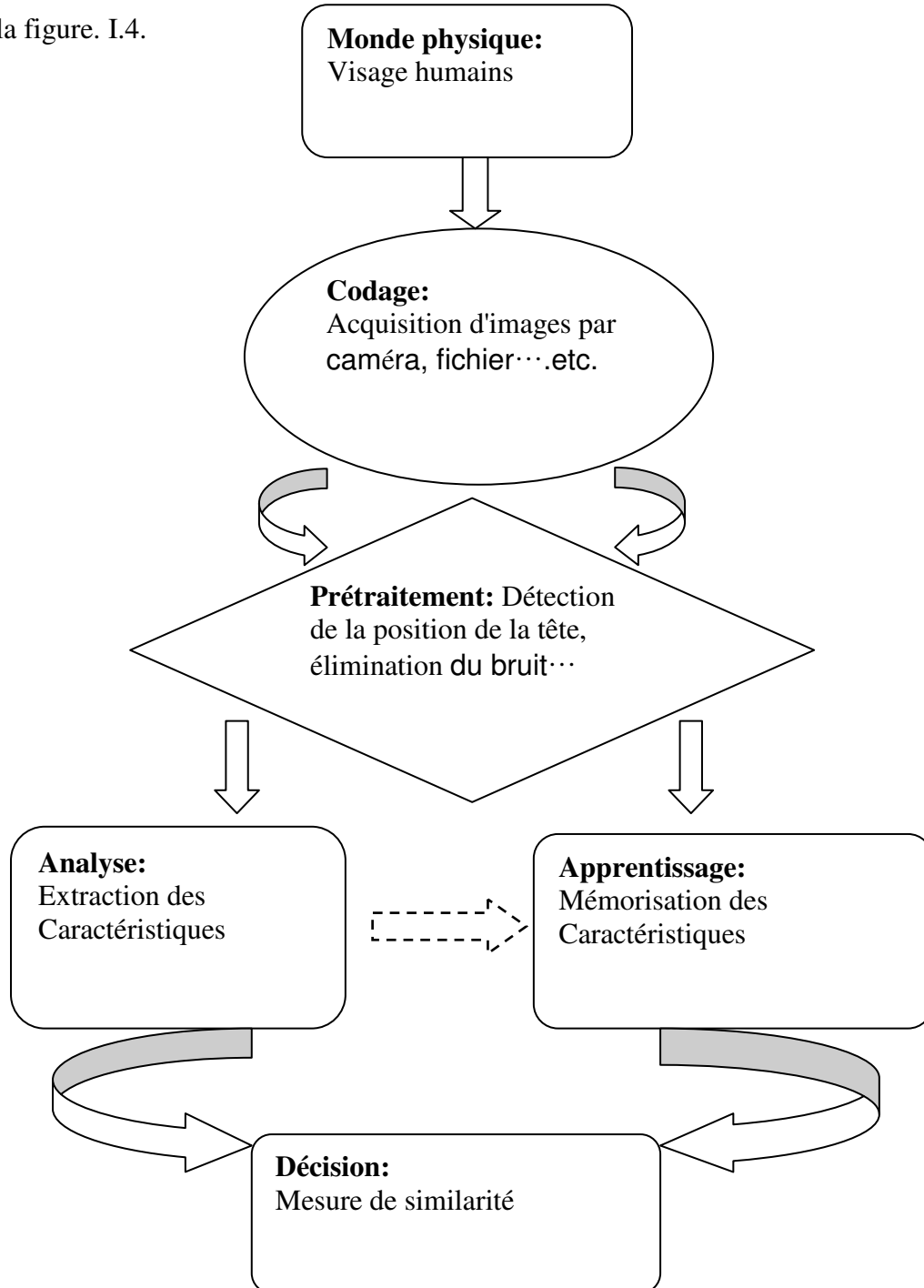


Figure I.4 : Le schéma général d'un système de reconnaissance de visages

Dans le monde physique, il y a trois paramètres à considérer : L'éclairage, la variation de posture et l'échelle. La variation de l'un de ces trois paramètres peut conduire à une distance entre deux images du même individu, supérieure à celle séparant deux images de deux individus différents [MOHO2].

Le codage consiste en l'acquisition d'image et sa digitalisation, il comporte un risque de bruit et donne lieu à une représentation 2-D (la matrice des niveaux de gris) pour un objet 3D (le visage) .

Dans le prétraitement il faut éliminer le bruit par des techniques de traitement et de restauration d'images et procéder à une détection de visages, cette opération est très complexe, surtout dans le cas où l'image contient plusieurs visages ou le cas où l'arrière plan n'est pas neutre. La restauration d'images ou l'élimination du bruit consiste à compenser les dégradations connues ou estimées et rétablir la qualité initiale de l'image.

Les **performances globales** de tout système automatique de reconnaissance dépendent amplement des performances de la détection de visages. Dans l'étape de détection, on identifie et on localise le visage dans l'image acquise au départ, indépendamment de la position, de l'échelle, de l'orientation et de l'éclairage. C'est un problème de classification où on assigne l'image à la classe visage ou à la classe non visage. On peut diviser les **approches de détection** en quatre catégories : les **méthodes basées sur la connaissance** où on code la connaissance humaine du visage, les **méthodes de correspondance de masques**, les **méthodes à caractéristiques invariables** où on utilise la couleur, les textures et les contours, et finalement les méthodes les plus répandues et qui sont ceux basées sur **l'apprentissage ou les statistiques** comme **PCA, SVM et Graph Matching**.

Dans l'étape d'analyse (appelée aussi **indexation, représentation, modélisation** ou **extraction de caractéristiques**), il faut extraire de l'image les informations qui seront sauvegardées en mémoire pour être utilisées plus tard dans la phase de décision. Le choix de ces informations utiles revient à établir un modèle pour le visage, elles doivent être discriminantes et non redondantes.

L'apprentissage consiste à mémoriser les représentations calculées dans la phase analyse pour les individus connus. Généralement les deux étapes d'analyse et d'apprentissage sont confondues et regroupées en une seule étape [MOHO2].

La décision : Pour estimer la différence entre deux images, il faut introduire une **mesure de similarité**. Il est important de noter que le système de vérification automatique de visage se base en sa totalité sur la méthode de localisation.

Dans la suite nous aborderons les différentes méthodes existantes pour la vérification et identification de visage. [MOHO2].

I.6 Mesure de qualité

Le problème qui nous préoccupe contient **deux classes**, à savoir d'une part les **clients** et d'autre part **les imposteurs**.

La matrice de confusion m est donc carrée de dimension deux. Si chaque client doit être accepté et chaque imposteur rejeté, on peut écrire :

$$m = \begin{pmatrix} TBA & TFA \\ TFR & TBR \end{pmatrix} \quad (I.1)$$

Les TFR et TFA sont respectivement les Taux de Faux Rejet et Taux de Fausse Acceptation, ils doivent être bien sûr le plus faible possible. Les TBA et TBR sont respectivement le taux de bonne acceptation et le taux de bon rejet. Ces chiffres caractérisent ce que l'on appelle en reconnaissance de visage le pouvoir d'identification. Nous supposons que le système de vérification soit impitoyable, et extrêmement strict. Le TFA sera bien sûr faible, mais le TFR sera fort élevé. Au contraire, un système laxiste sera caractérisé par un TFA élevé et un TFR plutôt bas. Le juste milieu se situe quelque part entre les deux, et, si les coûts des erreurs sont égaux, il se trouvera au Taux d'Égale Erreur ou TEE illustré par la figure I.5.

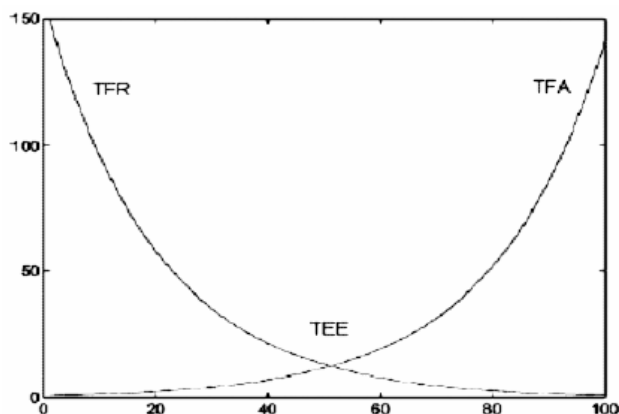


Figure I.5 : Les TFA, TFR et TEE en fonction du laxisme du système.

Tous ces taux d'erreurs sont calculés dans deux ensembles : d'abord dans un ensemble d'évaluation, qui va permettre de fixer plus ou moins le TEE en faisant varier les paramètres d'acceptation et de rejet du système. Ensuite dans un ensemble de test, en utilisant les paramètres fixés précédemment. Ainsi, on peut vérifier la robustesse du système [SBBL05].

Comment mesurer le performances d'un tel système ?

Les erreurs que l'on peut faire sont de deux sortes :

FR =(false rejection) rejeter faussement un client

FA =(false acceptance) accepter faussement un imposteur

$$\text{TFR} = \frac{\text{Nb de FR}}{\text{Nb de clients}} \quad (\text{I.2})$$

$$\text{TFA} = \frac{\text{Nb de FA}}{\text{Nb d' imposteurs}} \quad (\text{I.3})$$

$$\text{TEE} = \frac{\text{Nb de FR} + \text{Nb de FA}}{\text{Nb total d'accès}} \quad (\text{I.4})$$

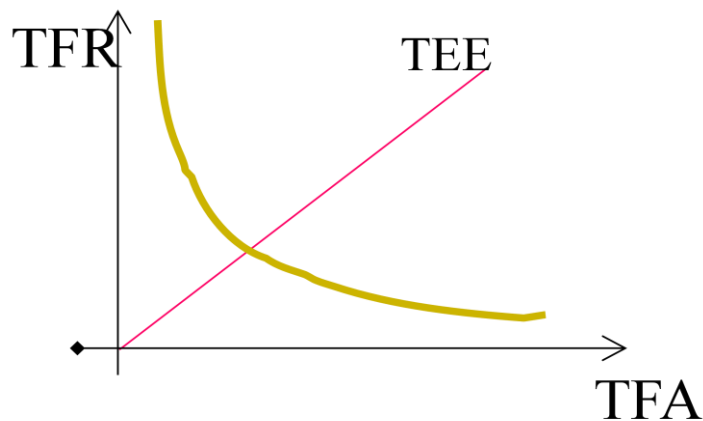


Figure I.6 : La position de TEE de TFR en fonction du TFA

I.7 Les classes des techniques de reconnaissance de visages

Les systèmes de reconnaissance de visage peuvent grossièrement être classés en deux grandes catégories :

- ❖ les méthodes basées sur la reconnaissance des caractéristiques d'un visage humain d'une part,
- ❖ les méthodes dites globales d'autre part.

I.7.1 Les Méthodes locales (géométriques)

On les appelle aussi les méthodes à traits, à caractéristiques locales, ou analytiques ces méthodes **basées** sur les caractéristiques du visage recherchent et analysent les **éléments caractéristiques** d'un visage tels que les yeux, la bouche, le nez, les joues...Après le traitement de chacun de ces éléments, l'ensemble des résultats

obtenus est combiné pour procéder à la reconnaissance du visage. On peut par exemple déterminer la géométrie du visage à partir de ces éléments, notamment en calculant les distances les séparant (distance entre les deux yeux, entre les deux joues etc...), leurs proportions respectives comme en anthropométrie. Cette catégorie de méthodes est robuste par rapport aux variations de la position du visage dans l'image. Un certain nombre de stratégies ont modélisé et classé les visages sur la base de distances normalisées et angles entre points caractéristiques. Cette phase d'extraction des traits caractéristiques du visage constitue l'étape clé du processus, car la performance du système entier en dépend. L'**avantage** de ces méthodes est qu'elles **prennent en compte la particularité du visage** en temps que forme naturelle à reconnaître, et un nombre réduit de paramètres (de 9 à 14 distances au maximum à considérer), en exploitant les résultats de la recherche en neuropsychologie et psychologie cognitive sur le système visuel humain. La **difficulté** éprouvée quand il s'agit de prendre en considération plusieurs vues du visage ainsi **que le manque de précision dans la phase "extraction"** sont des points constituant leur inconvénient majeur [MOD].

Les différents étages de la face

En théorie, le visage comporte trois étages:

- L'étage frontal.
- L'étage nasal, depuis la ligne des sourcils jusqu'au plan sous-nasal.
- L'étage buccal, depuis le plan sous-nasal jusqu'à la pointe du menton.

L'étage buccal peut être découpé en trois sous étages:

- L'étage sous-nasal jusqu'à la ligne bi commissurale.
- L'étage partant de la ligne bi commissurale jusqu'au sillon labio- mentonnier.
- L'étage allant du sillon labio-mentonnier jusqu'au menton.

Habituellement, on a une proportion d'un tiers par étage. Il existe aussi un parallélisme des lignes faciales entre les lignes bipupillaire, biophriaque (ligne des sourcils), bicommissurale et bigoniaque [KHT05].

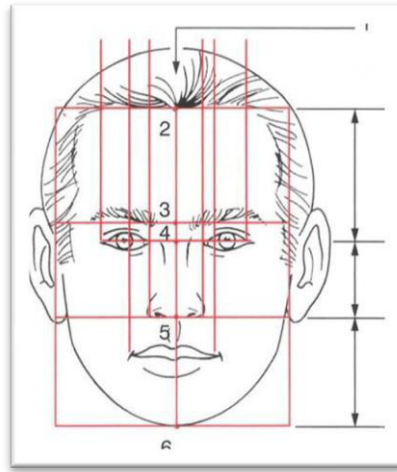


Figure 1.7 : Harmonie faciale
 (rapport des trois étages de la tête) (d’après Bennaceur et al. 2005).
 1. plan de symétrie faciale . 2. trichion . 3. onychion . 4. nasion . 5. subnasal .
 6.gnathion.

I.7.2 Les Méthodes Globales (holistiques)

Cette classe regroupe les méthodes qui mettent en valeur les **propriétés globales** du visage. Le visage est traité comme un tout. Elles se basent principalement sur l’information pixel. Dans ces méthodes (**Neural Networks, Support Vecteur Machine, Principal Component Analysis (Eigenfaces), HMM...**) on génère une base d’exemples à partir de laquelle un classifieur va apprendre ce qu’est un visage (**apprentissage**). Ces **systèmes sont très performants, mais très lents en phase d’apprentissage donc lourds à mettre en œuvre**. Pour atteindre le temps réel (~40ms) ils nécessitent une implémentation spécifique (**ZISCs, FPGA ou DSP**).

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

- ❖ **L’Approche connexionniste** L’inconvénient de cette approche est que l’**apprentissage est long**. A ce jour, il n’est pas claire comment de tels systèmes vont s’étendre à des problèmes plus larges, il faut prendre en compte un grand nombre d’individus.
- ❖ **L’Approche ACP (ou Les Visages Propres)**, son but est de capturer la variation dans une collection d’images de visages et d’utiliser cette information pour coder et comparer les visages (en termes mathématiques : trouver les vecteurs propres de la matrice de covariance de l’ensemble des images de visages). Le nombre possible de visages propres peut être approximé en utilisant seulement les meilleurs visages propres qui correspondent aux plus grandes valeurs propres. Cette approche rencontre

le **problème du coût des calculs élevé** et celui de la **détermination du nombre de visage propres utiles**.

❖ **L'Approche Stochastique** quand les images frontales sont balayées de haut en bas il y a un ordre naturel dans lequel les caractéristiques apparaissent, et de ce fait peut être modélisé d'une manière pratique en utilisant un modèle caché de Markov (**HMM** : Hidden Markov Model). Ce modèle rencontre les **problèmes de prises de vue** des images en entrée.

❖ **L'Approche Probabiliste** cette approche repose essentiellement sur la **théorie de décision** pour résoudre les problèmes de classement et de classification, et pour cela en utilise généralement la classification fondée sur le **théorème de Bayes**. Yang et Ahuja présente une méthode pour détecter des visages humains à partir d'images en couleur. Un modèle de la couleur de la peau humaine basé sur une analyse statistique multi variante est construit pour capturer les propriétés chromatiques, et dans Yang, Ahuja et Kriegman présentent une méthode de probabilité pour détecter les visages humains qui utilisent un mélange d'analyseurs de facteur. Cette approche pose le **problème de la complexité de calcul qui est très élevée**.

❖ **L'Approche Réseaux de Neurones** basée sur les réseaux de neurones artificiels ou RNA sont des assemblages fortement connectés d'unités de calcul. Chacune des unités de calcul est un neurone formel qui est, en soi, une formulation mathématique ou un modèle très simplifié d'un neurone biologique. Les **RNA** ont des capacités de **mémorisation**, de **généralisation** et d'une certaine forme d'**apprentissage**.

On classe généralement les réseaux de neurones en deux catégories:

- les **réseaux faiblement connectés à couches** que l'on appelle des réseaux « **feed forward** »
- **réseaux directs** et les **réseaux fortement connectés** que l'on appelle des **réseaux récurrents**.

Dans ces deux configurations, on retrouve des connexions totales ou partielles entre les couches. Les **réseaux de neurones** peuvent être utilisé tant pour la **classification**, la **compression de données** ou dans le **contrôle de systèmes complexes en automatisme**.

Cette **approche repose** essentiellement sur la notion d'**apprentissage** qui est depuis de nombreuses années au cœur des recherches en intelligence artificielle. L'idée est d'identifier à partir d'exemples un visage (ici une personne). De manière plus formelle, l'apprentissage du réseau a pour but l'extraction des informations pertinentes à l'identification.

L'**avantage** de ce modèle est le **gain de temps** considérable. Cependant, l'utilisation d'exemples pour apprendre apporte **le risque** de ne pouvoir résoudre que des situations déjà rencontrées, où un phénomène de **sur-apprentissage** qui spécialiserait le réseau uniquement sur les exemples connus sans généraliser [MOHO2].

I.7.3 Méthodes hybrides

Les méthodes hybrides sont des approches qui combinent les caractéristiques holistiques et locales afin d'améliorer les performances de la reconnaissance de visages. En effet, les caractéristiques locales et les caractéristiques globales ont des propriétés tout à fait différentes. On peut espérer pouvoir exploiter leur complémentarité pour améliorer la classification. Le **tableau I.1** récapitule qualitativement la différence entre les deux types de caractéristiques.

Facteurs de variations	Caractéristiques locales	Caractéristiques globales
Illuminations	Très sensible	Sensible
Expressions	Pas sensible	Sensible
Pose	Sensible	Très sensible
Bruit	Très sensible	Sensible
Occlusion	Pas sensible	Très sensible

Tableau I.1 Comparaison des méthodes basées sur les caractéristiques locales ou globales.

Nous pouvons voir que les **caractéristiques locales** et **globales** réagissant **différemment sensibles aux facteurs de variation**. Par exemple, les changements d'illumination peuvent avoir plus d'influence sur les caractéristiques locales, tandis que les changements d'expression ont plus d'impact sur les caractéristiques holistiques. Ainsi, les **méthodes hybrides** peuvent constituer **une approche efficace pour réduire la complexité des classifieurs et améliorer leur capacité de généralisation**. Malgré ces avantages, les travaux de recherche dans ce domaine restent relativement peu nombreux, probablement en raison des difficultés mentionnées ci-dessus, tandis que les méthodes hybrides classiques (c'est-à-dire qui utilisent plusieurs exemples d'apprentissage par personne), comme les modèles

d'apparence flexibles ou le LFA hybride ne sont pas généralement appropriées pour traiter le cas d'un seul exemple d'apprentissage [SOU08].

I.8 Conclusion

Dans ce chapitre, nous avons présenté une rapide historique de développement biométrique ensuite la définition de « la Biométrie » ainsi que les nouvelles technologies de la surveillance « Les empreintes digitales, La main, L'œil, La voix L'oreille, La signature et l'ADN ». Nous avons aussi abordé les différentes étapes du processus d'identification ainsi que les performances d'un système de vérification biométrique « les taux d'erreurs TFA, TFR ». 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 projet de fin d'étude, notamment l'invariance à l'illumination. Les techniques utilisées aux différentes étapes de la reconnaissance de visage sont détaillées dans les chapitres suivants.

II.1 Introduction

La reconnaissance de visages est une technologie biométrique en vogue, elle est très utilisée dans les applications de contrôle d'accès. Dans la littérature, on trouve plusieurs méthodes globales, locales et hybrides de reconnaissance de visages. Dans ce chapitre, nous introduisons et nous expliquons les différents algorithmes classiques de reconnaissance de visages, notamment : **ICA, LVQ, PCA, LDA...** Notre travail est basé sur une méthode qui se base sur l'**ACP2D** et **LDA2D** qui sera détaillée dans le chapitre III.

II.2 Les méthodes de reconnaissance de visage

II.2.1 Le réseau de neurones (LVQ)

L'application des réseaux de neurones artificiels dans la reconnaissance de visages a visé plusieurs problèmes : la classification Homme-Femme, la reconnaissance de visages et la classification des expressions faciales [**RED06**].

Un réseau de neurones est un système de traitement de l'information qui a été développé comme généralisation des modèles mathématiques assortissant la connaissance humaine. Ils se composent d'un grand nombre d'unités de traitement hautement reliées appelées neurones, travaillant ensemble pour exécuter une tâche de classification donnée.

Un réseau de neurones est un processeur parallèle distribué, ayant une prospérité naturelle pour stocker une connaissance expérimentale. Il ressemble au cerveau humain en trois aspects: la **connaissance** est acquise par le réseau par un **processus d'apprentissage**, des forces de connexion reliées ensemble, connues sous le nom de poids synaptiques, sont employées pour **stocker la connaissance**, et chaque neurone a un état interne appelé seuil ou fonction d'activation (ou fonction de transfert) utilisée pour **classifier les vecteurs**.

On trouve plusieurs types de réseaux de neurones : l'**apprentissage compétitif** (non-supervisé, parfois appelé **VQ**), le **LVQ (apprentissage compétitif supervisé)**, les **cartes auto-organisatrices**, le **perceptron multicouches avec apprentissage par rétro propagation**, les **réseaux probabilistes**, les **réseaux à fonctions radiales de base (FRB)**, et les **réseaux récurrents** [**RED06**].

Une classification par réseau de neurones comporte les étapes suivantes :

- Tout d'abord une phase de prétraitement des images d'apprentissage et l'association à chaque **image d'apprentissage (entrée du réseau)** un vecteur de sortie.
- Puis vient l'**étape d'initialisation (création des couches du réseau)**.
- On fait l'**apprentissage (supervisé) du réseau**, jusqu'à **atteindre** une certaine **erreur minimale** (le réseau apprend à bien classifier les images d'apprentissage).
- On présente ensuite au réseau une nouvelle image à identifier (**phase de reconnaissance ou de simulation ou d'activation du réseau**) qui sera **finalement affectée à une classe donnée [RED06]**.

II.2.2 Analyse en Composantes Principales (PCA)

L'algorithme **PCA** est né des travaux de *M. A. Turk* et *A. P. Pentland* au **MIT Media Lab**, en **1991**. Il est aussi connu sous le nom de *Eigenfaces* car il utilise des vecteurs propres et des valeurs propres (respectivement *Eigenvectors* et *Eigenvalues* en anglais).

L'idée principale consiste à exprimer M images de départ selon une base de vecteurs orthogonaux particuliers - les vecteurs propre- contenant des informations indépendantes d'un vecteur à l'autre.

Les données originales seront donc exprimées d'une manière plus appropriée à la reconnaissance du visage.

En **théorie de l'information**, 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.

Une image $\mathbf{I}_{(m,n)}$ est traitée comme un vecteur $\Gamma_{i(m \times n, 1)}$ dans un espace vectoriel de grande dimension ($\mathbf{N} = \mathbf{m} \times \mathbf{n}$), par concaténation des colonnes (**Figure II.1**).

$$\mathbf{I}_1 = \begin{pmatrix} a_{1,1} & \dots & a_{1,m} \\ \vdots & \ddots & \vdots \\ a_{n,1} & \dots & a_{n,m} \end{pmatrix} \Rightarrow \Gamma_1 = \begin{pmatrix} a_{1,1} \\ \vdots \\ a_{n,1} \\ \vdots \\ a_{1,m} \\ \vdots \\ a_{n,m} \end{pmatrix}$$

Figure II.1: Passage d'une image vers un vecteur dans un espace vectoriel de grande dimension.

Les coefficients $a_{i,j}$ représentent les valeurs des pixels en niveau de gris, codés de 0 à 255.

Après avoir rassemblé les **M images** dans une unique matrice, nous obtenons **une matrice d'images** Γ , où chaque colonne représente une image Γ_i :

$$\Gamma = \begin{pmatrix} a_{1,1} & b_{1,1} & \dots & z_{1,1} \\ \vdots & \vdots & \dots & \vdots \\ a_{n,1} & b_{n,1} & \dots & z_{n,1} \\ \vdots & \vdots & \dots & \vdots \\ a_{1,m} & b_{1,m} & \dots & z_{1,m} \\ \vdots & \vdots & \dots & \vdots \\ a_{n,m} & b_{n,m} & \dots & z_{n,m} \end{pmatrix}$$

On calcule ensuite l'**image moyenne** Ψ de toutes les images collectées. Cette image peut être vue comme le centre de gravité du jeu d'images :

$$\Psi = \frac{1}{M} \sum_{i=1}^M \Gamma_i \tag{II.1}$$

On ajuste ensuite les données par rapport à la moyenne. L'image moyenne est alors soustraite de chaque image avec la formule suivante :

$$\Phi_i = \Gamma_i - \Psi, i = 1 \dots M \tag{II.2}$$

On calcule ensuite la matrice de covariance du jeu de données. Cette matrice peut être vue comme une matrice de moments d'ordre 2 :

$$C = \sum_{i=1}^M \Phi_i \Phi_i^T = AA^T, A = [\Phi_1 \Phi_2 \dots \Phi_M] \quad (\text{II.3})$$

La prochaine étape consiste à calculer les vecteurs propres et les valeurs de cette **matrice de covariance C** de taille $(N \times N)$, c'est-à-dire de l'ordre de la résolution d'une image.

Le **problème** est que cela peut parfois être très difficile et **très long** ! En effet, si $N > M$ (si la résolution est supérieure au nombre d'images), il y aura seulement **M-1 vecteurs propres qui contiendront de l'information** (les vecteurs propres restants auront des valeurs propres associées nulles). Par exemple, pour 100 images de résolution 320×240, nous pourrions résoudre une matrice L de 100×100 au lieu d'une matrice de 76800×76800 pour ensuite prendre les combinaisons linéaires appropriées des images Φ_i . Le gain de temps de calcul serait considérable ! Typiquement, nous passerions d'une complexité de l'ordre du nombre de pixels dans une image à une complexité de l'ordre du nombre d'images.

Voici comment procéder pour accélérer les calculs :

- Considérons les vecteurs propres e_i de $C = AA^T$, associés aux valeurs propres λ_i

$$C e_i = \lambda_i e_i \quad (\text{II.4})$$

- Les vecteurs propres v_i de $L = A^T A$, associés aux valeurs propres μ_i sont tels que:

$$L v_i = \mu_i v_i \quad (\text{II.5})$$

Soit :

$$A^T A v_i = \mu_i v_i \quad (\text{II.6})$$

- En multipliant à gauche par A des deux côtés de l'égalité, nous obtenons :

$$A A^T A v_i = A \mu_i v_i \quad (\text{II.7})$$

Puisque $C = AA^T$, nous pouvons simplifier :

$$C(Av_i) = \mu_i(Av_i) \quad (\text{II.8})$$

De (II.4) et (II.8), nous voyons que $A_{\cdot i}$ et μ_i sont respectivement les vecteurs propres et les valeurs propres de C :

$$\begin{cases} e_i = Av_i \\ \lambda_i = \mu_i \end{cases} \quad (\text{II.9})$$

Nous pouvons donc trouver les valeurs propres de cette **énorme** matrice C en trouvant les valeurs propres d'une matrice L beaucoup **plus petite**. Pour trouver les vecteurs propres de C , il suffit juste de **pré-multiplier** les vecteurs propres de L par la matrice A .

Les **vecteurs propres** trouvés sont ensuite **ordonnés** selon leurs valeurs propres correspondantes, de manière décroissante. Plus une valeur propre est grande, plus la variance capturée par le vecteur propre est importante. Cela implique que la majeure partie des informations est contenue dans les premiers vecteurs propres. Une part de la grande efficacité de l'**algorithme PCA** vient de l'étape suivante qui consiste à ne sélectionner que les k meilleurs vecteurs propres (ceux avec les k plus grandes valeurs propres). On définit alors un espace vectoriel engendré par ces k vecteurs propres, que l'on appelle l'espace des visages E_v ("**Face Space**").

Les images originales peuvent être reconstituées par combinaison linéaire de ces vecteurs propres. Les représentations graphiques de ces vecteurs rappellent un peu des images fantômes, chacune mettant en avant une partie du visage, on les appelle **Eigenfaces** (Figure II.2).

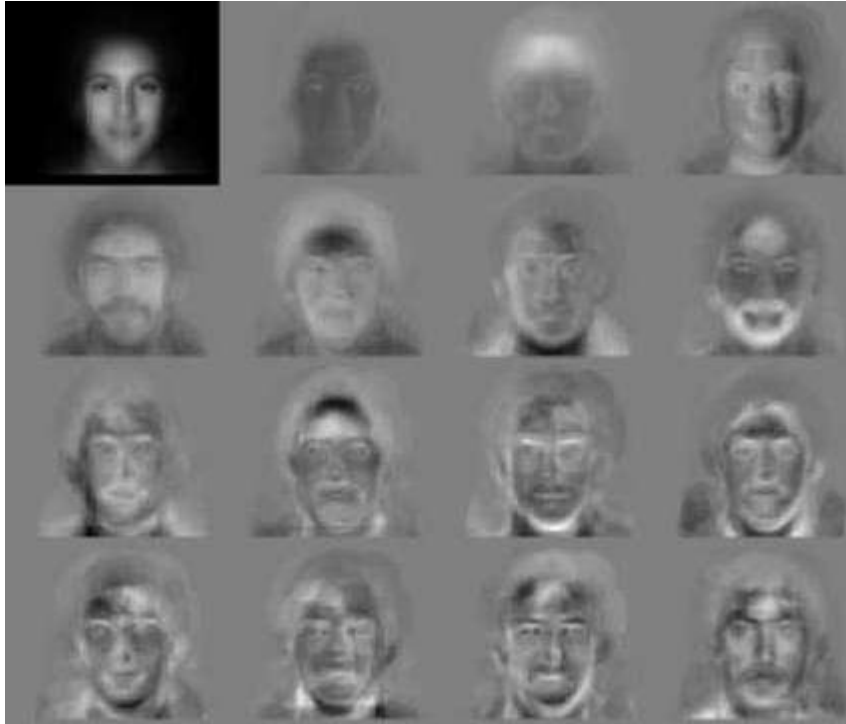


Figure II.2: Image moyenne et les 15ères EigenFaces.

Nous allons maintenant projeter nos images de départ sur E_v . Une image Γ_i est alors transformée en ses composantes eigenfaces par une simple opération de

$$\omega_k = e_k^T (\Gamma_i - \Psi), k = 1, \dots, M' \quad (\text{II.10})$$

projection vectorielle:

Les vecteurs ω_k sont appelés *poids* et forment une matrice $\Omega^T = [\omega_1, \omega_2, \dots, \omega_{M'}]$ qui décrit la contribution de chaque eigenface dans la représentation de l'image d'entrée.

La matrice Ω est alors utilisée pour trouver quelle est, parmi un nombre pré-défini de classes, celle qui décrit le mieux une image d'entrée.

La méthode la plus simple pour déterminer quelle classe de visage fournit la meilleure description d'une image d'entrée est de trouver la classe de visage k qui minimise la distance euclidien où Ω_k est un vecteur qui décrit la k^e classe de visage.

$$\varepsilon_k^2 = \|\Omega - \Omega_k\|^2 \quad (\text{II.11})$$

Un visage appartient à une classe k quand le minimum ε_k est en dessous d'un certain seuil θ_ε . Dans le cas contraire, le visage est classé comme étant *inconnu* et peut éventuellement être utilisé pour créer une nouvelle classe de visage. La création de la matrice de poids Ω_T est équivalent à la projection du visage original sur E_v . Puisque la distance ε entre l'image de visage et E_v est simplement le carré de la distance entre l'image d'entrée réajustée par rapport à la moyenne :

$$\Phi = \Gamma - \Psi \text{ et } \Phi_f = \sum_{i=1}^{M'} \omega_i e_i \tag{II.12}$$

Sa projection sur E_v est :

$$\varepsilon^2 = \|\Phi - \Phi_f\|^2 \tag{II.13}$$

Il y a alors **quatre possibilités** pour une image d'entrée d'être reconnue ou non (**Figure II.3**).

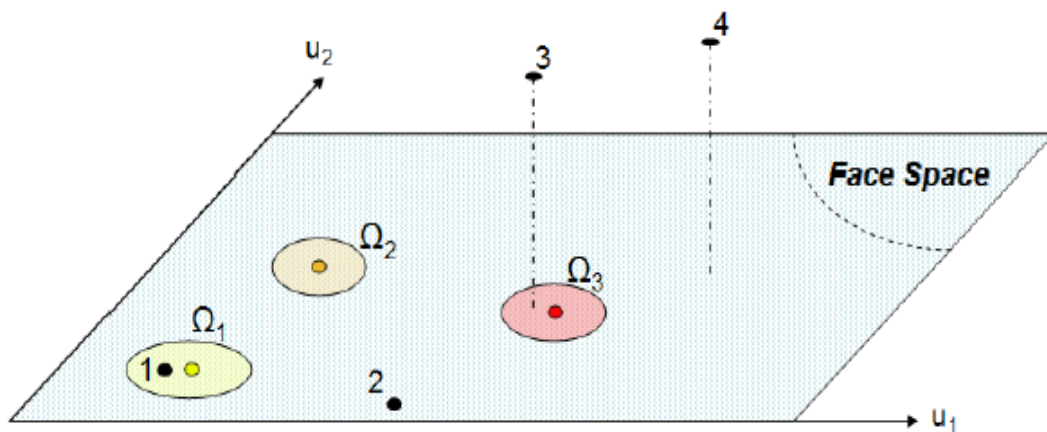


Figure II.3 : Une version simplifiée de E_v illustrant les quatre résultats de la projection d'une image sur E_v .

Dans ce cas, il y a deux vecteurs propres (u_1 et u_2) et trois classes d'individus connus ($\Omega_1, \Omega_2, \Omega_3$). Dans le cas 1, un individu est reconnu et identifié. Dans le cas 2, un individu inconnu est présent. Les deux derniers cas (3 et 4) indiquent que l'image n'est pas une image de visage. Pour le cas 3, l'image est éloigné de E_v mais la projection est proche d'une classe connue, on parle alors de fausse acceptation. Ces remarques sont résumées dans le tableau II.1

	Espace des Visages (E_v)	Classe de Visage
Cas 1	proche	proche
Cas 2	proche	éloigné
Cas 3	éloigné	proche
Cas 4	éloigné	éloigné

Tableau II.1: Les quatre possibilités qui apparaissent lors de la phase de reconnaissance.

En résumé, l'algorithme **PCA** est une **méthode globale** utilisant en premier lieu les niveaux de gris des pixels d'une image. Sa **simplicité** à mettre en oeuvre contraste avec une **forte sensibilité** aux changements d'éclairément, de pose et d'expression faciale.

Néanmoins, le PCA ne nécessite **aucune connaissance à priori** sur l'image et se révèle plus **efficace** lorsqu'il est couplé à la mesure de **distance MahCosine**.

Le principe selon lequel on peut construire un sous-espace vectoriel en ne **retenant** que les « **meilleurs** » vecteurs propres, tout en conservant beaucoup d'information utile, fait du PCA un algorithme **efficace** et **couramment utilisé en réduction de dimensionnalité**.

Enfin, l'**étude** théorique de l'algorithme PCA est très **pédagogique** et permet d'acquérir de solides bases pour la reconnaissance 2D du visage. C'est un algorithme incontournable!

II.2.3 Analyse en Composantes Indépendantes (ACI)

L'analyse en composantes principales (**PCA**) est une méthode classique et très répandue dans le domaine biométrique. C'est une technique optimale de recherche de représentation réduite qui minimise l'erreur de reconstruction, cependant les vecteurs de base tenant compte de l'erreur de reconstruction peuvent ne pas être optimaux pour coder l'information appropriée de l'image à la classification. L'*analyse en composantes indépendantes (ICA)* est une généralisation de **PCA** qui utilise en plus des statistiques d'ordre 2, les statistiques d'ordre supérieur, ce qui peut produire une représentation de données plus puissante. Le **but** d'**ICA** est de **trouver** des vecteurs de base (images de base) localisés dans l'**espace** et qui sont **statistiquement indépendants**, en réduisant au minimum la **dépendance statistique**.

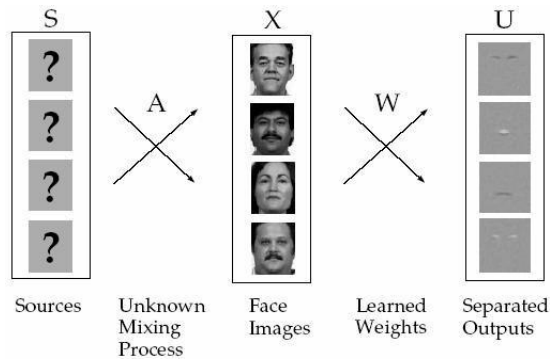


Figure II.4 : Le modèle de synthèse d'image de l'architecture 1 d'ICA

Avant l'apprentissage, on soustrait l'image moyenne des images d'apprentissage X et on les filtre par un filtre de blanchiment "Whitening Filter" $W_z = 2 * (XX^T)^{-1/2}$ pour supprimer les statistiques d'ordre 1 et 2 (la moyenne et les covariances sont mises à zéro et les variances sont égalisés). Pour trouver un ensemble d'images composantes indépendantes, Les **images d'apprentissage X** sont considérées comme étant une combinaison linéaire d'images de base statistiquement indépendantes S , où A est une **matrice de mixage** inconnue. Les images de base sont retrouvées par une matrice de filtre d'apprentissage $W_i = W * W_z$, qui produit des sorties U statistiquement indépendantes. Ces **images de base** sont considérées comme un **ensemble de caractéristiques faciales statistiquement indépendantes**, où les valeurs des pixels de chaque image caractéristique sont statistiquement indépendantes des valeurs des pixels des autres images caractéristiques.

The diagram shows a grayscale face image on the left, followed by an equals sign and a series of terms: $b_1 * u_1 + b_2 * u_2 + \dots + b_n * u_n$. Each u_i is represented by a small grayscale image showing a specific facial feature, such as eyes, nose, or mouth.

Figure II.5: La représentation d'ICA = (b_1, b_2, \dots, b_n)

La représentation d'ICA est constituée des coefficients b de la combinaison linéaire des images de base U qui forment chaque image de visage.

Le **nombre** de composantes indépendantes extraites par l'algorithme d'ICA est **égal** à la **dimension** de l'**entrée** (le nombre d'images d'apprentissage), et si on veut **contrôler** ce nombre, au lieu d'exécuter ICA sur les n images originales, on

l'exécute sur un ensemble de **m combinaisons linéaires** de ces images, où **m < n**. Communément, on choisi pour ces combinaisons linéaires un sous ensemble formé des **premières composantes principales** de la **matrice de donnée**, ainsi l'exécution d'ICA sur l'ensemble de ces m composantes principales **produit m images sources U_i statistiquement indépendantes**.

Dans cette première architecture d'ICA, les images de base U_i sont **statistiquement indépendantes, mais les coefficients b_i ne le sont pas**. Au lieu de séparer les images de visages en ensembles d'images indépendantes, il existe une deuxième architecture d'ICA, dans laquelle **les coefficients sont statistiquement indépendants**, en séparant les pixels en un ensemble de variables indépendantes codant les images.

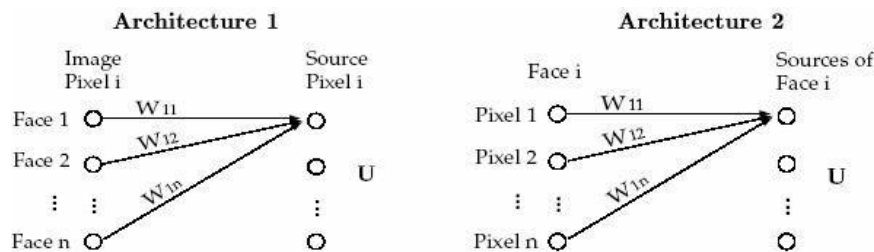


Figure II.6 Les deux architectures d'ICA

L'architecture 1 pour trouver des images de base statistiquement indépendantes (la séparation des images de visages produit des images composantes indépendantes), et l'architecture 2 pour trouver un code factoriel (la séparation des pixels produit un code factoriel).

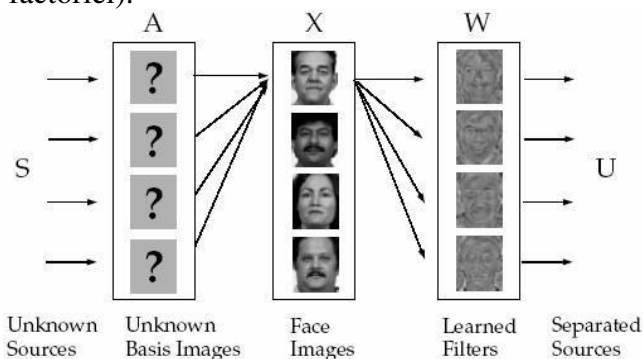


Figure II.7: Le modèle de synthèse d'image de l'architecture 2 d'ICA

Dans l'architecture 2 d'ICA, chaque **image** de l'ensemble de **données** est considérée comme étant **créée à partir** d'un ensemble d'**images de base** en A, et

d'un vecteur de causes d'images "image causes" fondamentales statistiquement indépendantes en S . L'algorithme d'ICA essaye d'inverser les images de base en trouvant un ensemble de **filtres W_i** qui **produisent des sorties statistiquement indépendantes**.

$$\text{Image} = u_1 * a_1 + u_2 * a_2 + \dots + u_n * a_n$$

Figure II.8: La représentation factorielle d'ICA = (u_1, u_2, \dots, u_n)

La représentation factorielle du code est constituée des coefficients indépendants u , de la combinaison linéaire des images de bases en A qui forment chaque image de visage x [RED06].

II.2.4 Analyse Discriminante Linéaire (LDA)

L'algorithme **LDA** est né des travaux de **Belhumeur et al.** De l'Université Yale (**New Haven, USA**), en **1997**. Il est aussi connu sous le nom de **Fisherfaces**.

Contrairement à l'algorithme PCA, l'algorithme **LDA effectue** une véritable *séparation de classes* (**Figure. II.9**).

Pour pouvoir l'utiliser, il faut donc au préalable organiser la base d'images d'apprentissage en plusieurs classes : une classe par personne et plusieurs images par classe. Le **LDA** analyse les vecteurs propres de la **matrice de dispersion des données**, avec pour **objectif** :

- **maximiser** les **variations inter-classes**
- tout **en minimisant** les variations **intra-classes**.

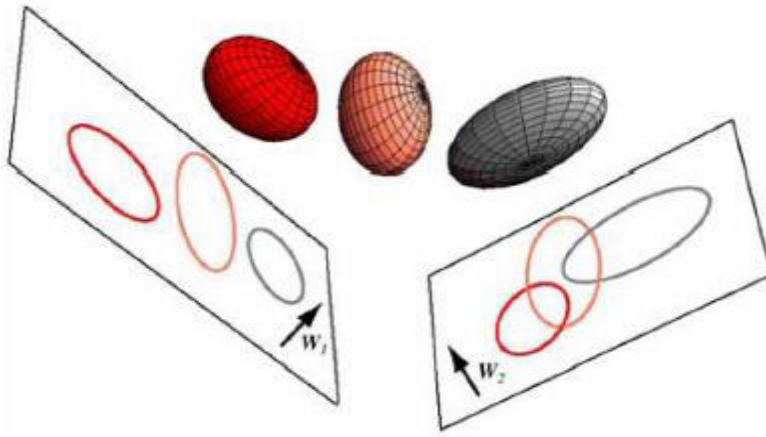


Figure II.9: Illustration du principe de séparation optimale des classes par le LDA.

Trois distributions 3D sont projetées sur deux sous-espaces 2D décrits par les vecteurs W_1 et W_2 . Puisque le LDA essaye de trouver la plus grande séparation parmi les classes, on voit bien que W_1 est ici le vecteur optimal [MOR09].

Tout **comme** dans le **PCA**, on rassemble les **images de la base d'apprentissage** dans une grande matrice d'images Γ où **chaque colonne** représente une **image** Γ_i , puis on calcule

l'image moyenne Ψ . Ensuite, pour chaque classe C_i , on calcule l'image moyenne Ψ_{C_i} , avec q_i le nombre d'images dans la classe C_i .

$$\Psi_{C_i} = \frac{1}{q_i} \sum_{k=1}^{q_i} \Gamma_k \tag{II.14}$$

Chaque image Γ_i de chaque classe C_i est ensuite recentrée par rapport à la moyenne.

On obtient alors une nouvelle image Φ_i :

$$\Phi_i = \Gamma_i - \Psi_{C_i} \tag{II.15}$$

Vient ensuite le calcul de nos différentes matrices de dispersion. On notera c le nombre total de classes (i.e. le nombre d'individus), q_i le nombre d'images dans la classe C_i et M le nombre total d'images.

1. La Matrice de Dispersion Intra-Classe (\mathbf{S}_w):

$$\mathbf{S}_w = \sum_{i=1}^c \sum_{\Gamma_k \in C_i} (\Gamma_k - \Psi_{C_i})(\Gamma_k - \Psi_{C_i})^T \quad (\text{II.16})$$

2. La Matrice de Dispersion Inter-Classe (\mathbf{S}_b):

$$\mathbf{S}_b = \sum_{i=1}^c q_i (\Psi_{C_i} - \Psi)(\Psi_{C_i} - \Psi)^T \quad (\text{II.17})$$

3. La Matrice de Dispersion Totale (\mathbf{S}_T):

$$\mathbf{S}_T = \sum_{i=1}^M (\Gamma_i - \Psi)(\Gamma_i - \Psi)^T \quad (\text{II.18})$$

Une fois ces matrices calculées, nous devons trouver une projection optimale \mathbf{W} qui minimise la dispersion intra-classe, relative à la matrice \mathbf{S}_w , tout en maximisant la dispersion inter-classe, relative à la matrice \mathbf{S}_b . En d'autres termes, nous devons trouver \mathbf{W} qui maximise le **critère d'optimisation de Fisher** $J(\mathbf{T})$:

$$\begin{aligned} W &= \arg \max_T (J(T)) \\ \Rightarrow \max(J(T)) &= \frac{|T^T \mathbf{S}_b T|}{|T^T \mathbf{S}_w T|} \Big|_{T=W} \end{aligned} \quad (\text{II.19})$$

\mathbf{W} peut alors être trouvée en résolvant le problème généralisé aux valeurs propres [MOR09] :

$$\mathbf{S}_b \mathbf{W} = \lambda_w \mathbf{S}_w \mathbf{W} \quad (\text{II.20})$$

Une fois \mathbf{W} trouvée, le même schéma que le **PCA** concernant la projection des images apprises ainsi que la projection d'une image test est appliqué.

Ainsi, la projection vectorielle d'une image apprise réajustée par rapport à la moyenne Φ_i est définie par :

$$g(\Phi_i) = W^T \Phi_i \quad (\text{II.21})$$

La phase de reconnaissance d'une image test Φ_t s'effectue en projetant Φ_t sur W^T :

$$g(\Phi_t) = W^T \Phi_t \quad (\text{II.22})$$

Enfin, on effectue une mesure de distance entre l'image test et l'image projetée sur l'espace vectoriel engendré par W^T . Par exemple, pour la distance euclidienne, on calcule la distance d_{ti} :

$$d_{ti} = \|g(\Phi_t) - g(\Phi_i)\| \quad (\text{II.23})$$

d'où :

$$d_{ti} = \sqrt{\sum_{k=1}^c (g(\Phi_t) - g(\Phi_i))^2} \quad (\text{II.24})$$

Finalement, une image test est dans la classe dont la distance est minimale par rapport à toutes les autres distances de classe.

En résumé, l'algorithme **LDA** permet d'effectuer une **véritable séparation de classes, selon un critère mathématique** qui **minimise** les variations entre les images d'un même individu (**variations intra-classe**) tout en **maximisant** les variations entre les images d'individus différents (**variations inter-classes**). **Cependant**, pour des **problèmes « sous-échantillonnés »** en reconnaissance du visage, c'est-à-dire lorsque le nombre d'individus à traiter est plus faible que la résolution de l'image, **il est difficile d'appliquer le LDA** qui peut alors faire apparaître des matrices de dispersions singulières (non inversibles). Afin de **contourner ce problèmes**, certains algorithmes basés sur le LDA ont récemment été mis au point (par exemple, les algorithmes **ULDA, OLDA** et **NLDA**).

II.3 Conclusion

Dans ce chapitre, nous avons présentés les différentes méthodes de reconnaissance de visage les plus proches du modèle que nous nous proposons d'étudier. Ces méthodes sont basées sur le processus de classification en utilisant **LVQ, ICA, PCA, LDA**. Nous avons essayé de présenter des modèles mathématiques spécifiques et efficaces, pour la tâche de reconnaissance des visages ainsi qu'une explication des techniques **PCA** et **LDA** qui sont essentielles à notre étude. Enfin différentes fonctions de mesure de similarité sont présentées. Celles-ci sont utilisées dans la dernière étape du système de reconnaissance du visage pour l'identification.

Le chapitre suivant sera consacré à l'étude de la **PCA2D** et **LDA2D** et leur application au système de reconnaissance de visage.

III.1 Introduction

Dans la littérature, plusieurs algorithmes de reconnaissance du visage ont été créés pendant ces dernières années. La performance de ces algorithmes dépend fortement de la qualité des résultats de détection et de normalisation des visages.

Cela veut dire que: Plus la précision obtenue est élevée, plus les conditions (d'acquisition, d'éclairage, pose, etc...) se rapprocheront de celles de la phase d'apprentissage, ce qui donne une authentification efficace.

III.2 Prétraitement

Nous voulons utiliser les images de la base de donnée **XM2VTS** pour notre application, des étapes de prétraitement sont nécessaires dans le but de préparer ces images afin de les utiliser dans l'étape d'EigenFaces pour l'authentification de visage et qui est basée sur la comparaison d'images.

Ces images contiennent plusieurs effet indésirables comme : l'arrière plan, les cheveux, les cols de chemise etc... Cela veut dire que la localisation et de garder seulement une fenêtre qui contienne l'essentiel du visage les yeux, le nez, la bouche, donc nous supprimons tous détails indésirables qui ne servent à rien et gonflent inutilement la taille des données. Ainsi, nous réalisons la phase de prétraitement et nous normalisons les images que nous utilisons, et d'en extraire une représentation canonique. Aussi, les images doivent toutes être centrées et calibrées, et doivent être toutes de la même dimension et codées de la même manière.

Pour cela, la phase de prétraitement se décompose en plusieurs étapes :

- le découpage d'images,
- conversion des images en niveaux de gris.
- finalement la détection de contours par la transformation de Hough.

Nous essayons dans ce qui suit de faire l'étude de la transformation de Hough. C'est un outil mathématique que nous avons seulement utilisé. Rappelons que notre objectif reste l'étude et la réalisation d'un système de reconnaissance de visage à base de l'association de l'ACP2D et la LDA2D.

III.2.1 La transformée de Hough

La transformée de **Hough** est une technique de reconnaissance des formes inventée en **1962** par **Paul Hough**, utilisée dans le traitement d'images numériques. L'application la plus simple permet de **reconnaître les lignes d'une image**, mais des modifications peuvent être apportées pour reconnaître n'importe quelle forme : c'est la *transformée généralisée de Hough* développée par Richard Duda et Peter Hart en 1972.

Le **principe** de la transformée de **Hough** est qu'il **existe un nombre infini de lignes qui passent par un point**, dont la **seule différence est l'orientation (l'angle)**.

Le **but** de la transformée est de **déterminer lesquelles** de ces lignes passent **au plus près du schéma attendu**.

Afin de déterminer que deux points se trouvent sur une même ligne potentielle, on doit créer une représentation de la ligne qui permet une comparaison dans ce contexte. Dans la transformée de **Hough**, dite aussi transformée standard de **Hough** ou **SHT**, chaque ligne est un vecteur de coordonnées paramétriques :

θ : l'angle

ρ : la norme du vecteur (la longueur du segment perpendiculaire à la droite d'angle θ et passant par l'origine)

En transformant toutes les lignes possibles qui relient un point à un autre, c'est-à-dire en calculant la valeur de ρ pour chaque θ , on obtient une sinusoïde unique appelée *espace de Hough*. Si les courbes associées à deux points se coupent, l'endroit où elles se coupent dans l'espace de Hough correspond aux paramètres d'une droite qui relie ces deux points [MEG09].

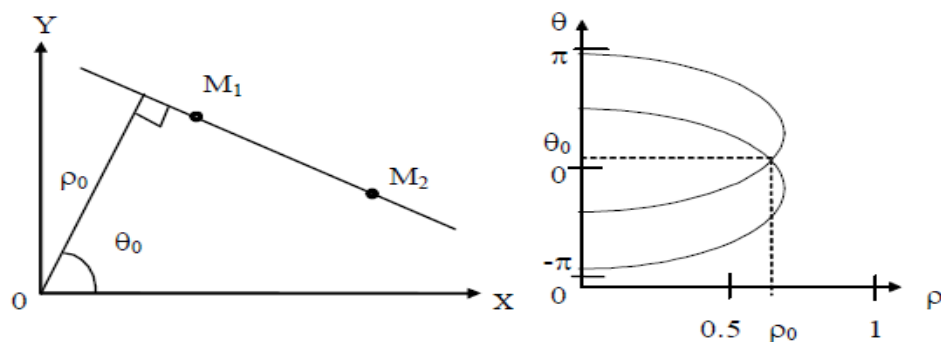


Figure III.1 : Transformée de Hough (a) Plan cartésien (x y) (b) Plan des paramètres (a b)

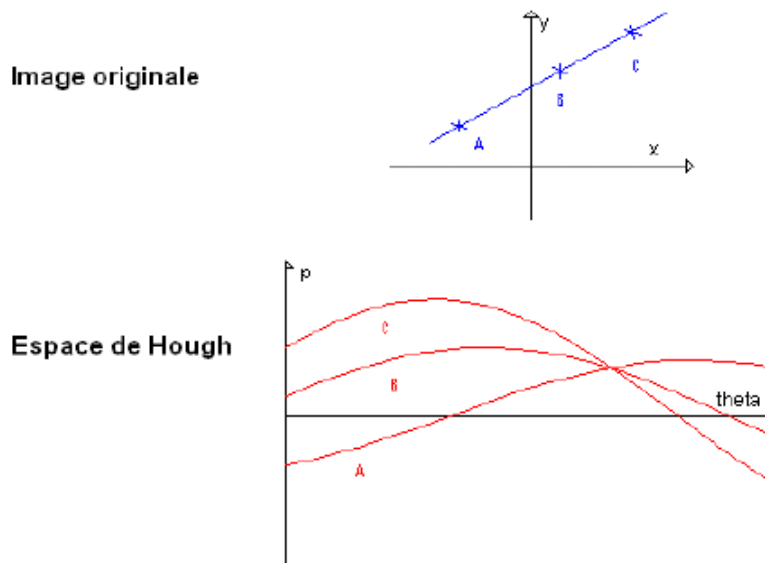


Figure III.2 : Une représentation d'une droite dans l'espace de Hough

III.3 Méthodes bidimensionnelles d'extraction de caractéristiques

III.3.1 L'analyse en composantes principales bidimensionnelle (ACP2D)

III 3.1.1 Principe

Proposée par **Yang** en **2004**, l'**ACP2D** est la première méthode d'extraction de signatures et de réduction de dimensions **basée sur** l'analyse en composantes principales (**ACP**) mais qui **traite directement les images** de visages comme des **matrices sans avoir à les transformer en vecteurs** comme l'exige les anciennes méthodes globales. Les auteurs de reportent que l'**ACP2D** présente une **performance élevée** en termes de reconnaissance, **de temps de calcul** et de **reconstruction des images** originales **comparée à Fisherfaces, ICA, Kernel PCA et Eigenfaces**. Cependant cette **supériorité n'a pas été théoriquement justifiée** dans leur article[**ROU07**].

Pour un ensemble d'apprentissage de **M** matrices de visages, l'idée de cette technique est de projeter une matrice **X** (**n × m**) via une transformation linéaire telle que:

$$Y_i = X.R_i \quad (\text{III.1})$$

Où Y_i est dit vecteur composante principale de dimension $(n \times 1)$.

R_i est le vecteur de projection de taille $(m \times 1)$. Le vecteur optimal R_i de la projection est obtenu en maximisant le critère de variance totale généralisé:

$$J(R) = R^T \cdot G_t \cdot R \quad (\text{III.2})$$

où G_t est la matrice de covariance des images de dimension $(m \times m)$ donnée par :

$$G_t = \frac{1}{M} \sum_{j=1}^M (X_j - \bar{X})^T (X_j - \bar{X}) \quad (\text{III.3})$$

avec X_j : la $j^{\text{ème}}$ image de la base d'apprentissage.

\bar{X} : l'image moyenne totale de toutes les images de la base d'apprentissage.

En général un seul axe de projection optimal n'est pas suffisant. On doit sélectionner un ensemble d'axes de projection tel que:

$$\begin{aligned} \{R_1, R_2, \dots, R_{d_1}\} &= \arg \max J(R) \\ R_i^T R_j &= 0, i \neq j, i, j = 1, \dots, d_1 \end{aligned} \quad (\text{III.4})$$

Ces axes sont les vecteurs propres de la matrice de covariance G_t correspondant aux « d_1 » plus grandes valeurs propres.

L'extraction des caractéristiques d'une image X par l'ACP2D se fait donc selon :

$$Y = X \cdot R \quad (\text{III.5})$$

Où $R = [R_1 \ R_2 \dots \ R_{d_1}]$ est la matrice de projection.

$Y = [Y_1 \ Y_2 \dots \ Y_{d_1}]$ est la matrice caractéristique de l'image X .

De façon similaire à la méthode des EigenFaces, l'ACP2D utilise, pour la reconstruction d'une image de visage, la matrice caractéristique (vecteurs composantes principales) et les vecteurs propres de la matrice G_t à partir de l'équation (III.5) comme suit:

$$\tilde{X} = Y \cdot R^T = \sum_{k=1}^{d_1} Y_k \cdot R_k^T \quad (\text{III.6})$$

Pour $\mathbf{d}_1 = \mathbf{m}$, on obtient l'image originale tandis que pour $\mathbf{d}_1 < \mathbf{m}$, la reconstruction est approximative mais meilleure que celle des EigenFaces [ROU07].

III 3.1.2 Choix du nombre de vecteurs propres

Le **nombre de vecteurs propres** associés aux plus grandes valeurs propres à retenir est **un grand défaut** de cette technique. Pour le choisir, les chercheurs ont adopté différentes solutions [ROU09] :

- Pour un ensemble de **115 images**, **Sirovitch** et **Kirby** ont trouvé que 40 EigenPictures sont suffisantes 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.

Pratiquement, le choix de « \mathbf{d}_1 » peut se faire soit en fixant un certain pourcentage \mathbf{K} tel que:

$$\frac{\sum_{i=\mathbf{d}_1+1}^n \lambda_i}{\sum_{i=1}^n \lambda_i} < K \quad (\text{III.7})$$

avec \mathbf{n} le nombre total des valeurs propres ou en se basant sur le fait que la $\mathbf{i}^{\text{ème}}$ valeur propre est égale à la variance le long de la $\mathbf{i}^{\text{ème}}$ composante principale et on cherche donc « \mathbf{d}_1 » le long du spectre des valeurs propres (Eigenvalue Spectrum) juste avant qu'il s'annule [ROU09].

III.3.2 L'analyse discriminante linéaire bidimensionnelle (ADL2D)

III 3.2.1 Principe

En s'inspirant de l'ACP2D, une autre méthode reposant sur l'ADL a été introduite avec différents noms à savoir : **ADL2D orientée** qui se présente en deux versions ; **ADL2DoC** (orientée en colonnes) et **ADL2DoL** (orientée en lignes) et **ADF2D** avec ses deux algorithmes; **LU2DFDA** et **RU2DFDA** (Left-multiplying Unilateral et Right-multiplying Unilateral 2D Fisher Discriminant Analysis respectivement). Cette méthode est plus performante que l'ADL et **corrige le problème** de la **singularité de la matrice de covariance intra- classes** ce qui la rend applicable pour le problème de petite taille (Small Sample Size problem).

Pour une base d'apprentissage de **M** matrices de visages regroupées en **C** classes où chaque classe contient **c n** images de visages de tailles (**n×m**) , le principe [ROU09] des deux versions de l'ADL2D, résumé dans les paragraphes suivants, consiste à projeter toutes les images de la base sur une matrice de projection en maximisant un certain critère[ROU07].

III 3.2.2 L'analyse discriminante linéaire bidimensionnelle orientée en lignes (ADL2DoL)

L'ADL2DoL cherche à maximiser les variances **inter-classes** et minimiser les variances **intra-classes** en appliquant une projection linéaire à droite de la matrice de visage **X (n×m)** sur une matrice **L(m× d₂)** telle que:

$$\mathbf{Z} = \mathbf{L}^T \cdot \mathbf{X} \quad (\text{III.8})$$

Où la matrice **W(n× d₂)** est dite matrice caractéristique (ou matrice de la signature associée à l'image **X**).

La matrice de projection **L** est obtenue en maximisant le **critère de Fisher** généralisé suivant:

$$\mathbf{L}^* = \underset{\mathbf{L} \in \mathbb{R}^{(m \times d_2)}}{\operatorname{argmax}} \frac{|\mathbf{L}^T \Sigma_b \mathbf{L}|}{|\mathbf{L}^T \Sigma_w \mathbf{L}|} \quad (\text{III.9})$$

Où Σ_b est la matrice de covariance **inter-classes** et Σ_w est la matrice de covariance **intra-classes** données respectivement par:

$$\Sigma_b = \sum_{c=1}^C n_c (\bar{X}_c - \bar{X})(\bar{X}_c - \bar{X})^T \quad (\text{III.10})$$

$$\Sigma_w = \sum_{c=1}^C \sum_{i=1}^{n_c} (X_i - \bar{X}_c)(X_i - \bar{X}_c)^T \quad (\text{III.11})$$

Avec : \bar{X}_c matrice moyenne de la classe c

\bar{X} : matrice moyenne totale (de toutes les matrices de la base d'apprentissage)..

Les colonnes de la matrice \mathbf{L} sont les vecteurs propres de la matrice $(\Sigma_w^{-1} \Sigma_b)$ associées aux « \mathbf{d}_2 » plus grandes valeurs propres.

III 3.2.3 L'analyse discriminante linéaire bidimensionnelle orientée en colonnes (ADL2DoC)

Dans ce cas, la projection de chaque matrice de visage \mathbf{X} est effectuée à gauche telle que $\mathbf{Z} = \mathbf{P}^T \cdot \mathbf{X}$ est la matrice caractéristique de dimension $(\mathbf{d}_2 \times \mathbf{m})$ où \mathbf{P} $(\mathbf{n} \times \mathbf{d}_2)$ est la matrice de **projection optimale** maximisant le **critère de Fisher** généralisé :

$$P^* = \arg \max_{P \in \mathcal{R}^{(n \times d_2)}} \frac{|P^T S_b P|}{|P^T S_w P|} \quad (\text{III.12})$$

ayant pour colonnes, les \mathbf{d}_2 premiers vecteurs propres de la matrice $(S_w^{-1} S_b)$ associés aux « \mathbf{d}_2 » plus grandes valeurs propres.

S_w et S_b désignent respectivement les matrices **intra** et **inter classes** généralisées de la base d'apprentissage:

$$S_w = \sum_{c=1}^C \sum_{i=1}^{n_c} (X_i - \bar{X}_c)^T (X_i - \bar{X}_c) \quad (\text{III.13})$$

$$S_b = \sum_{c=1}^C n_c (\bar{X}_c - \bar{X})^T (\bar{X}_c - \bar{X}) \quad (\text{III.14})$$

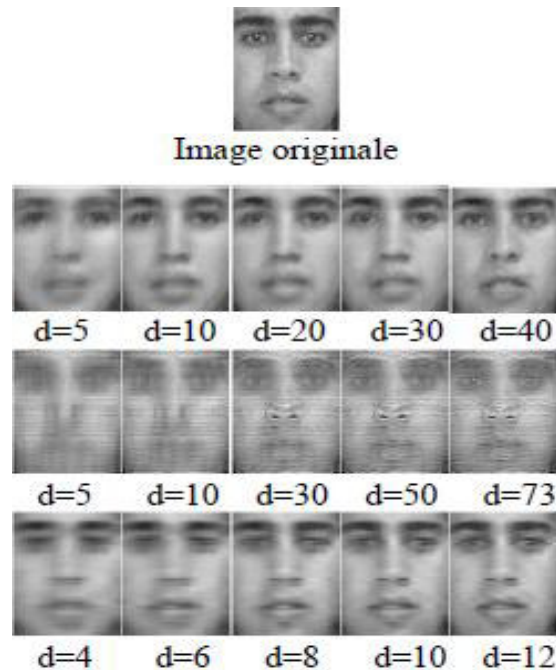


Figure III.3 : Comparaison entre les images reconstruites par : eigenfaces (1^{ère} ligne), ADL2DoC (2^{ème} ligne) et ACP2D (3^{ème} ligne).

III.4 Supériorité de l'ACP2D et de l'ADL2D orientée

Il a été démontré que l'ACP2D est l'ACP appliquée sur les lignes de toutes les images de visages et que LU2DFDA et RU2DFDA sont l'ADL appliquée respectivement sur les colonnes et les lignes des matrices de visages. Par conséquent, la supériorité de l'ACP2D sur l'ACP et celle de l'ADL2D sur l'ADL peut être justifiée par les arguments suivants [ROU07] :

- calcul direct et facile de la matrice de covariance de dimension $(m.m)$ ou $(n.n)$ au lieu de $(n.m, n.m)$ ou même $(M.M)$ comme l'a proposé Turk et Pentland pour réduire le temps et la complexité de calcul des EigenFaces.
- préservation de l'information spatiale bidimensionnelle .
- utilisation pour la classification de la somme des distances euclidiennes entre les vecteurs composantes principales formant la matrice caractéristique d'une image test et leur similaires des images d'apprentissage.

III.5 Inconvénients de l'ACP2D et de l'ADL2D orientée

Malgré les avantages offerts par ces deux méthodes, elles souffrent de quelques inconvénients à savoir [ROU07] :

- elles nécessitent plus de coefficients pour représenter les visages à reconnaître que l'ACP (EigenFaces) et l'ADL.
- elles effectuent une projection unilatérale (multiplication à droite ou à gauche) réduisant les redondances au niveau des lignes ou des colonnes séparément .
- elles dépendent du nombre de vecteurs propres à retenir qui lui-même dépend de la base de visages utilisée.

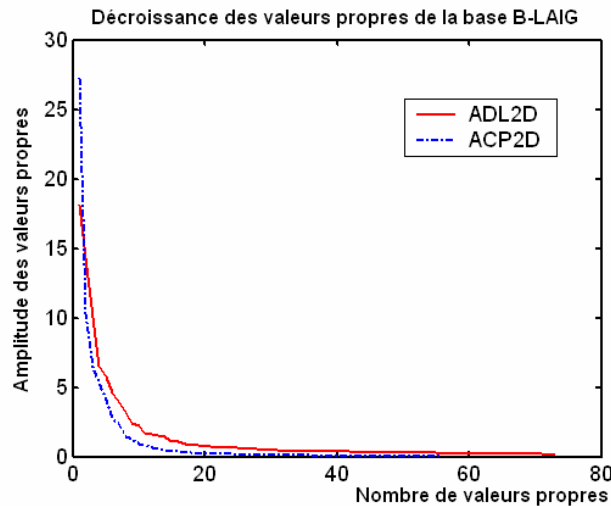


Figure III.4 : Décroissance des valeurs propres pour l'ACP2D et l'ADL2DoC

En réalité, le nombre de vecteurs propres associés aux **plus grandes valeurs propres à retenir est un grand défaut** de l'ACP qui, pour le choisir les chercheurs ont adopté différentes solutions: soit heuristiquement, soit en fonction des performances obtenues (erreur et taux) soit en fonction de l'allure des EigenFaces présentant des visages [ROU07].

III.6 Conclusion

Dans ce chapitre, nous avons essayé de présenter une conception d'un modèle de reconnaissance d'une modalité biométrique (visage) spécifique et efficace. Premièrement avec une détection de contours par la transformée de **Hough** pour la tâche de description de visage. Une explication du processus de classification est faite basée sur une approche bidimensionnelle pour l'extraction de caractéristiques la **PCA2D** et **LDA2D**.

Le chapitre suivant est consacré à l'implémentation de ce modèle ; ainsi que son application à l'identification et l'authentification de visage en utilisant la base de donnée XM2VTS.

IV.1 Introduction

Un système automatique de reconnaissance de visage est un système biométrique utilisant le visage à des fins d'identification et/ou de vérification de personnes à partir de leurs images de visages fixes ou de séquences vidéo en comparant les caractéristiques de cet individu avec celles stockées dans une base de référence **XM2VTS**. Dans le cadre de notre travail nous nous intéressons à une **Approche Bidimensionnelle Hybride** qui appartient à la **troisième classe** des méthodes de reconnaissance de visage. La particularité de notre méthode réside dans l'aspect **2D** des techniques utilisées et cela dans le souci de réduire davantage la taille des données à traiter. L'association des techniques **ACP2D** et **ADL2D** nous permet de travailler sur les matrices d'image sans avoir à les transformer en vecteur comme dans le cas de l'ACP .

Dans ce chapitre nous présentons les différentes étapes de l'implémentation d'un système d'identification basé sur l'Approche Bidimensionnelle Hybride **ACP2D** et **ADL2D**. Nous présentons d'abord la partie prétraitement basée sur la méthode de transformation de Hough, puis nous développons la partie d'extraction de paramètres par L'**ACP2D** associée à la **LDA2D** suivie d'une classification par une mesure de similarité en utilisant diverses distances métriques. Les résultats obtenus pour différents filtres et distances sont présents à la fin du chapitre IV. Et, finalement de le meilleur score est retenu pour un filtre et une distance bien choisis.

IV.2. Généralité sur les bases de données existantes

Plusieurs bases de données contenant des informations qui permettent l'évaluation des systèmes de reconnaissance de visages sont disponibles sur le marché. Toutefois, ces bases de données sont généralement adaptées aux besoins de quelques algorithmes spécifiques de reconnaissance. Le **tableau (IV.1)** récapitule les principales caractéristiques de ces bases .Il résume, pour chacune d'elle, les conditions d'acquisition des images de visages (changements d'illumination, de pose, d'expression faciale), la disponibilité ou non des données 3D et aussi nombre de sessions d'acquisition. Par ailleurs, plusieurs protocoles expérimentaux ont été élaborés afin d'évaluer le nombre important d'algorithmes de reconnaissance proposés ces dernières années. Des efforts particuliers de standardisation ont été aussi déployés pour définir des protocoles d'évaluation standardisés. Ainsi, la série

d'évaluations **FERET** menées par le « National Institute of Standards and Technologie (**NIST**) » a permis la comparaison entre neuf systèmes de reconnaissance de visages proposés par des institutions et des entreprises dans le contexte de l'identification et de la vérification. D'autres évaluations ont été effectuées par la suite, notamment celles du « Face Recognition Vendor Test (**FRVT**)» [SOU08].

Base de données	Nombre de personnes	Pose	Illumination	Facial Expression	Temps	Données 3D
AR	116	1	4	4	2	non
BANCA	208	1	++	1	12	non
CAS-PEAL	66-1040	21	9-15	6	2	non
CMU Hyper	54	1	4	1	1-5	non
CMU PIE	68	13	43	3	1	non
Equinox IR	91	1	3	3	1	non
FERET	1199	9-20	2	2	2	non
Havard RL	10	1	77-84	1	1	non
KFDB	1000	7	16	5	1	non
MIT	15	3	3	1	1	non
MPI	200	3	3	1	1	oui
NDHID	300	1	3	2	10/13	oui
NIST MID	1573	2	1	--	1	non
ORL	10	1	--	--	--	non
UMIST	20	--	1	--	1	non
U.Texas	284	--	1	--	1	non
U.Oulu	125	1	16	1	1	non
XM2VTS	295	--	1	--	4	oui
Yale	15	1	3	6	1	non
Yale B	10	9	64	1	1	non

Tableau IV.1 : Principales caractéristiques des bases de visages.

Le tableau contient le nombre de personnes enregistré, le nombre de vues sous des poses et conditions d'illumination différentes, ainsi que le nombre de sessions au cours desquelles des vues d'une même personne ont pu être collectées. Les cas où l'un des éléments n'a pas été mesuré, ou était non contrôlé durant la prise de vue, est noté.

En parallèle, les protocoles d'évaluation **XM2VTS** ont essentiellement été introduits pour l'évaluation des algorithmes multimodaux (voix, visage) et 3D, et ceci pour l'authentification uniquement. Pour toutes ces raisons nous utilisons pour notre application la base **XM2VTS** [SOU08].

IV.2.1 La base XM2VTS

C'est une base multimodale développée au sein du projet européen ACTS, elle contient des images fixes, des séquences vidéo et des images faciales 3D de 295 personnes. Elle est utilisée pour la vérification d'identité. La base de données XM2VTS a été réalisée sur une longue période ce qui a permis d'obtenir plusieurs images d'une même personne et donc une grande variabilité d'apparence (changement de coiffure, présence et/ou absence de lunettes, etc...). Par contre, seules les expressions neutres ont été considérées. La base de données XM2VTS contient quatre sessions. Une session "parole" où les personnes regardent vers le bas en lisant une phrase phonétiquement équilibrée. Une image de visage avec une bouche fermée a été choisie. Deux sessions avec et sans lunettes ont été acquises pour des sujets qui portent régulièrement des lunettes. La quatrième session correspond à une acquisition 3D de visages. Pour la tâche de vérification, un protocole standard d'estimation de performances a été développé. Appelé «**Lausanne protocol splits randomly**», il consiste à séparer les individus en deux classes, client et imposteur. Le groupe client contient 200 sujets, alors que le groupe imposteur est divisé en 25 imposteurs pour l'évaluation et 70 imposteurs pour le test. Huit images des quatre sessions sont utilisées.

La base de données est divisée en trois ensembles :

- **La base de données apprentissage** : L'ensemble apprentissage est l'ensemble de **référence**. Il permet de construire les modèles de **clients**. Il contient l'information concernant les personnes connues du système.
- **La base de données évaluation** : Enfin, l'ensemble évaluation est aussi utilisé pour les expériences de fusion pour l'apprentissage. Il permet de **fixer** les **paramètres** du système de reconnaissance de visage.
- **La base de données de test** : L'ensemble test est utilisé pour **calculer les scores** des clients et des imposteurs. En fonction de ces scores, **un seuil** est **choisi** afin de déterminer si une **personne est acceptée ou non**. D'après le protocole de Lausanne le seuil est choisi de telle manière à ce que l'ensemble de test satisfasse certains niveaux de performances. Finalement l'ensemble test est sélectionné pour simuler un scénario d'authentification réel où l'identité de l'imposteur est inconnue au système. Il permet de **tester le système** en lui présentant des images de personnes lui étant totalement inconnues.

Nous avons utilisée la base XM2VTS pour évaluer notre approche d'extraction du visage et de ses caractéristiques [OUA_MEH 09] [SOU08].

IV.2.1.1 Le protocole de Lausanne

Donc, ce protocole de Lausanne est lié directement à la vérification d'identité. Son principe est de diviser la base de données en deux classes, **200** personnes pour les clients, et **95** pour les imposteurs. Il partage la base de données en trois ensembles : Les imposteurs de l'ensemble de test ne doivent pas être connus du système, ce qui signifie qu'ils ne seront utilisés que pendant la toute dernière phase de test, lorsque le système est supposé fonctionnel et correctement paramétré. Il existe deux configurations différentes, la **configuration I** et la **configuration II**. Nous n'utiliserons la **configuration I** dans notre travail. Dans la **configuration I**, pour la formation de l'ensemble d'apprentissage trois images par client sont employées afin de créer les caractéristiques ou modèles clients. L'ensemble d'évaluation est constitué de trois autres images par client, ils sont utilisés essentiellement pour fixer les paramètres de l'algorithme de reconnaissance ou de vérification des visages. L'ensemble de test est formé par les deux autres images restantes. Pour la classe des imposteurs, les **95** imposteurs sont répartis dans deux ensembles : **25** pour l'ensemble d'évaluation et **70** pour l'ensemble de test. La répartition des images selon la configuration **I** est représentée par le Tableau **IV.2**.

Session	Pose	Clients	Imposteurs	
1	1	Apprentissage	Evaluation	Test
	2	Evaluation		
2	1	Apprentissage		
	2	Evaluation		
3	1	Apprentissage		
	2	Evaluation		
4	1	Test		
	2			

Tableau IV.2 : Répartition des images de la base de données selon la configuration I

Les tailles des différents ensembles de la base de données selon les deux configurations cités précédemment sont reprises dans le tableau **IV.3**.

Ensemble	Clients	Imposteurs
Apprentissage	600 images (3 par personne)	0 images
Evaluation	600 images (3 par personne)	200 images (8 par personne)
Test	400 images (2 par personne)	560 images (8 par personne)

Tableau IV.3 Répartition des photos dans les différents ensembles

Puisque notre application va se baser sur la comparaison d'images, donc il est important de savoir les nombres maximaux de comparaisons qu'ont peut atteindre. Selon la **configuration I**, et pour fixer les paramètres dans l'ensemble d'évaluation, on peut compter, **9** comparaisons par client (**1800** en tout) et **200** comparaisons par imposteur (**40000** en tout).

Ensemble	Clients	Imposteurs
Evaluation	9 par personne (1800 en tout)	200 par personne (40000 en tout)
Test	6 par personne (1200 en tout)	560 par personne (224000 en tout)

Tableau IV.4 : Nombre de comparaisons possibles

Un échantillon des images de la base de données est présenté dans les **figures (IV .1) et (IV .2)**.

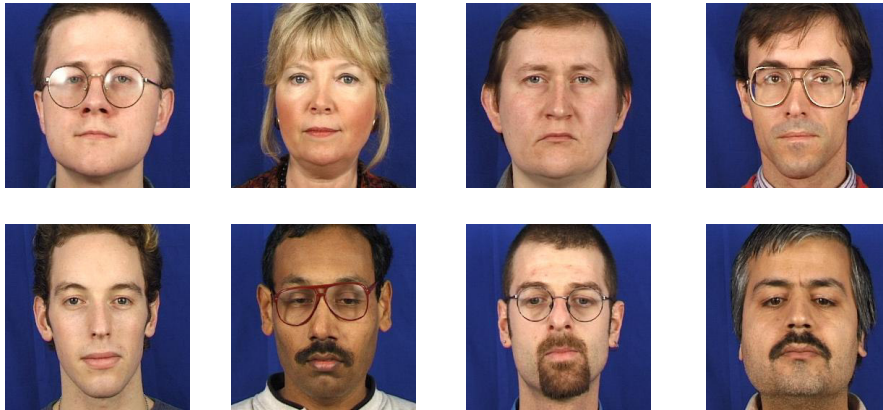


Figure IV. 1 : Exemple d'images extraites de la base XM2VTS.

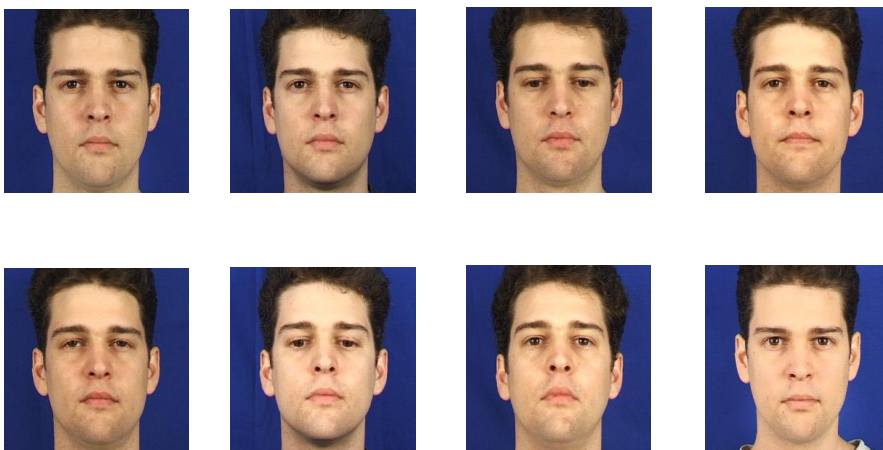


Figure IV.2: Exemples des images d'une personne de la base de données XM2VTS.

IV.3 Architecture structurelle d'un système d'identification

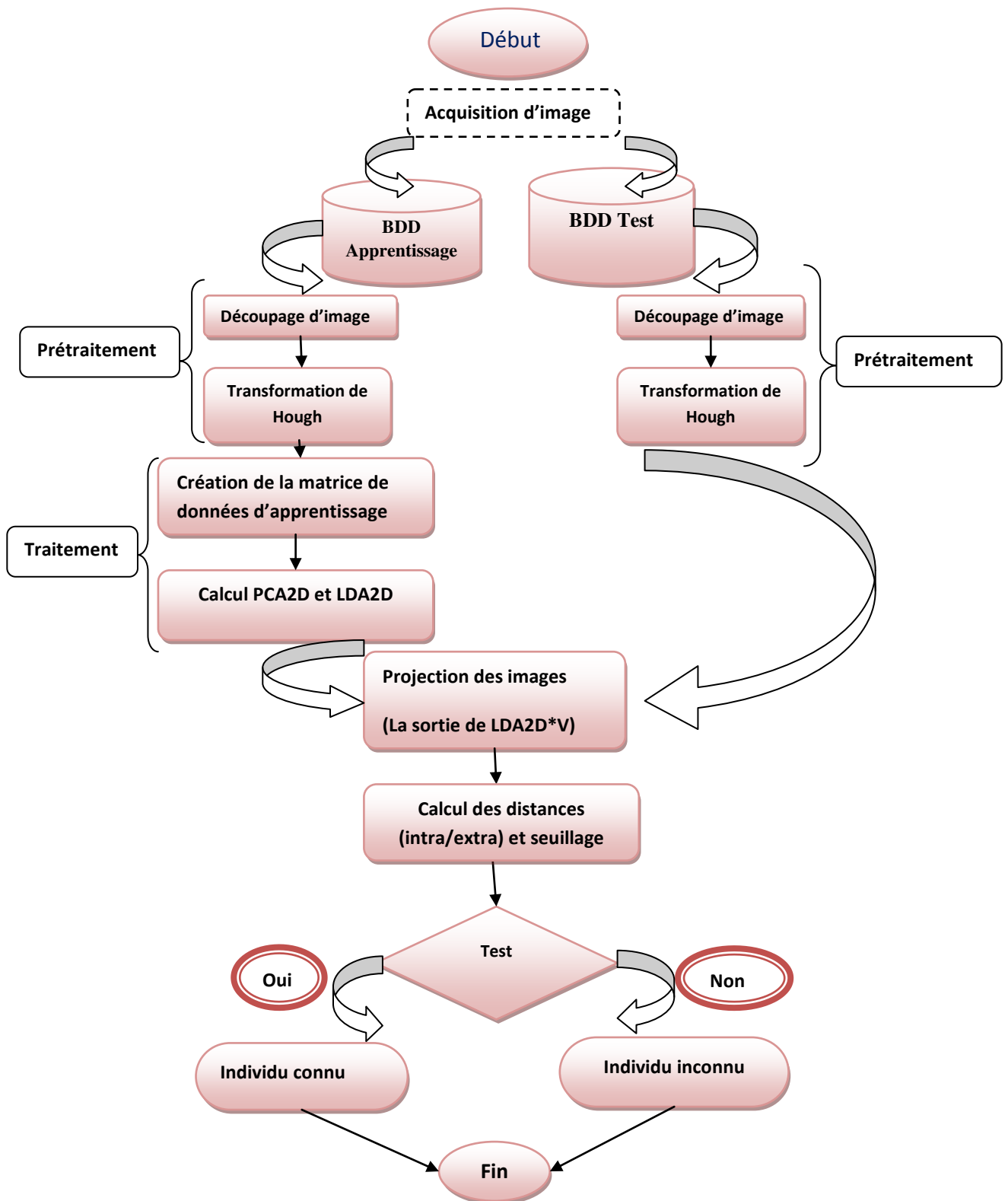


Figure IV.3 : Organigramme du Système de Vérification d'identité à l'aide d'images de Visages

IV.3.1 Prétraitement d'image

Le prétraitement est une phase importante dans le processus globale d'identification. C'est une méthode simple qui augmente en général les performances du système. Elle permet souvent une première réduction des données et elle atténue les effets d'une différence de conditions lors des prises de vues. Pour notre travail nous avons besoin de deux phase de prétraitement : Le découpage de l'image puis la conversion de la couleur en niveaux de gris et ceci dans le souci de la réduction de données.

a. Découpage des images et conversion Couleur → NG

Une image acquise habituellement appartient à l'espace RGB (une combinaison de trois couleurs : Rouge- Vert- Bleu). La première étape nécessaire à la réalisation de notre projet est la traduction de l'image à traiter en niveaux de gris. Pour ce faire, il suffit de travailler sur les couleurs RGB de l'image. On doit préalablement récupérer les trois teintes r, g et b; puis les égaliser grâce à la formule suivante :

$$Y = 0.3R + 0.59V + 0.11B \quad (\text{IV.1})$$

Enfin, il suffit de redonner à l'image ces nouvelles composantes R, G, B. Celle-ci est alors grise. Nous avons implémenté cette fonction dans l'unique but de pouvoir travailler sur un plus grand nombre d'images. On peut également utiliser directement la fonction '`rgb2gray`' disponible sur le logiciel **Matlab7.7** (R2008b). Le **découpage** de l'image consiste à conserver les maximums des variations intrinsèques du visage, et de supprimer les autres informations comme les cheveux et les éléments de l'habit afin d'améliorer les performances du système d'authentification.

Une fenêtre rectangulaire de taille (161x121) est centrée autour des caractéristiques les plus stables liées aux yeux, aux sourcils, au nez et à la bouche a été utilisée. La

Figure IV.4.

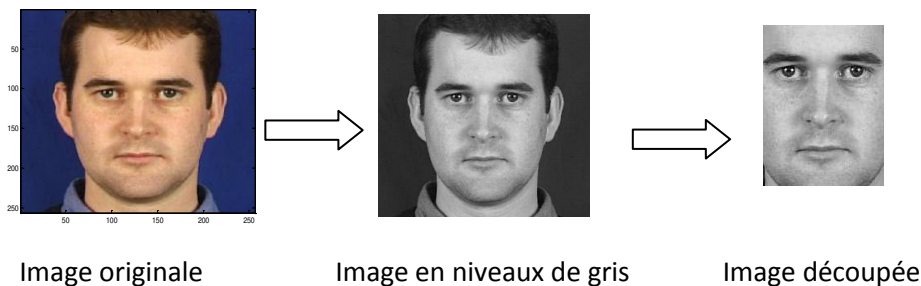


Figure IV.4 : Exemple de découpage d'une image

b. Filtrage

Le traitement des images faciales nécessite généralement un filtrage. Dans le cas particulier de détection de contours par transformation de Hough nous filtrons tout d'abord l'image en établissant un seuillage et le meilleur seuil est retenu.

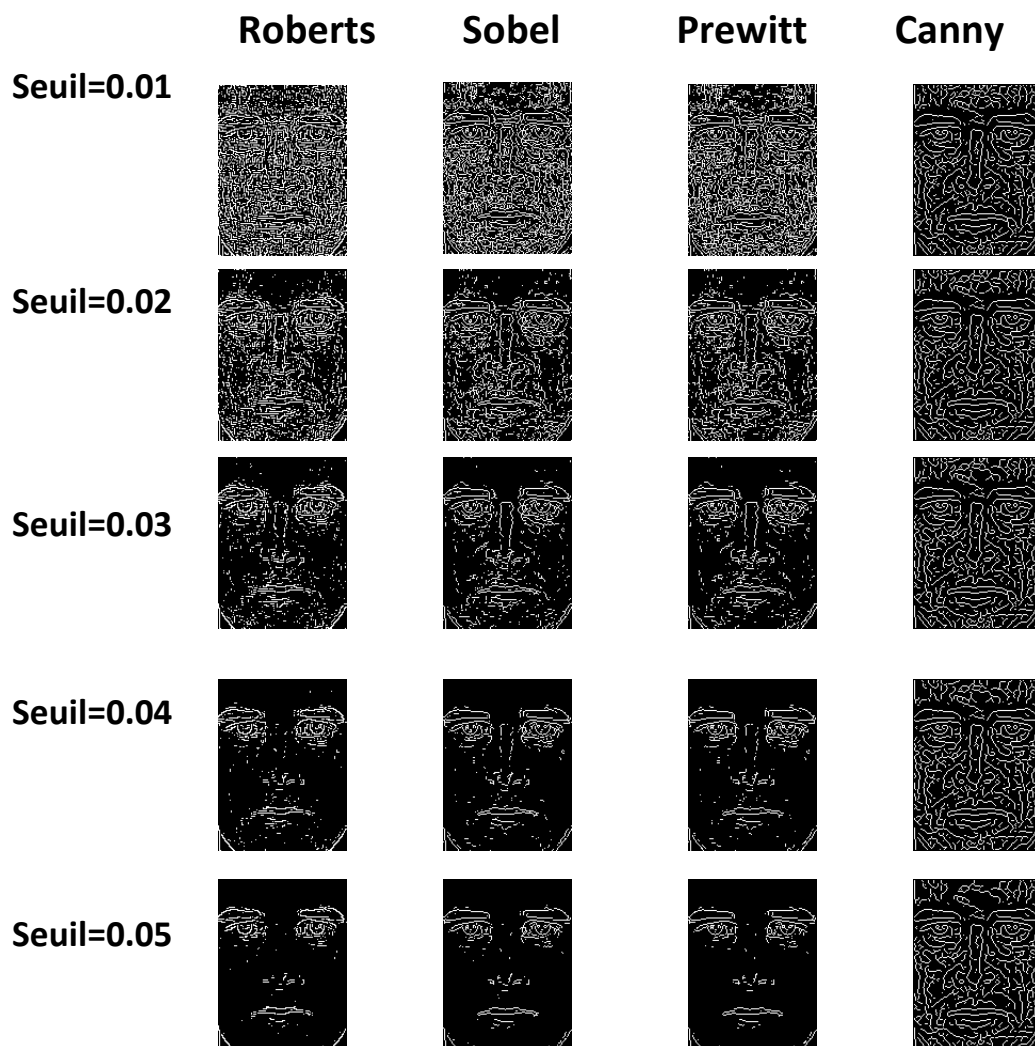


Figure IV.5 : L'effet de seuil dans chaque filtre

IV.3.2 Les descripteurs de l'image (Extraction de paramètres)

Dans cette partie, nous utilisons le descripteur suivant :

- Le descripteur basé sur la transformation de **Hough**. La **TH** est connue pour sa bonne détection des contours qui est une information essentielle aux types d'images à identifier

IV.3.2.1 Transformée de Hough

La transformée de Hough est une méthode permettant d'extraire et de localiser des groupes de points respectant certaines caractéristiques. La transformée de Hough est largement utilisée en traitement d'images.

IV.3.2.2 Algorithmes de la transformée de Hough

La transformée de **Hough** est un outil puissant dans l'analyse des formes et utilisée aussi pour l'extraction des traits globaux des formes dans l'image. Elle donne de bons résultats, même en présence de bruits dans l'image. La transformée de Hough (TH) opère sur des données binaires des points contours de l'image. La quantification du plan des paramètres ρ θ revient à quantifier l'intervalle $0 \leq \theta < \pi$ pour la dimension de θ et l'intervalle $-R \leq \rho \leq R$ pour la dimension de ρ , avec R la moitié de la diagonale de l'image.

ρ_k et θ_k pour pas de quantification des dimensions de ρ et θ .

n_θ et n_ρ le nombre de valeurs discrètes dans les intervalles de ρ et θ .

Les valeurs de ρ et θ discrétisées s'écrivent comme suit :

$$\theta = t * \theta_k \quad 0 \leq t \leq n_\theta \quad (\text{IV .2})$$

$$\rho = -R + r * \rho_k \quad 0 \leq r \leq n_\rho \quad \text{Avec : } n_\theta = \pi / \theta_k \quad \text{et} \quad n_\rho = 2R / \rho_k$$

Pour chaque point contour de coordonnées (x, y) de l'image, nous calculons pour chaque valeur discrète de θ la valeur discrète de ρ suivant l'équation suivante :

$$\rho = x * \cos\theta + y * \sin\theta \quad (\text{IV .3})$$

Ensuite, nous incrémentons la cellule correspondante dans le tableau accumulateur. Nous faisons la même chose pour tous les points contours de l'image.

L'organigramme ci-dessous de la figure décrit ce calcul. [OUA_MEH 09].

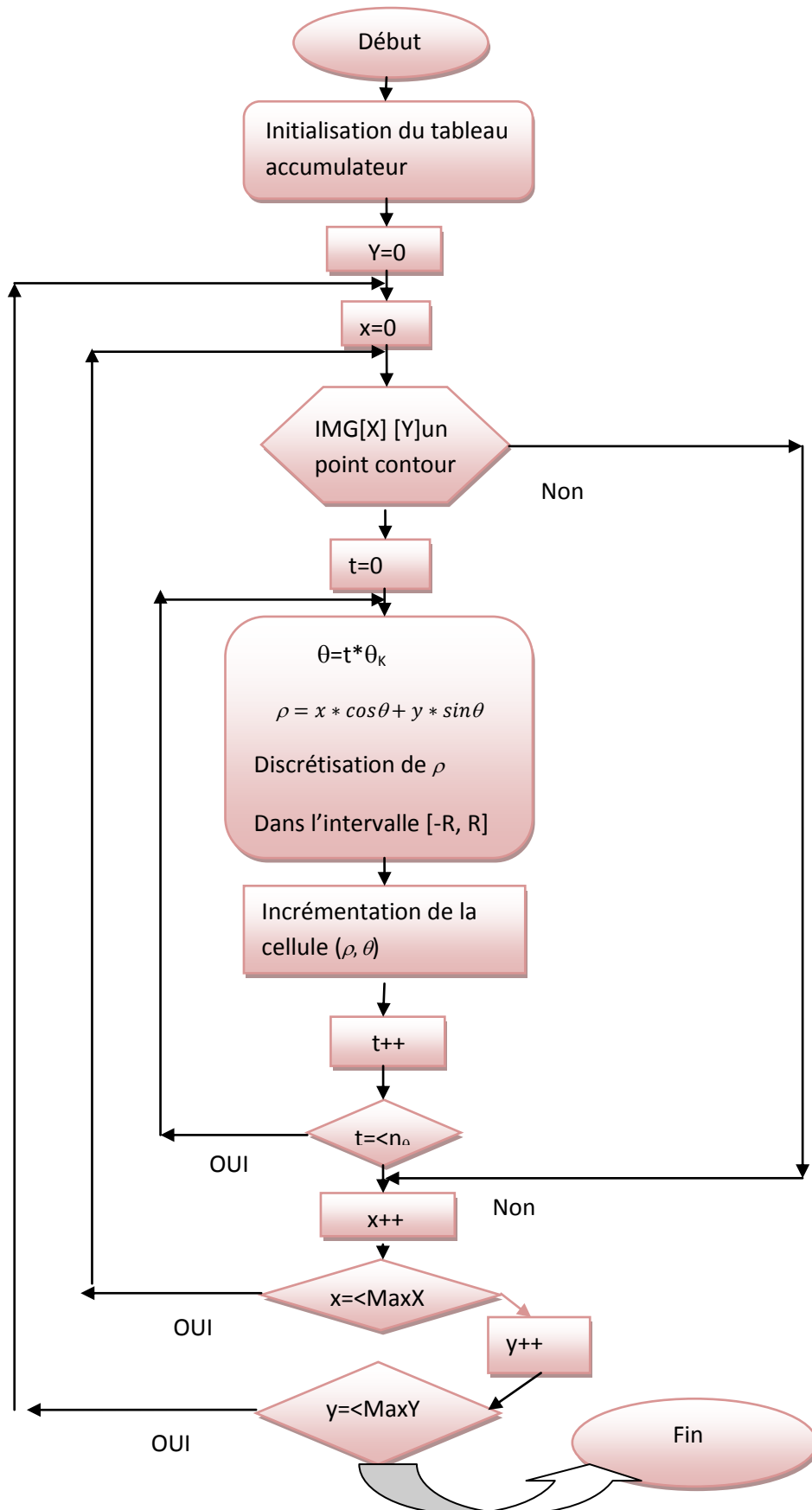


Figure IV.6 : Organigramme de la Transformée de Hough

IV.3.2.3 Photo normalisation

La photo normalisation a un double effet : d'une part elle supprime pour tout vecteur un éventuel décalage par rapport à l'origine et ensuite elle supprime tout effet d'amplification (multiplication par un scalaire). Pour chaque image nous effectuons l'opération suivante :

$$\text{photonormalisation}(x) = \frac{x - \text{mean}(x)}{\text{std}(x)} \quad \text{(IV.4)}$$

$\text{std}(x)$: l'écart type de la variable x .

Les opérations de prétraitement sur l'image d'entrée sont collectées dans une seule fonction nommée **prétraitement** () ; l'algorithme de cette fonction est donnée par :

% Y : image d'entrée (à traiter).

% prétraitement () : fonction de pratiquement.

Prétraitement (Y)

- Lire Y
- Transférer Y de RGB à niveau de gris (Z=rgb2gray(Y))
- Découper Z
- Filtrer Z par : (Roberts/Sobel/Canny)
X=edge (Z, 'Roberts', 0.03)
- Transformation de Hough
- Enchaîner (superposer) les lignes ou les colonnes de l'accumulateur de la TH (A)
- Normaliser A
- **Enregistrer A**

Algorithme IV.1 l'algorithme de la fonction de prétraitement pour la TH

IV.4 Création de la matrice de données d'apprentissage

IV.4.1 L'analyse en composantes principales bidimensionnelle (ACP2D)

Nous créons les matrices des images et l'enchaînement de leurs vecteurs. Nous procédons tout d'abord au calcul de la **ACP** à partir des images de la base de données apprentissage.

Calcul de l'ACP2D

Les étapes de calcul sont les suivants:

- 1- Prétraitement.
- 2- Créer la matrice de données X (des vecteurs images de l'ensemble d'apprentissage).
- 3- Calcul l'image moyenne totale de toutes les images de la base d'apprentissage.

$$\mu(n) = \frac{1}{M} \sum_{m=1}^M X_m(n) \quad (\text{IV.5})$$

$$\bar{X}_m = X_m - \mu(n) \quad (\text{IV.6})$$

- 4- Calcul la matrice de covariance

$$G_t = \frac{1}{M} \sum_{j=1}^M (X_j - \bar{X})^T (X_j - \bar{X}) \quad (\text{IV.7})$$

- 5- calcul les valeurs propres et vecteurs propres
- 6- lire des vecteurs par ordre décroissant des valeurs propres et prendre les d_1 premiers vecteurs propres.
- 7- calcul des vecteurs propres et normalisation.
- 8- Calcul de la base orthonormée $\mathbf{U}_{\text{ACP2D}}$.

Les étapes de calcul **ACP2D** sont assemblées dans une fonction nommée **ACP2D ()**

```

% P : nombre d'images d'apprentissage.
% Y : image d'apprentissage.
% prétraitement () : fonction de prétraitement.
% X : matrice de donnée d'apprentissage.

=====
• création de la matrice d'apprentissage.
Pour i=1 jusqu'à P
• X(i) = prétraitement (Y(i))
• Calcul l'image moyenne totale de toutes les images X
• Calcul la matrice de covariance  $G_t$ 
• Calcul les valeurs propres et vecteurs propres par
fonction eig (  $G_t$  ) .
• Lire des vecteurs par ordre décroissant des valeurs
propres Vprop.
• Calcul de la base orthonormée
 $U_{ACP2D} = X(i) * VpropACP$ .
• normalisation U.

```

Algorithme IV.2 : Algorithme de la fonction ACP2D

IV .4.2 L'analyse discriminante linéaire bidimensionnelle orientée en colonnes (ADL2DoC)

Calcul de la discrimination linéaire ADL2DoC

Les étapes de calcul sont les suivants:

- 1- Calcul matrice moyenne de la classe c \bar{X}_c .

$$\bar{X}_c = \frac{1}{3} \sum_{i=1}^M (X_i + X_{i+1} + X_{i+2}) \quad (\text{IV.8})$$

- 2- Calcul l'image moyenne totale de toutes les images de la base d'apprentissage.

$$\mu(n) = \frac{1}{M} \sum_{m=1}^M X_m(n) \quad (\text{IV.9})$$

$$\bar{X}_m = X_m - \mu(n) \quad (\text{IV.10})$$

3- Calcul la matrice S_w

$$S_w = \sum_{c=1}^C \sum_{i=1}^{n_c} (X_i - \bar{X}_c)^T (X_i - \bar{X}_c) \quad (\text{IV.11})$$

4- Calcul la matrice S_b

$$S_b = \sum_{c=1}^C n_c (\bar{X}_c - \bar{X})^T (\bar{X}_c - \bar{X}) \quad (\text{IV.12})$$

5- Résolution du problème généralisé entre S_B et S_w : $S_B V = \Lambda S_w V$.

6- Ordonner les vecteurs propres par ordre décroissant des valeurs propres de la matrice $J = S_w^{-1} * S_b$, et prendre les d_2 premiers vecteurs propres V_{ADL2D} .

7- Calcul de la base d'ADL2DOC : $V_{ADL2D} * U$.

Les étapes de calcul **ADL2DOC** sont assemblées dans une fonction nommée **ADL2DOC ()**.

L'algorithme de cette fonction est comme suit :

```

% X : matrice de donnée d'apprentissage.
% S_w : Withing class scatter.
% S_B Bitwin class scatter.
% V_ADL2D: les vecteurs propres de la matrice : J = S_w^{-1}*S_B

```

- Appel la base orthonormée U_{ACP2D} .
- Calculer la matrice S_w .
- Calculer la matrice S_B .
- Calculer V_{ADL2D}
- Calculer la base de Fisher $U_{ADL2D} = V_{ADL2D} * U$.
- **Enregistrer** U_{ADL2D} .

Algorithme IV.3 : Algorithme de la fonction **ADL2DoC ()**.

IV.4.3 Projection des images

Nous projetons les images d'apprentissage, les images d'évaluation et les images de test dans la base de Fisher calculée dans la phase précédente.

IV.4.4 Calcul des distances

Nous avons deux types de distances :

- La distance intra (**distance_intra()**) : mesure la distance entre deux images de visage de même individu.
- La distance extra (**distance_extra()**) : mesure la distance entre deux images de visage de deux individus différents.

Les distances employées dans notre expérience pour la mesure similarité sont de la distance Euclidienne (norme L1), distance Euclidienne (norme L2), Covariance, Mahalanobis et Corrélacion sont définis comme suit [BED08] :

A .La norme L1 : entre deux vecteurs A et B est la somme des valeurs absolue de la différence entre les composantes des deux vecteurs A et B. Elle est donnée par la relation suivant :

$$L1 = \sum_{i=1}^N |A_i - B_i| \quad (\text{IV.13})$$

B .La norme L2 : Connue aussi sous le nom norme euclidienne, c'est la somme du différence carré entre les composantes des deux vecteurs A et B. Elle est donnée par l'équation suivante

$$L2 = \sum_{i=1}^N (A_i - B_i)^2 \quad (\text{IV.14})$$

C. Covariance

La covariance mesure l'angle entre les deux vecteurs normaux A et B. Elle est donnée par la relation suivante :

$$\text{Cov}(A, B) = \frac{A \cdot B}{\|A\| \cdot \|B\|} \quad (\text{IV.15})$$

D. Distance de Mahalanobis

C'est la somme du produit des composantes des deux vecteurs A et B et les valeurs propres d'une dimension spécifique. Elle est donnée par l'équation suivante :

$$\text{Mah}(A, B) = - \sum_{i=1}^N A_i B_i C_i \quad (\text{IV.16})$$

Avec : $C_i = 1/\lambda_i$

E.Corrélation

Elle mesure le taux de changement entre les composantes de deux vecteurs A et B .

Elle est donnée par la relation :

$$Corr(A, B) = \sum_{i=1}^N \frac{(A_i - \mu_A)(B_i - \mu_B)}{\sigma_A \sigma_B} \quad (\text{IV.17})$$

Où : σ_A = l'écart type de A , μ_A = le moyenne de A

σ_B = l'écart type de B , μ_B = le moyenne de B

IV.4.5 Classification

Dans le problème de vérification d'identité, nous cherchons à définir, pour chaque personne, ou de manière globale un seuil .Ce seuil va déterminer le minimum de ressemblance entre deux images pour admettre qu'il s'agit de la même personne. Ce minimum de ressemblance va s'exprimer comme une distance maximale entre les caractéristiques des deux images.

Pour fixer se seuil nous allons utiliser l'ensemble d'évaluation.

Théoriquement $\max(\text{distance intra}) < \min(\text{distance extra})$ dans cette condition, il est possible de déterminer un seuil S qui détermine une classification parfaite (sans erreurs $TEE=0$).

Il suffit de prendre :

$$S = \frac{\max(\text{distance intra}) + \min(\text{distance extra})}{2} \quad (\text{IV.18})$$

Malheureusement cette situation ne se rencontre que très rarement en pratique.

Par conséquent, un compromis doit être fait dans le choix du seuil, et ce dernier est très dépendant de l'application: **FA** doit avoir une valeur très faible dans les applications bancaires afin de garantir plus de sécurité. Généralement, un système d'authentification opère entre les deux extrêmes pour la majorité des applications: on ajuste le seuil de décision pour atteindre une valeur désirée et prédéfinie de **FR** ou de **FA**. Donc nous devons choisir un critère à minimiser pour fixer le seuil de manière à atteindre le **TEE** ($TFA=TFR =TEE$, Taux d'erreur égale) [OUA_MEH 09].

Dans notre expérience nous choisissons le seuil de manière à atteindre TEE tel que :

$$(TFA - TFR) < 10^{-3} \quad (\text{IV. 19})$$

L'algorithme suivant permet de calculer : TEE et le seuil

```

% DIS_cl : intervalle des distances intra
% DIS_imp : intervalle des distances extra
% TFA : Taux de fausse acceptations.
% TFR : Taux de faux rejets.
% seuil : le seuil pour avoir TFA=TFR.

=====

% Algorithme de seuillage
max_int =max (max(DIS_cl),min(DIS_imp));
min_int=min (max(DIS_cl),min(DIS_imp));
Seuil= (min_int+max_int)/2;
TFA= (length (find (DIS_imp<seuil))/length(DIS_imp))*100;
TFR= (length (find (DIS_cl>seuil))/length(DIS_cl))*100;
Tanque (abs(TFA-TFR)> 10(-3))
    si TFA>TFR
        max_int=seuil;
        seuil=( max_int+min_int)/2;
    sinon
        min_int=seuil;
        seuil=( min_int+max_int)/2;
    fin
    TFA=(length(find
(DIS_imp<seuil))/length(DIS_imp))*100;
    TFR= (length(find (DIS_cl>seuil))/length(DIS_cl))*100;
fin

```

Algorithme IV.4 : Algorithme de calcul de TFA, TFR et seuillage.

IV.4.6 Décision

Dans un processus d'authentification, la stratégie de décision nous permet de choisir entre les deux alternatives suivantes: l'identité de l'utilisateur correspond bien à l'identité de la personne proclamée (ou recherchée) ou elle ne correspond pas. En comparant le modèle de l'utilisateur à celui qui existe dans la base de données. Cette phase est basée sur le seuil prédéfini [OUA_MEH 09].

L'algorithme IV.5 explique le processus de la phase de décision

```
% imtest : image de test (image de l'utilisateur)
% X : model de la personne proclamée stocké dans la
base des donnés
% DIS : distance entre X et Y. (projetées dans le même
sous espace) en utilisant l'une des norme de mesure de
similarité.
```

```
=====
```

- Lire imtest
- $Y = \text{praitraitement}(\text{imtest})$
- Projeter Y
- Seuillage
- $\text{DIS} = \text{mesurer distance entre } X, Y$

Si $\text{DIS} < \text{seuil}$

Personne connue (identité parfaite)

Sinon

Personne inconnue (identité non parfaite)

Fin

Algorithme IV.5 : Algorithme de test (décision).

Le programme ainsi conçu se présente sous plusieurs fonctions, l'essentiel est présenté sous la forme suivante :

- **Lire()** pour lire l'ensemble des images d'apprentissage, d'évaluation et de test.
- **Prétraitement()** pour préparer les images ainsi lues à la phase de traitement.
- **ADL2DoC ()**.cette fonction renvoie une base orthogonale (**Fisher basis**) dans laquelle on projette toutes les images d'apprentissage, d'évaluation et de test.
- **distance_intra()** renvoie les valeurs des distances entre les images des clients évaluation et les images d'apprentissage.
- **distance_extra()** renvoie les valeurs des distances entre les images des imposteurs évaluation et les images d'apprentissage.
- **seuillage ()** cette fonction sert à estimer les valeurs de **TEE** et le seuil.
- **Décision ()** pour Accepter ou Rejeter la proclamation associée à une personne donnée.

IV.5 Résultats

En effet, il existe différentes fonctions de mesure de score on peut légitimement se poser la question : quel fonction de mesure choisir ? Pour répondre à cette question nous avons effectué nos expériences sur plusieurs fonctions de mesure pour pouvoir réaliser une étude comparative et retenir les meilleurs résultats. Nous avons présenté ces derniers avec une méthode basique l'Eigen Faces. Notre programme principal se présente comme suit :

- * Prétraitement avec photo normalisation
- * En applique Approche Bidimensionnelle Hybride **ACP2D** et **ADL2D**.
- * Mesure des distances: Euclidienne norme L1, Euclidienne norme L2, covariance, Mahalanobis et corrélation.
- * Seuillage : Globale.

IV.5.1 Extraction des paramètres avec l'ACP2D et ADL2DOC

Avant de réaliser la transformation de Hough, nous effectuons un filtrage en utilisant le filtre de **Roberts** qui s'avère le meilleur sur le plan fonction de mesure de score d'après une étude menée par **A.OUAMANE et A.MEHDAOUI** [1].

Nous cherchons à trouver le nombre de paramètres caractéristiques **Np** de l'ACP2D et de la ADL2DOC donnant le meilleur score en utilisant différentes

distances : Euclidienne norme L_1 , Euclidienne norme L_2 , covariance, Mahalanobis et corrélation. Nous fixons le nombre de paramètre **ACP2D** et faisons varier le nombre de paramètres **ADL2DOC** pour déterminer la meilleure fonction de mesure de score. Par exemple, le choix des paramètres suivants :

ACP2D	60	80	100	200
ADL2DOC	10	10	10	10
	20	20	20	30
	30	40	40	50
	40	50	50	80
	50	60	80	120
	60	70	90	150

Tableau IV.5 : Le tableau des paramètres **ACP2D** et **ADL2DOC**

- **Choix du nombre de paramètres et de la distance**

Nous étudions notre système de reconnaissance de visage et fixons les différents paramètres dans la BDD évaluation.

1) Cas $N_p(\text{ACP2D})=60$

Nous essayons d'étudier notre système en maintenant le nombre de paramètres **ACP2D** à **60** et dans ce cas bien précis nous faisons **varier** comme indiqué sur les quatre tableaux les paramètres **LDA2D**. Et, pour chaque distance nous calculons le **TFA** et **TFR**.

Avec

Le taux de réussite **TR** = $(100 - (\text{TFR} + \text{TFA}))$. (IV.20)

FR =(false rejection) rejeter faussement un client

FA =(false acceptation) accepter faussement un imposteur

$$\text{TFR} = \frac{\text{Nb de FR}}{\text{Nb de clients}} \quad (\text{IV.21})$$

$$\text{TFA} = \frac{\text{Nb de FA}}{\text{Nb d' imposteurs}} \quad (\text{IV.22})$$

$$\text{TEE} = \frac{\text{Nb de FR} + \text{Nb de FA}}{\text{Nb total d'accès}} \quad (\text{IV.23})$$

LES DISTANCES								
Euclidienne norme L_1					Euclidienne norme L_2			
ADL2D	TFA %	TFR %	TEE %	TR %	TFA %	TFR %	TEE %	TR %
10	5.0825	5	5.0091	89.9175	4.7650	4.8333	4.8257	90.4016
20	4.4000	4.3333	4.3407	91.2666	3.7775	3.8333	3.8271	92.3891
30	4.3525	4.3333	4.3354	91.3141	3.6925	3.6666	3.6695	92.6408
40	4.7575	4.6666	4.6767	90.5758	3.9575	4	3.9952	92.0425
50	4.3200	4.3333	4.3318	91.3466	3.9400	4	3.9933	92.0600
60	4.8825	4.8333	4.8387	90.2841	3.9100	4	3.9900	92.0900

Tableau IV.6 : Le tableau de ACP2D=60 et ADL2DOC= [10 20 30 40 50 60].

La distance présentant les meilleures performances est maintenue. Le meilleur résultat est obtenu pour la distance L_2 pour un nombre de paramètre **60(ACP2D)** et **30(ADL2D)** avec un **TEE=3.6695%** et un **TR=92.6408%**.

LES DISTANCE												
	Covariance				Mahalanobis				Corrélation			
ADL2D	TFA%	TFR%	TEE%	TR%	TFA%	TFR%	TEE%	TR%	TFA%	TFR%	TEE%	TR%
10	5.8475	5.8333	5.8349	88.3191	22.1100	22.1666	22.1603	55.7233	6.7950	6.7950	6.8290	86.3716
20	5.3500	5.3333	5.3351	89.3166	14.9200	14.8333	14.8429	70.2466	5.5675	5.6666	5.6556	88.7659
30	5.1475	5.1666	5.1645	89.6858	12.2450	12.1666	12.1753	75.5883	5.2600	5.3333	5.3251	89.4066
40	5.4050	5.5000	5.5097	89.0950	10.0800	10	10.0088	79.9200	5.5350	5.5000	5.5038	88.9650
50	5.5875	5.5000	5.4894	88.9125	9.75000	9.6666	9.6759	80.5833	5.5725	5.6666	5.6562	88.7608
60	5.4200	5.3333	5.3429	89.2466	9.5825	9.6666	9.6573	80.7508	5.2425	5.1666	5.1750	89.5908

La suite du tableau IV.6

Le plus mauvais score est obtenu pour la distance de Mahalanobis.

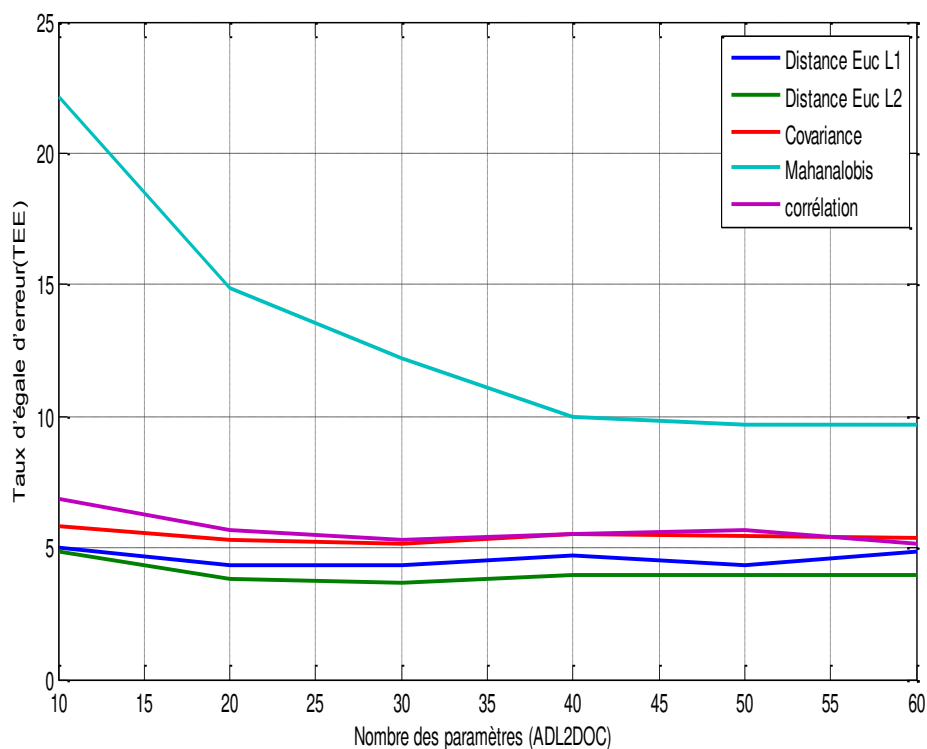


Figure IV.7 : Taux d'égalité erreur dans l'ensemble d'évaluation pour différentes fonctions de mesure de score Cas $N_p(\text{ACP2D})=60$

La figure IV.7 illustre les courbes de TEE en fonction du nombre de paramètres d'ADL2D pour un N_p d'ACP2D égal à 60. Nous confirmons que la distance maintenue est bien la distance Euclidienne L_2 .

2) Cas $N_p(ACP2D)=80$

Nous passons ensuite à l'étude des performances du système de reconnaissance en fixant cette fois ci le nombre de paramètres à 80.

LES DISTANCE										
	Euc L ₁		Euc L ₂		Cov		Mah		Cor	
ADL2D	TEE%	TR%	TEE%	TR%	TEE%	TR%	TEE%	TR%	TEE%	TR%
10	5.3387	89.2841	5.3254	89.4041	7.1595	85.7308	21.3301	57.3616	7.3251	85.4066
30	4.5108	90.9025	4.1581	91.7433	5.1578	89.7458	12.6645	74.6858	4.8287	90.3741
50	5.1575	89.7483	4.3243	91.4141	5.1600	89.7258	9.3304	81.3591	5	90
60	5.1670	89.6633	4.1670	91.6633	4.9952	90.0425	8.1734	83.6058	4.9891	90.0900
70	5.5088	88.9200	4.1770	91.5733	4.1770	89.7383	8.1681	83.6533	5.1662	89.6708

Tableau IV.7 : Le tableau total de ACP2D=80 et ADL2DOC= [10 30 40 50 60 70]

Le meilleur résultat est obtenu pour un nombre de paramètres $ADL2D = 30$ et dans le cas de la distance Euclidienne L_2 avec un $TEE=4.1581\%$, $TR= 91.7433\%$ les mauvais résultats sont encore une fois donné par la distance de **Mahalanobis**.

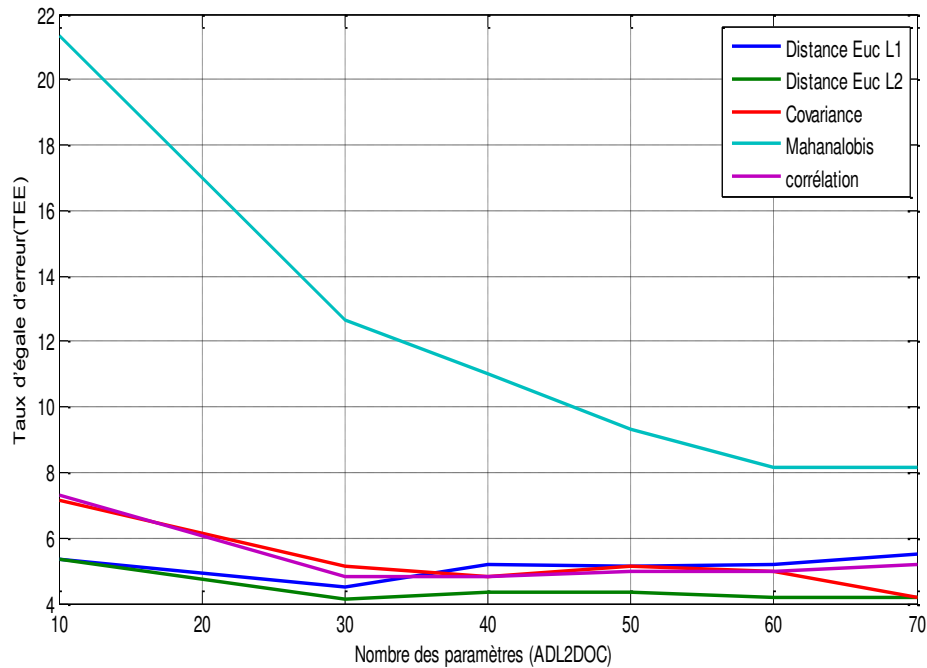


Figure IV.8 : Taux d'égalité d'erreur dans l'ensemble d'évaluation pour différentes fonctions de mesure de score Cas $N_p(ACP2D)=80$.

La **figure IV.8** confirme la déduction que nous avons faite auparavant.

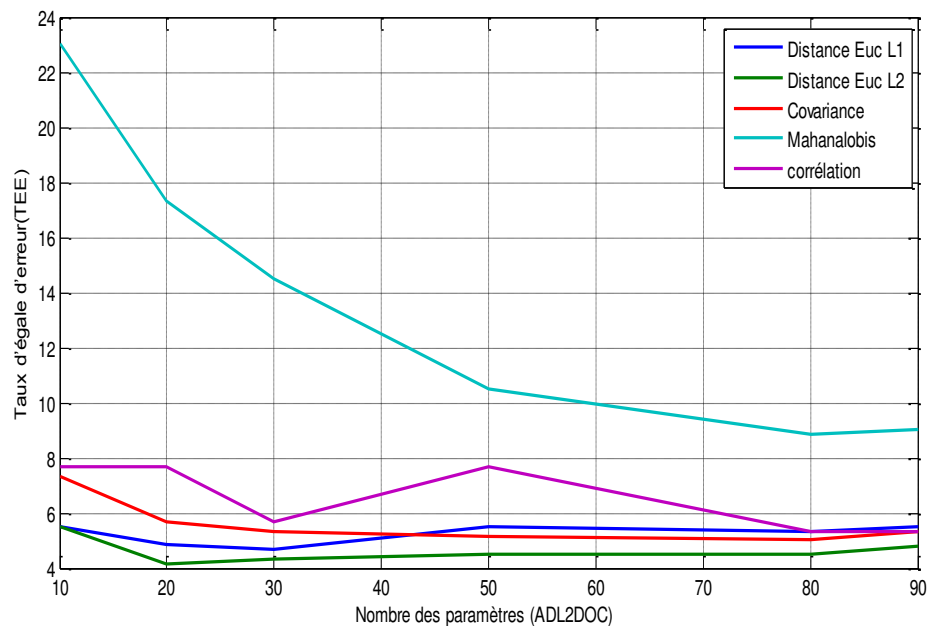
3) Cas $N_p(AC P2D)= 100$

Dans le cas de l'ACP2D=100 et l'ADL2DOC= [10 20 30 50 80 90].

LES DISTANCES										
	Euc L ₁		Euc L ₂		Cov		Mah		Cor	
ADL2D	TEE%	TR%	TEE%	TR%	TEE%	TR%	TEE%	TR%	TEE%	TR%
10	5.5108	89.2841	5.4911	89.4041	7.3229	85.7308	23.0011	57.3616	7.6745	85.4066
20	4.8329	90.3366	4.1706	91.6308	5.6659	88.6733	17.3329	65.3366	7.6745	87.7083
30	4.6684	90.6508	4.3229	91.4266	5.3376	89.2941	14.5061	70.9450	5.6659	88.6733
50	5.4988	89.0100	4.5094	90.9149	5.1714	89.6233	10.4902	79.0875	7.6745	89.4291
80	5.3251	89.4066	4.4980	91.0175	5.0058	89.8895	8.82824	82.3791	5.3262	89.3966
90	5.4961	89.0350	4.8235	90.4216	5.3299	89.3641	9.0002	81.9975	5.3412	89.2616

Tableau IV.8 : Le tableau total de ACP2D=100 et ADL2DOC= [10 20 30 50 80 90].

Le résultat le meilleur dans ce cas c'est lorsque le nombre de paramètre **ACP2D** est



égal **100** et **ADL2D= 20** avec un **TR=91.6308%**, **TEE=4.1706%**

Figure IV.9 : Taux d'égale erreur dans l'ensemble d'évaluation pour différentes fonctions de mesure de score Cas $N_p(\text{ACP2D})=100$.

La distance L_2 reste celle qui nous donne les meilleures performances comme la montre bien la **figure IV.9**.

4) Cas $N_p(ACP2D)=200$

Le cas de $ACP2D=200$ et $ADL2DOC= [10\ 30\ 50\ 80\ 120\ 150]$.

LES DISTANCE										
ADL2D	Euc L ₁		Euc L ₂		Cov		Mah		Cor	
	TEE%	TR%	TEE%	TR%	TEE%	TR%	TEE%	TR%	TEE%	TR%
10	8.9888	82.1000	7.8268	84.3916	9.4969	81.0275	25.3329	49.3366	10.1639	79.6908
30	7.8232	84.4241	6.6559	86.7633	7.1620	85.7083	13.1623	73.7058	7.4961	85.0350
50	8.3349	83.3191	6.8235	86.4216	6.5030	86.9725	9.3237	81.4191	9.3237	87.0100
80	8.3362	83.3066	7.337	85.2966	6.0033	87.970	7.4913	85.0775	6.1567	87.7558
120	7.99694	84.0275	6.5097	86.9125	5.8274	88.3866	7.5011	84.9900	5.9922	88.0700
150	9.4997	81.0025	6.9980	86.0175	5.6659	88.6733	8.3265	83.3941	5.8385	88.2866

Tableau IV.9 : Le tableau total de $ACP2D=200$ et $ADL2DOC= [10\ 30\ 50\ 80\ 120\ 150]$.

Le meilleur score est obtenu contrairement aux autres cas pour la distance covariance qui nous donne un **TR= 88.6733%** et un **TEE= 5.6659%** avec un nombre de paramètre **ACP2D=200 ADL2D=150**.

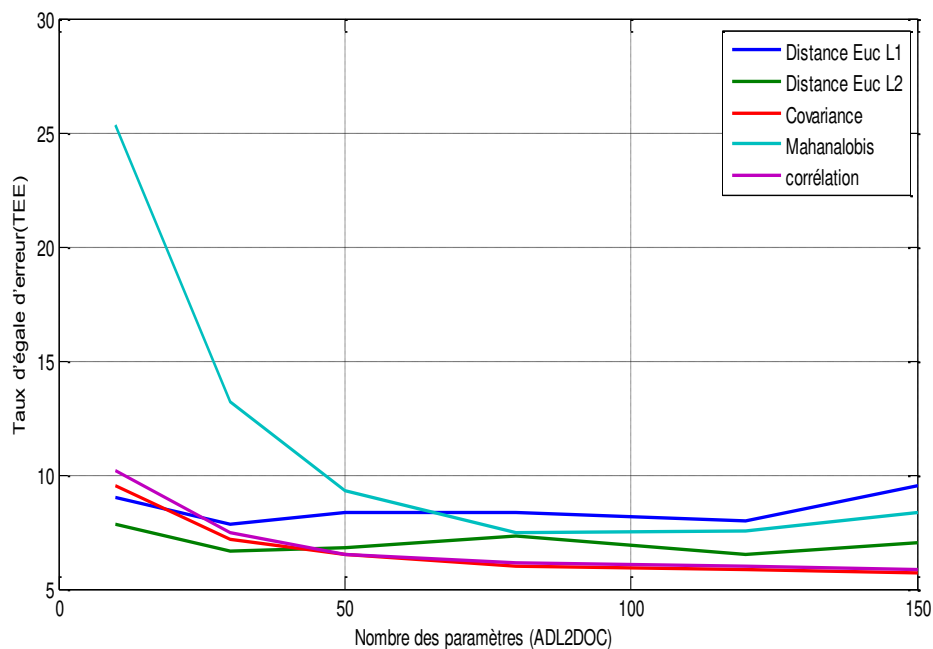


Figure IV.10 : Taux d'égale erreur dans l'ensemble d'évaluation pour différentes fonctions de mesure de score Cas $N_p(\text{ACP2D})=200$.

Les courbes confirment les résultats obtenus.

5) Récapitulation

Nous essayons de présenter tous les résultats pour diverses distances et un nombre de paramètres ACP2D variable appartenant à l'intervalle [60 : 200]. La **figure IV.11** montre bien que la distance **L₂** reste la meilleure pour le **Np(ACP2D)=60** et un **Np(ADL2D)= 30**.

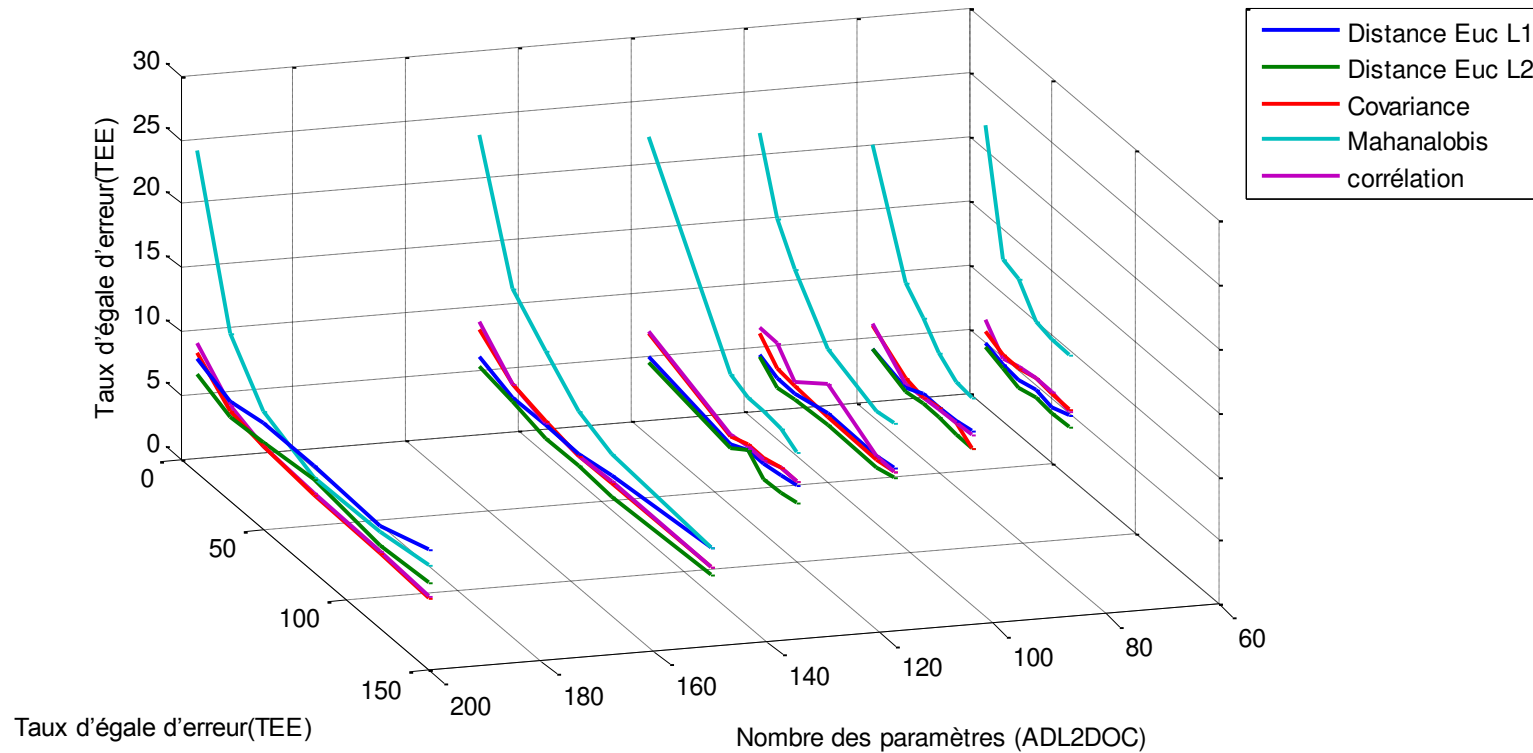


Figure IV.11 : Taux d'égalité d'erreur dans l'ensemble d'évaluation pour différentes fonctions de mesure de score ACP2D varie et ADL2DOC varie

Finalement, nous pouvons dire que la distance Euclidienne L_2 reste la meilleure car elle présente les meilleurs scores pour un nombre de paramètre très limité $N_p(\text{ACP2D}) = 60$ et $N_p(\text{ADL2D}) = 30$ avec $\text{TEE} = 3.6666\%$, $\text{TR} = 92.6408\%$ contre un $\text{TR} = 88.6733\%$, $\text{TEE} = 5.6659\%$ pour la distance de covariance nécessitant un nombre de paramètre plus important $N_p(\text{ACP2D}) = 200$ et $N_p(\text{ADL2D}) = 150$. Notons que d'après les expériences effectuées que en augmentant le nombre de paramètre et en maintenant la distance L_2 le TEE augmente et le TR diminue. Donc, nous pouvons dire que la norme Euclidienne L_2 est la meilleure des distances étudiées et présente le meilleur score dans le cas du modèle choisi pour un nombre de paramètre $N_p(\text{ACP2D}) = 60$ et $N_p(\text{ADL2D}) = 30$.

IV.5.2 Le test du système de reconnaissance de visage

Pour cela nous utilisons la base de test qui contient des individus totalement inconnus à notre système. Et, nous essayons de calculer les nouveaux scores pour tester la stabilité du système. D'après les résultats obtenus dans IV.5.1 nous choisissons L_2 et le nombre de paramètre fixés à $N_p(\text{ACP2D})=60$ et $N_p(\text{ADL2D})=30$.

fonctions de mesure de score	Ensemble évaluation		Ensemble de test				nombre de paramètres	
	TEE %	TR %	TFA %	TFR %	TEE %	TR %	ADL	ACP
Euc L2	4.8257	90.4016	5.0261	6	5.7475	88.9738	10	60
Euc L2	3.8271	92.3891	4.2389	6	5.5434	89.7610	20	60
Euc L2	3.6695	92.6408	4.4176	5.5000	5.2193	90.0823	30	60
Euc L2	3.9952	92.0425	4.8072	5.5000	5.3203	89.6927	40	60
Euc L2	3.9933	92.0600	4.4959	6	5.6100	89.5040	50	60
Euc L2	3.9900	92.0900	4.2991	6	5.5590	89.7008	60	60

Tableau IV.10 : comparaison entre l'ensemble d'évaluation et l'ensemble de test

Le tableau montre que le système est bien stable vu que les paramètres obtenus pour la Base de donnée évaluation st assez proches de ceux obtenus pour la Base de donnée test. La différence trouvés entre le $TEE=3.6695\%$ et le $TEE_t=5.2193\%$ ($TR=92.6408\%$; $TR_t=90.0823\%$) est dû au fait que la BDD test contient plus d'imposteurs à peu près trois fois.

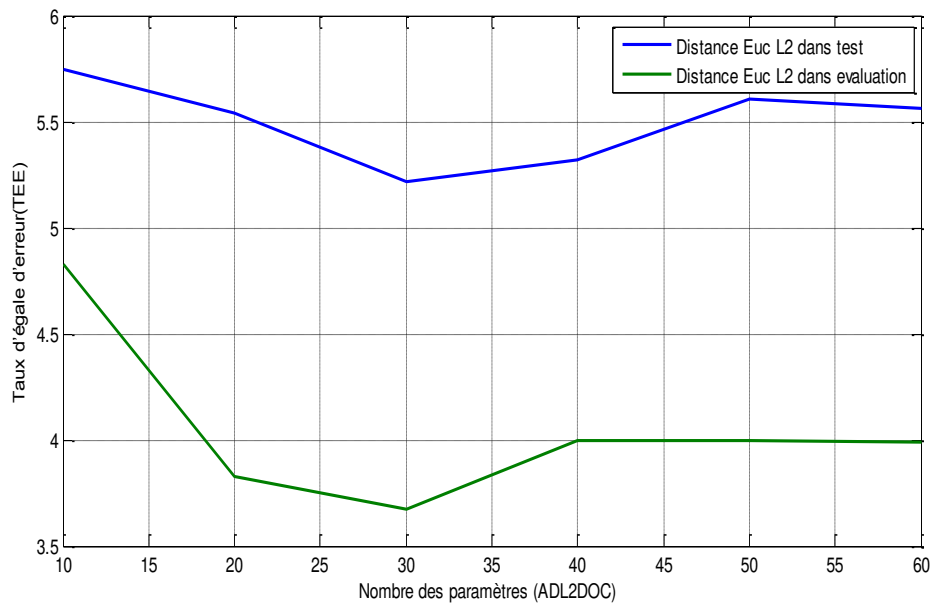


Figure IV.12 : Taux d'égale erreur dans l'ensemble d'évaluation et l'ensemble de test
Les courbes confirment les résultats obtenus.

IV.5.3 Taux de réduction de capacité de BDD

IV.5.3.1. Taux de réduction de capacité pour LDA 1 seul dimension :

Taille original BDD=112M octets et taille de BDD projetée = 8.54 M octets

$$\text{Taux} = \frac{\text{taille de BDD projetée}}{\text{taille originale BDD}} * 100. \quad (\text{IV .24})$$

Taux de réduction=100 – taux.

Taux=7.62%

Taux de réduction=100 – 7.62=92.38%.

IV.5.3.2 Taux de réduction de capacité pour PCA2D LDA2D :

Taille original BDD=112 Moctets et Taille de BDD projetée = 4.27 Moctets

$$\text{Taux} = \frac{\text{taille de BDD projetée}}{\text{taille originale BDD}} * 100.$$

Taux de réduction=100 – taux.

Taux=3.81 %

Taux de réduction=100 – 3.81=96.19%.

Nous remarquons une nette amélioration du taux de réduction pour PCA2D et LDA2D. En plus elle est plus rapide que LACP1D LDA1D.

IV .5 Conclusion

Le filtre que nous avons utilisés dans notre étude est le filtre de **Roberts** avec un seuil =0.03, ce seuil a été choisi car c'est le cas où les contours sont le mieux détectés.

Nous tentons ensuite de chercher le meilleur mesure de score par le filtre de **Roberts** entre différentes fonctions des mesures de score : Euclidienne norme L_1 , Euclidienne norme L_2 , Covariance, Mahalanobis et Corrélation.

Nous constatons que le résultat obtenu avec la distance **Euclidienne norme L_2** est le meilleur avec un **TEE** de l'ordre de **3.6695%** et le taux de réussite **TR=92.6408%** en utilisant seulement **60** paramètres de **ACP2D** et **30** paramètres de **ADL2DOC** ce résultats est obtenu dans l'ensemble d'évaluation.

Le système est bien stable vu que les paramètres obtenus pour la base de donnée évaluation sont assez proches de ceux obtenus pour la base de donnée test. La différence trouvés entre le **TEE=3.6695 %** et le **TEE_t= 5.2193%** (**TR=92.6408%**, **TR_t=90.0823%**) est dû au fait que la BDD test contient plus d'imposteurs à peu près trois fois.

D'après l'étude faite et toutes les expériences réalisées nous pouvons dire que l'**ACP2D** et l'**ADL2DOC** reste une méthode de discrimination très efficace d'une façon générale par exemple Taux de réduction de capacité pour **LDA une dimension est de 92.38%** contre **Taux de réduction de capacité pour PCA2D LDA2D =96.19%**.

Et le temps d'exécution pour **LDA une seule dimension 3.2s** le alors que temps d'exécution pour **PCA2D LDA2D 1.2s**.

Conclusion générale

Dans les applications de contrôle d'accès, la biométrie constitue une solution efficace, simple et surtout pas chère, qui assure de bonnes performances. Et parmi toutes les technologies biométriques qui existent, la reconnaissance de visages est l'une des technologies les plus utilisées et les plus adaptées.

Vu la naissante nécessité d'utiliser des applications de contrôle d'accès, la reconnaissance de visages a émergé comme un secteur actif de recherches, enjambant des disciplines telles que le traitement d'image, l'identification de modèle, et la vision d'ordinateur.

Le problème qui nous préoccupe contient deux classes, à savoir d'une part les clients et d'autres part les imposteurs. Un système d'authentification impitoyable et extrêmement strict indique un **TFA** (Taux de Fausse Acceptation) faible et un **TFR** (Taux de Faux Rejet) élevé. Par contre un système laxiste sera caractérisé par un **TFA** élevé et un **TFR** plutôt bas. Le juste milieu situe quelque part entre les deux, et si les taux d'erreurs sont égaux, il se trouvera au taux d'égale erreur ou **TEE**.

Tous ces taux d'erreurs ont été calculés dans deux ensembles d'abord dans un ensemble d'**évaluation**, qui va permettre de **fixer** plus ou moins le **TEE** en faisant varier les paramètres d'acceptation et de rejet du système. Ensuite dans un ensemble de **test** en utilisant les paramètres fixés précédemment. Ainsi, on peut **vérifier la robustesse du système d'authentification**.

Les systèmes d'authentification de visage utilisent habituellement la représentation de l'image du visage en niveaux de gris comme caractéristique d'entrée. Notre souci est l'amélioration des performances du système d'authentification, pour cela nous avons introduit après la conversion en niveaux de gris l'opération de découpage avec une image (**161*121**). Ensuite la transformation de **Hough** est réalisée après le filtrage et le seuillage de l'image découpée. Pour l'extraction des vecteurs de caractéristiques du visage (les valeurs propres) nous avons utilisé la méthode d'analyse en composantes principales bidimensionnelle (**ACP2D**). Un nouveau problème se présente ; **L'ACP2D** bonne pour la réduction de donnée (en effet avec les **100** premiers vecteurs propres on peut extraire plus de **90 %** de l'information de l'image d'entrée) ; mais elle ne tient pas compte de l'aspect de

discrimination alors que cette propriété est fort importante dans un système d'authentification efficace et cet inconvénient croît avec la grande variété d'expressions faciales des clients (L'éclairage, le changement de la coiffure, Le maquillage pour les femmes, le port de lunettes etc...). On ne peut pas donc atteindre des tâches telles que l'authentification des visages en utilisant une telle **ACP2D** pour le codage.

Une solution proposée pour résoudre ce problème est l'utilisation de l'analyse discriminante linéaire bidimensionnelle (**ALD2D**). Cette technique regroupe les images de même classe (personne) et sépare les autres. Pour augmenter les performances du système d'authentification, nous avons projeté les visages propres de L'**ACP2D** dans un espace **ALD2D** où la classification est intéressante.

Les résultats trouvés montrent que la transformation de **Hough** améliore les performances du système d'authentification. Pour la tâche d'authentification, une étape de mesure de similarité est nécessaire. Différentes fonctions de mesure de similarité ont été étudiées. Les résultats obtenus se résument dans ce qui suit. Le filtre que nous avons utilisé dans notre étude est le filtre de **Roberts** avec un seuil =0.03, ce seuil a été choisi car c'est le cas où les contours sont le mieux détectés.

Nous tentons ensuite de chercher le meilleur mesure de score entre différentes fonctions des mesures de score : Euclidienne norme L_1 , Euclidienne norme L_2 , Covariance, Mahalanobis et Corrélation.

Nous constatons que le résultat obtenu avec la distance **Euclidienne norme L_2** est le meilleur avec un **TEE** de l'ordre de **3.6695%** avec taux de réussite **TR=92.6408%** en utilisant seulement **60** paramètres de **ACP2D** et **30** paramètres de **ADL2DOC** cette résultat a l'ensemble de évaluation.

Le système est bien stable vu que les paramètres obtenus pour la Base de donnée évaluation sont assez proches de ceux obtenus pour la Base de donnée test. La différence trouvés entre le $TEE=3.6695\%$ et le $TEE_t=5.2193\%$ ($TR=92.6408\%$, $TR_t=90.0823\%$) est dû au fait que la BDD test contient plus d'imposteurs à peu près trois fois.

D'après l'étude faite et toutes les expériences réalisées nous pouvons dire que **l'ACP2D** et **l'ADL2DOC** reste une méthode de discrimination très efficace d'une façon générale par exemple **Taux de réduction pour LDA seul dimension de**

capacité Taux de réduction=92.38%. Taux de réduction de capacité pour PCA2D LDA2D =96.19%.

Et le temps d'exécution pour LDA un **seul dimension est 3.2s** et le temps d'exécution pour **PCA2D, LDA2D est 1.2s.**

PERSPECTIVES

Pour améliorer de plus les performances du système d'authentification nous proposons plusieurs types d'amélioration :

1. L'utilisation de la transformation de Hough pour la détection des cercles et des ellipses.
2. création d'un programme pour la recherche du seuil .
3. Proposer d'autres distances pour la mesure de similarité .
4. Utiliser la couleur de l'image comme caractéristique supplémentaire .
5. Utiliser d'autres filtres.
6. Intégrer un détecteur dans notre système de reconnaissance de visage.
7. La multimodalité et la fusion des données : en utilisant le profil des visages, la parole, les empreintes digitales, etc. ..
8. L'amélioration de la détection de l'image de visage

Bibliographie

[SOU08] : **SOUHILA GUERFI ABABSA**, Authentification d'individus par reconnaissance de caractéristiques biométriques liées aux visages 2D/3D, THÈSE pour obtenir le titre de Docteur de l'Université Evry Val d'Essonne
Spécialité: Sciences de l'Ingénieur université d'Evry val d'Essonne, 2008

[BED08] : **BEDRA MANSOURA**, **Mémoire de Fin d'Etudes** Pour l'obtention du diplôme D'Ingénieur d'état en électronique, Université Med Khider Biskra soutenue en Juin 2008.

[OUA_MEH09] : **OUAMANE Abdelmalik. MEHDAOUI Abdelghaffar**, Identification et Authentification des visages en biométrie, Mémoire de Fin d'Etudes Pour l'obtention du diplôme D'Ingénieur d'état en Electronique
Option : Communication, Université Mohamed Khider Biskra Faculté des sciences et de la Technologie Département de Génie Electrique Juin 2009

[MAR03] : **Marie-Claude - SVEF**, LA BIOMETRIE Documentation de Nicolas Six
email : six@benchmark.fr MC, jeudi 20 mars 2003

[CSI05] : **Cité des sciences et de l'industrie 2005 biométrie le corps identité**
www.cite-sciences.fr

[MOHO2]: **Mohamed Tayeb Laskri - Djallel Chefrour** , Who_Is: système d'identification des visages humains, Département d'Informatique - Université Badji Mokhtar . Article, 2002 Annaba B.P. 12 Annaba 23000 Algérie Email:
mtlaskri@wissal.dz

[KHT05] : **AL KHATIB Mohamed-Aban** ,Le rôle de l'anthropologie dans les techniques émergentes de reconnaissance faciale Université Claude Bernard - Lyon 1
Département de Biologie Humaine. Article, Jeudi 15 septembre ,2005

[SBBL05] : D. SAIGAA N. BENOUDJIT K. BENMAHAMED S. LELANDAIS
authentification d'individus par reconnaissance de visages, Département
d'Automatique Université Mohamed Khider Biskra (email:d_saigaa@yahoo.fr)
Algérie, Département d'électronique Université Elhadj Lakhdar Batna (e-mail :
benoudjit_nabil@yahoo.fr) Algérie, Département d'électronique Université Ferhat
Abbes Sétif Algérie, Laboratoire des systèmes complexes LSC université D'Evry
France, Juin 2005

[MOD] : Moad Benkiniouar, Mohamed Benmohamed ,Méthodes d'identification
et de reconnaissance de visages en temps réel basées sur AdaBoost

[BOU06] : BOUTARFA Abdelhalim, reconnaissance de formes 3d par approche
neuronale associant la transformée de hough en robotique mobile application a la
productique, **these**, Présentée au Département d'Electronique Faculté des Sciences de
l'Ingénieur, Université de Batna Pour l'obtention du diplôme de Doctorat Es-Sciences
en Electronique Industrielle Option : Contrôle, Soutenue le 2006

[RED06]: REDA JOURANI, Reconnaissance de visages, mémoire pour l'obtention
du Diplôme des Etudes Supérieures Approfondies, Université Mohammed V-Agdal ,
Novembre2006.

[MOR09]: NICOLAS MORIZET, Reconnaissance Biométrique par Fusion
Multimodale du Visage et de l'Iris, présentée pour obtenir le grade de Docteur, École
Doctorale d'Informatique, Télécommunications et Électronique de Paris, Mars 2009.

[MEG09]: MEGHALSI Mohamed Reda , OUAFEK Redouane , La détection des
visages dans une image, Pour l'obtention du diplôme d'ingénieur, Ecole nationale
Supérieure en Informatique (E.S.I) Oued-Semar, Alger, septembre 2009.

[ROU09]: C. ROUABHIA, J. KHELIL & H. TEBBIKH , Laboratoire
d'Automatique et Informatique de Guelma –LAIGU niversité 8 mai 45. BP
402,Article, Guelma c_rouabhia@yahoo.fr ,2009.

[ROU07]: Ch.Rouabhia, A/h.Boualleg et H.Tebbikh , Approche Bidimensionnelle Hybride Neuro-ACPD2D pour la Reconnaissance Automatique de Visages. Laboratoire d'Automatique et Informatique de Guelma -LAIG -Université de Guelma, Algérie ,c_rouabhia@yahoo.fr, bouallegah@yahoo.fr , tebbikh@yahoo.com ,2007.

ANNEXE A

A.1. Qu'est ce qu'une image ?

On peut trouver de nombreuses définitions du mot « image ».
En consultant par exemple le petit Robert une des définitions données, considère l'image comme : « la reproduction exacte ou la représentation analogique d'une être, ou d'une chose ».

On peut donc appeler image tout ce qui a une représentation d'une chose (autre qu'elle même). Si on nomme objet tout ce qui à une existence ou immatérielle, on définit alors :

A.1.1. Les images physiques visibles

Qui sont parfaitement matérielle et de nature volatile où permanente.

* les images volatiles : sont soient les images optiques (image données par les instruments d'optique, hologrammes,...), soit des images électro-optiques (écran d'un tube cathodique,...).

* les images permanents : sont les reproductions de toute sortes (clichés photographiques, dessins, peintures, documents imprimés, etc....)

[OUA_MEH 09]

A.1.2. Les images physiques non visibles

Ce sont des images de nature « immatérielle », Tel que : les cartes de population, ou de températures .en général ce sont les représentations de tous les paramètres physiques non directement visible. [OUA_MEH 09]

A.1.3. L'image numérique

Il est clair que les images manipulées par l'ordinateur sont numériques (série de bits).

L'image numérique est l'image dont la surface est divisée en éléments de taille fixe appelés pixels, ayant comme caractéristique le niveau de gris ou de couleur. La numérisation d'une image est la conversion de celle-ci en une image numérique représentée par une matrice bidimensionnelle de valeurs

numériques $f(x, y)$ qui sont les niveaux de gris des coordonnées réelles (x, y) .

[OUA_MEH 09]

A.2. Qu'est ce qu'un contour?

Par définition, un contour est une brusque variation de niveau de gris dans une image d'une amplitude a et avec une pente p . Un contour peut être défini comme une "marche d'escalier" si le contour est net, comme une "rampe" si le contour est plus flou ou comme un "toit" s'il s'agit d'une ligne sur un fond uniforme.

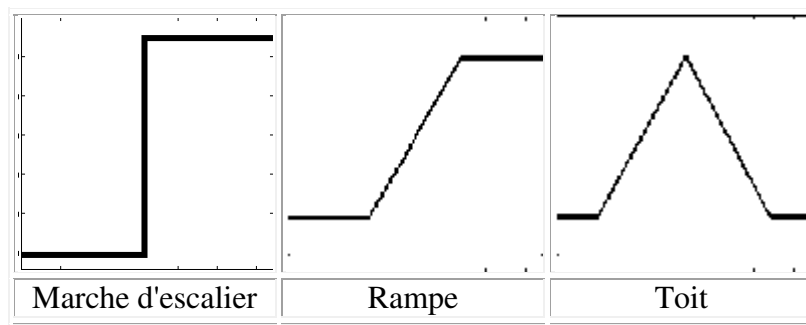


Figure. A.1 Un contour peut être défini comme une "marche d'escalier", "rampe" ou "toit"

La mise en évidence des contours peut se faire notamment par différentiation de l'image, grâce à des filtres représentant des opérateurs de différentiation tels les filtres détecteurs de Roberts, Sobel ou Prewitt, ou par des techniques morphologiques telles le gradient morphologique. [OUA_MEH 09]

ANNEXE B

II.4. La Transformée de Hough

La transformée de Hough est un outil de détection de courbes paramétriques dans l'image, elle a été proposée par Hough dans un brevet déposé en 1960. Inaperçue pendant plusieurs années, cette dernière a été vulgarisée par les travaux de Rosenfeld, Duda et Hart au début des années 70 et fait l'objet par la communauté scientifique depuis cette date à ce jour d'une particulière attention. Depuis les années 80, elle a quitté les laboratoires de recherche pour trouver des champs d'applications dans de nombreux domaines industriels tels que la vision par ordinateur et le traitement d'images. Elle est devenue une solution plus adaptée au problème de détection des lignes droites, cercles ou toute autre forme paramétrique dans l'image. [BOU06]

II.4.1. Principe de la transformée de Hough

Une droite est décrite dans le plan cartésien (x, y) par l'expression suivante :

$$F(y, x, a, b) = y - ax - b = 0 \quad (\mathbf{B.1})$$

Sachant que \mathbf{a} est la pente et \mathbf{b} l'ordonnée à l'origine des abscisses. Etant donné un ensemble de contours d'objets représentés par un ensemble de points discrets \mathbf{M}_i , nous cherchons à déterminer si un ou plusieurs sous-ensembles de points \mathbf{M}_i font partie d'une courbe dont les paramètres \mathbf{a} et \mathbf{b} restent à définir. Si nous cherchons à tester les \mathbf{n} points \mathbf{M}_i deux par deux, nous arriverons à un nombre exagéré d'itérations au moins supérieur à \mathbf{n}^2 .

Hough puis Rosenfeld ont proposé une méthode pour détecter les droites à l'aide des points du plan (x, y) . Son principe est de calculer pour chaque point \mathbf{M}_i de coordonnées (x_i, y_i) , du contour d'un objet, l'ensemble des paramètres \mathbf{a} qui vérifient l'équation $\mathbf{f}(y_i, x_i, \mathbf{a}, \mathbf{b}) = 0$ avec \mathbf{b} fixé.

Pour chaque point $\mathbf{M}_i(x_i, y_i)$, de l'image, il y a un ensemble de valeurs possibles pour les paramètres \mathbf{a} et \mathbf{b} . Cet ensemble forme une droite d'équation $\mathbf{b} = -\mathbf{a}x + y$ dans l'espace des paramètres (\mathbf{a}, \mathbf{b}) appelé espace de **Hough**. Deux points \mathbf{p}_i et \mathbf{p}_j de coordonnées (x_i, y_i) et (x_j, y_j) respectivement, appartenant à la même droite, forment des droites dans l'espace des paramètres (\mathbf{a}, \mathbf{b}) , qui se coupent au point \mathbf{N} de

coordonnées (a', b') . De cette façon tous les points qui appartiennent à la même droite forment des droites dans le plan des paramètres (ab) qui se coupent au même point. Ce concept est illustré dans les figures **B.3.a** et **B.3.b**.

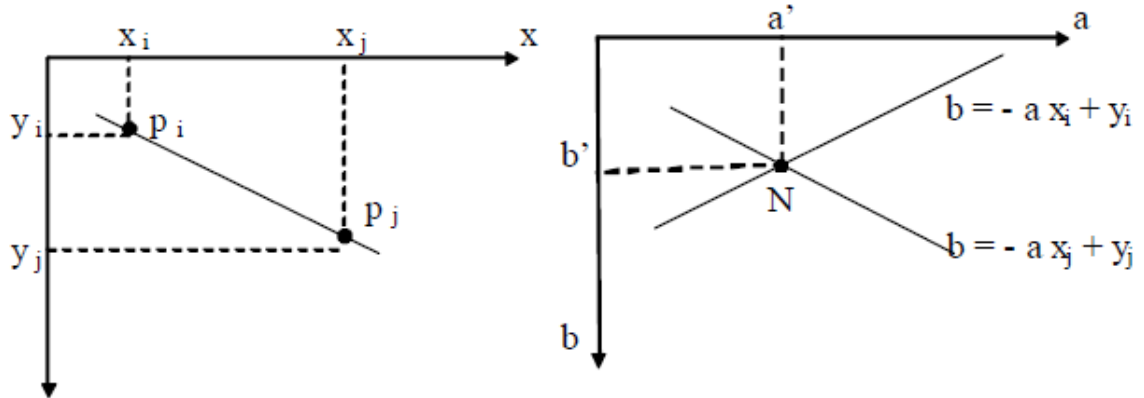


Figure B.1 Transformée de Hough

(a) Plan cartésien (xy)

(b) Plan des paramètres (ab)

Le traitement Hough consiste en une quantification du plan des paramètres en a cellules accumulatrices sur la figure B.2. Où (a_{min}, a_{max}) , et (b_{min}, b_{max}) sont les valeurs limites de l'intervalle de la pente a et de l'ordonnée à l'origine des abscisses

b.

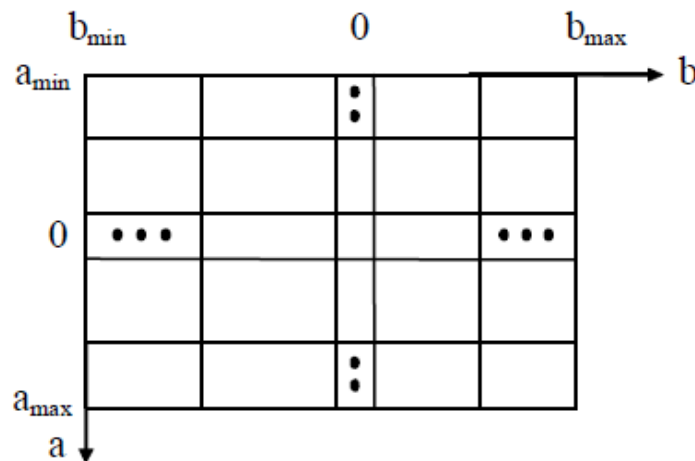


Figure B.2. Quantification du plan des paramètres (ab)

Chaque cellule de coordonnées (i, j) a une valeur accumulée $A(i, j)$ et correspond à la cellule de coordonnées (a_i, b_j) dans le plan des paramètres (a, b) . Initialement, ces cellules sont mises à zéro. Pour chaque point de l'image de coordonnées (x_k, y_k) on calcule pour chaque valeur de a quantifié a la valeur p a sur l'axe des a , son correspondant b en utilisant l'équation suivante : $b = -ax_k + y_k$. La

valeur résultante résultat \mathbf{b} est arrondie à la valeur la plus proche de \mathbf{b} quantifiées \mathbf{b}_q sur l'axe des \mathbf{b} . Si on obtient une valeur \mathbf{b}_q suite à \mathbf{a}_p choisie, on incrémente la valeur de la cellule correspondante:

$$A(p, q) = A(p, q) + 1 \quad (\text{B.2})$$

A la fin de cette procédure, la valeur \mathbf{n} de $\mathbf{A}(\mathbf{i}, \mathbf{j})$ dans une cellule (\mathbf{i}, \mathbf{j}) correspond à \mathbf{n} points dans le plan (\mathbf{x}, \mathbf{y}) qui vérifient l'équation $\mathbf{y} = \mathbf{a}_i \mathbf{x} + \mathbf{b}_j$, donc il existe \mathbf{n} points qui appartient à la droite de pente \mathbf{a}_i et de l'ordonnée à l'origine des abscisses \mathbf{b}_j .

L'inconvénient majeur de cette procédure réside dans son incapacité de détecter les droites verticales. Pour remédier à ce problème, un paramétrage polaire (ρ, θ) est plus satisfaisant. Ce paramétrage est illustré dans la figure .B.3.

Une droite est alors définie par l'équation suivante :

$$f(x, y, \rho, \theta) = \rho - x \cos \theta - y \sin \theta = 0 \quad (\text{B.3})$$

Avec ρ la distance perpendiculaire à la droite de l'origine du plan (\mathbf{x}, \mathbf{y}) et θ l'angle entre cette distance et l'axe de \mathbf{x} .

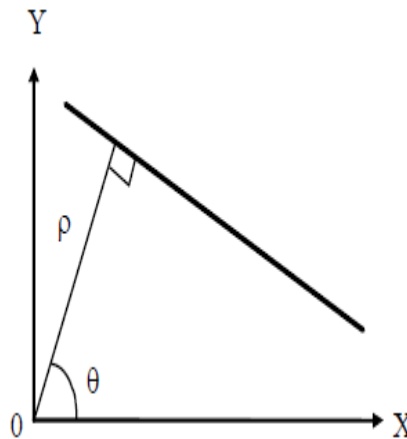


Figure B.3 Paramétrage polaire d'une droite

L'utilisation de cette représentation dans la construction du tableau accumulateur est identique à celle développée précédemment (représentation \mathbf{a} \mathbf{b} sur la (figure .B.3).

On précisera que le choix de quantification de l'espace des paramètres (ρ, θ) doit porter sur les trois objectifs essentiels suivants:

- Garantir une précision de détection aussi bonne que possible,

- Diminuer la mémoire nécessaire au stockage des accumulateurs,
- Accélérer les calculs.

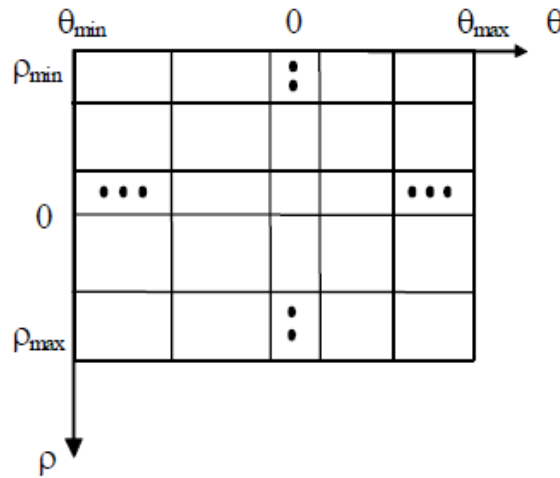


Figure B.4 Quantification du plan des paramètres (ρ θ)

Chaque point M_i de coordonnées (x_i, y_i) d'une droite se transforme dans le plan des paramètres (\mathbf{ab}) en une sinusoïde d'équation :

$$\rho = x_i \cos\theta + y_i \sin\theta \quad (\text{B.4})$$

Donc une droite sera représentée par un ensemble de sinusoïdes qui se coupent en un seul point de coordonnées polaires (ρ_0, θ_0) ρ , θ caractéristique de cette droite dans le plan des paramètres (se référer aux figures B. 5-a et B.5-b).

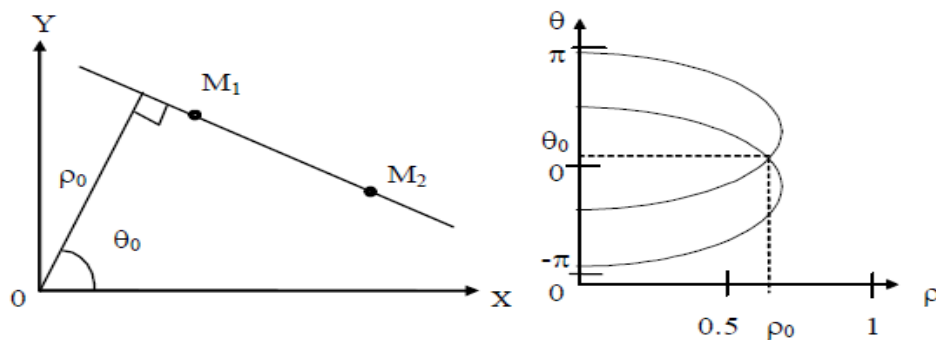


Figure B.5 Transformée de **Hough**

(a) Plan cartésien (\mathbf{xy})

(b) Plan des paramètres (\mathbf{ab})

ANNEXE C

C.1 Langage de programmation

Puisque nous nous intéressons au traitement des images, qui sont en fait des matrices de pixels. Nous avons choisi l'environnement de travail **Matlab7.7** (R2008b), parce que c'est un environnement basé sur les matrices et qui possède de bons algorithmes pour la manipulation de celle-ci (multiplication, factorisation....).

Il possède en outre des bibliothèques de manipulation d'images proposant des fonctionnalités intéressantes.

Enfin, c'est un logiciel gratuit et bénéficiant d'un large support dans la communauté scientifique.

C.2 Ressource

Les expérimentations et les tests ont été exécutés et réalisés sur deux ordinateurs différents :

- Un pentium ayant un processeur 2 G Core 2 duo, possédant une mémoire RAM de 2 Giga.
- Un pentium 4 pour un processeur de 3.6 Giga, une mémoire RAM de 512 Méga.