Towards unconstrained face recognition from one sample

Ngoc-Son Vu

UNIVERSITE DE GRENOBLE

INSTITUT POLYTECHNIQUE DE GRENOBLE

No. attribué par la bibliothèque

THESE

pour obtenir le grade de

DOCTEUR DE L'Université de Grenoble

délivré par l'Institut polytechnique de Grenoble

Spécialité : **Signal, Image, Parole, Télécoms**

préparée au laboratoire GIPSA-lab/DIS

dans le cadre de **l'École Doctorale *Électronique, Électrotechnique, Automatique, Traitement du Signal***

présentée et soutenue publiquement

par

Ngoc Son VU

le 19 novembre 2010

Titre : Contributions à la reconnaissance de visages à partir d'une seule image et dans un contexte non-contrôlé

Directrice de thèse : Mme. Alice Caplier

JURY :

M. Pierre-Yves Coulon	Président
Mme. Bernadette Dorizzi	Rapporteur
M. Christophe Rosenberger	Rapporteur
M. Jean-Marc Odobez	Examinateur
M. Christophe Blanc	Examinateur
Mme. Alice Caplier	Examinateur

Tra, vo cua anh,
con trai Hoàng Tít,
bo me va gia dinh than yeu.

Remerciements

Ce travail n'aurait jamais vu le jour sans l'aide précieuse d'Alice Caplier, qui m'a guidé tout au long de ces trois années de thèse. Merci Alice pour tes encouragements et tes conseils, ce fut un vrai plaisir de réaliser cette thèse avec toi.

Je tiens à remercier sincèrement Bernadette Dorizzi et Christophe Rosenberger qui ont accepté de juger ce travail. Leurs remarques et suggestions lors de la lecture de mon rapport m'ont permis d'apporter des améliorations à la qualité de ce dernier.

Merci également à Jean-Marc Odobez et Christophe Blanc pour l'intérêt qu'ils portent à ce travail en acceptant d'en être les examinateurs. La discussion avec Jean-Marc Odobez sur mon travail a été très enrichissant de façon ouverte et passionnante, et je l'en remercie sincèrement.

Un grand merci à Pierre-Yves Coulon qui a accepté de présider mon jury et aussi qui était très gentil de m'avoir parlé très ... très doucement dans les premiers jours où je suis venu au laboratoire ("d..i..s m..o..i s..i t..u a..s l..a c..l..é... p..o..u..r e..n..t..r..e..r?" mais je n'avais rien compris, oufs :(–)

Ma reconnaissance s'adresse également à Antoine Manzanera et Gérard Chollet qui m'ont initié à la recherche en me permettant de réaliser mon stage de Master dans leurs équipes.

Il m'est également impossible d'oublier mes collègues sympathiques au Gipsa-lab, je n'en cite ici que quelques un, Pierre Adam, Hannah Dee, Cécile Fiche et tous les autres avec qui j'ai eu le plaisir d'échanger des mots. Un grand merci à l'ensemble du personnel technique et administratif du laboratoire.

Cette thèse est financé par l'entreprise Vesalis, j'ai donc une pensée pour les personnes de l'entreprise, tout particulièrement Jean-Marc Robin et Christophe Blanc.

Je souhaite remercier tous mes proches qui ont su me soutenir durant cette thèse. Merci ma femme, Trà, pour son amour, son soutien et son encouragement lors de moments difficiles. Merci également Minh Hoàng Tít pour des moments agréables dans la vie (bien sur, quand tu n'es pas malade ;o)). Enfin, mes vifs remerciements s'adressent à mes parents et toute ma famille pour sa confiance et son soutien constant tout au long de mes études et de mon doctorat.

Table des matières

Table des figures

Liste des tableaux

Chapitre 1

Introduction

Identifier une personne à partir de son visage est une tâche aisée pour les humains. En est-il de même pour une machine ? Ceci définit la problématique de la reconnaissance automatique de visages [58, 22, 4], qui a engendré un grand nombre de travaux de recherche au cours des dernières années.

Le visage est une donnée biométrique [1]. Une donnée biométrique est une donnée qui permet l'identification d'une personne sur la base de ce qu'il est (caractéristiques physiologiques ou comportementales) [2] [3]. Les indices biométriques physiologiques sont des traits biologiques/chimiques innés, alors que les indices biométriques comportementaux sont associés à des comportements appris ou acquis. Le tableau 1.1 fournit une liste des indices biométriques couramment utilisés. Les données biométriques sont devenues des données incontournables pour le problème de l'identification sécurisée et de la vérification de personne. Les méthodes d'identification ou de vérification d'identité basées sur une donnée biométrique offrent les avantages suivants par rapport à des méthodes basées sur un mot de passe ou un code PIN. Premièrement, les données biométriques sont des données individuelles alors que les mots de passe peuvent être utilisés ou volés par quelqu'un d'autre que l'utilisateur autorisé. Deuxièmement, une donnée biométrique est très pratique car il n'y a rien à porter ou à mémoriser. Troisièmement, les technologies biométriques deviennent de plus en plus précises et ont un coût qui diminue constamment.

Parmi les éléments biométriques figurant dans le tableau 1.1, seules les données visage et parole appartiennent à la fois aux deux catégories physiologique et comportementale. Cela veut dire entre autre que la biométrie faciale nous permet d'exploiter de nombreuses informations relatives à une personne. Dans la vie quotidienne, le visage est probablement

1. National Institue of Standards and Technologies (NIST), Biometrics Web Site. http ://www.nist.gov/biometrics.
2. Biometric Catalog. http ://www.biomtricscatalog.org
3. Biometric Consortium. http ://www.biometrics.org.

Biométries physiologiques	ADN, visage, empreinte, forme de la main, iris, rétine, odeur, voix
Biométries comportementales	Démarche, visage, voix, écriture, signature

Tableau 1.1 – Les biométries couramment utilisées.

le trait biométrique le plus utilisé par les humains afin de reconnaitre les autres. Comme indiqué dans [47], le visage a de grands avantages par rapport aux autres biométries, parce qu'il est naturel, et facile à acquérir.

Dans un contexte de vidéosurveillance, la propriété la plus importante d'un système biométrique est sa capacité à pouvoir recueillir la donnée biométrique en contexte non-coopératif. Toutes les biométries du tableau 1.1, à l'exception du visage et de la démarche, ont besoin de la coopération du sujet. En plus des applications liées à l'identification et à la vérification de l'identité d'une personne, la reconnaissance de visages est également utile en interaction homme-machine, réalité virtuelle, indexation et multimédia.

Du fait de son vaste champ d'applications, la biométrie faciale est devenue un des thèmes de recherche les plus actifs dans le domaine de la vision par ordinateur, de la reconnaissance des formes, et de la compréhension d'images.

1.1 Panorama

Le problème de la reconnaissance de visages peut être formulé comme suit : étant données une ou plusieurs images d'un visage, la tâche est de trouver ou de vérifier l'identité d'une personne par comparaison de son visage à l'ensemble des images de visage stockées dans une base de données. Des informations supplémentaires telles que la race, l'âge, le sexe, ou la parole peuvent être utilisées pour réduire l'espace de recherche (ce qui permet d'améliorer les performances de reconnaissance).

En général, un système de biométrie faciale est constitué de deux modules : un module de segmentation de visages (détection ou localisation de visage), et un module de reconnaissance qui se déroule en trois étapes : normalisation ou prétraitement, extraction de caractéristiques faciales, classification (c.f. figure 1.1).

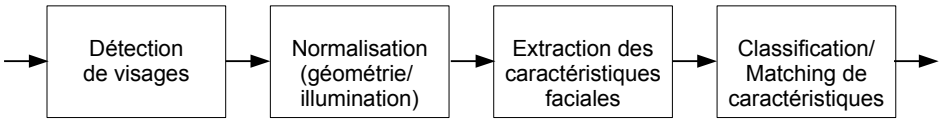


Figure 1.1 – Schéma général de reconnaissance de visages.

La détection des visages est la première étape importante de tous les systèmes à biométrie faciale. Étant donnée une image ou une séquence d'images, l'objectif de cette étape est de déterminer la présence ou non d'un visage dans l'image ainsi que sa localisation. Après la segmentation, le visage est normalisé. La normalisation implique généralement une normalisation géométrique des visages dans un but d'alignement et une normalisation d'éclairement dans un but de compensation des variations d'illumination. Les zones de visages normalisées sont ensuite utilisées pour l'extraction des caractéristiques faciales. Les caractéristiques extraites sont les informations utiles à la phase de reconnaissance et elles doivent être dans la mesure du possible discriminantes et robustes aux changements extérieurs, tels que la pose, l'expression, etc. Les caractéristiques faciales sont modélisées pour fournir la signature biométrique du visage qui est ensuite utilisée dans la phase de classification. Lors de cette dernière étape, on distingue deux tâches : l'identification et la vérification. En mode identification de visage, l'image à l'entrée du système est l'image d'un inconnu et le système doit rechercher l'identité de cet inconnu par comparaison de ses caractéristiques faciales à l'ensemble des caractéristiques faciales des visages de la base de données d'individus connus. En mode vérification, la personne à l'entrée du système déclare son identité et le rôle du système est de confirmer ou de rejeter l'identité revendiquée par comparaison de ses caractéristiques faciales uniquement avec celles de l'identité revendiquée.

1.2 Contexte : le projet BIORAFALE

Le travail de cette thèse s'inscrit dans le cadre du projet BIORAFALE [4] financé par OSEO qui vise à développer et tester en conditions réelles d'utilisation un démonstrateur d'aide efficace à la reconnaissance/vérification des individus interdits de stade. L'objectif est de proposer une aide efficace à la localisation des interdits de stade en exploitant les informations vidéo acquises à l'entrée des stades lors du passage des portiques d'entrée et de la phase de fouille. Le point clef du projet est donc le développement d'un système de reconnaissance en temps réel de visages en contexte de vidéosurveillance. Dans le cadre du projet, il y a deux grands challenges. Le premier challenge est que la reconnaissance de visage doit être faite dans un contexte d'acquisition non contrôlé et peu coopératif des images de visage. Le deuxième challenge est qu'a priori, nous ne disposons dans la base de données que d'une seule image frontale de la personne à reconnaître. En effet, le fichier national des interdits de stade est un fichier officiel dans lequel les individus sont enregistrés avec une vue de face et 2 vues de profil.

Le consortium associé à ce projet est constitué de partenaires industriels, publics et d'utilisateurs finaux dont le rôle est décrit rapidement ci-dessous.

4. Effectivement, cette thèse a commencé en décembre 2007 alors que le projet a débuté en décembre 2008 et pour une durée de 4 années.

• *La PME VESALIS* intervient dans le projet en tant que coordinateur et développeur pour certains algorithmes. Elle se charge de la définition des matériels et de la phase d'intégration (caméras, serveur,...), de l'optimisation des systèmes et des bases de données et du support technique.

• *La PME EFFIDENCE* intervient dans le projet en vue de participer au développement non pas d'un système de biométrie "automatique" mais d'un système de biométrie "intelligent et proactif" capable de raisonner et de s'adapter à son contexte d'utilisation. Pour permettre son exploitation dans des environnements non contrôlés, son savoir-faire est sollicité d'une part sur la modélisation des exigences de l'application (définition des connaissances que doit acquérir en ligne le système pour réaliser sa mission) et d'autre part, au niveau de la gestion intelligente de l'ensemble des caméras du site et des algorithmes de détection/identification.

• *Le GIPSA-LAB/DIS*, laboratoire de recherche associé au CNRS, intervient dans le projet afin de développer des recherches au niveau d'une part du conditionnement des données vidéo (estimation de la qualité des images, amélioration de la résolution des images de visage) et d'autre part, au niveau de la phase de reconnaissance d'individus proprement dite en s'appuyant sur des méthodes d'identification en conditions non contrôlées à base d'indices faciaux.

• *Le LASMEA*, laboratoire de recherche associé au CNRS, intervient dans le projet en vue du développement d'algorithmes de détection et de suivi de visages dans les séquences de vidéosurveillance. Des travaux récents ont été menés pour combiner des techniques de classification récentes (combinaison de *classifiers* par AdaBoost, SVM, RVM) dans un cadre de suivi probabiliste basé sur l'utilisation de filtres à particules.

• *L'EURECOM*, laboratoire de recherche privé, intervient dans le projet sur les aspects de reconnaissance de visage essentiellement : extraction de paramètres dynamiques en vidéo pour compléter et améliorer la reconnaissance basée sur l'apparence uniquement ; classification des visages selon divers critères : catégorie d'âge, présence d'une écharpe sur le visage, suivi d'un visage sur plusieurs caméras, etc. ; contribution à la définition et la mise en oeuvre d'un système multimodal.

• *L'INT* s'intéresse aux aspects juridiques et sociétaux du projet. Il se penche sur la problématique de la vidéosurveillance et aussi sur la protection des données personnelles tant au niveau de l'Union européenne qu'en France.

• *Le Clermont-foot* intervient dans le projet en tant qu'expert de vidéosurveillance de stade de foot et en mettant à disposition du projet ses installations pour l'acquisition de données vidéo réalistes et pour tout ce qui concerne la phase de test du système développé dans le cadre du projet.

L'ensemble de ces partenaires permet de couvrir l'ensemble des besoins décrits sur la

figure 1 depuis la conception des algorithmes jusqu'à leur intégration dans un démonstrateur final.

Le travail présenté dans le cadre de ce mémoire ne traite donc ni du problème de la segmentation des visages ni du problème de leur normalisation géométrique. On part du postulat que les régions des visages ont été correctement détectées et alignées. Notre travail se concentre donc sur l'étape de reconnaissance qui inclut les trois étapes de prétraitement des images, de définition et d'extraction des caractéristiques faciales et de classification. Dans la suite de ce travail, nous nous plaçons dans le cas le plus difficile pour lequel la reconnaissance de visage est faite dans un contexte d'acquisition des images non contrôlé et peu coopératif sachant que la base de référence ne contient qu'une seule image pour chaque individu à reconnaître.

Remarquons qu'à l'entrée du système, nous pourrons disposer d'une ou de plusieurs images du visage des personnes à reconnaître car les données d'entrée du système sont fournies par des caméras de vidéosurveillance. Cependant, dans le cadre de cette thèse, nous nous sommes limités (par manque de temps) au cas de la reconnaissance à partir d'une seule image d'entrée.

1.3 Problématique

La reconnaissance de visages pose de nombreux défis car les visages sont des objets déformables 3D et nous nous limitons dans ce travail à une reconnaissance à partir d'une image 2D de visage. A l'heure actuelle, il existe de nombreux algorithmes de reconnaissance de visage performants sous réserve que les conditions d'acquisition des visages soient contraintes, ce qui nécessite que les sujets soient coopératifs et veuillent se faire identifier. Mais les problèmes de reconnaissance de visages en environnements non-contrôlés ne sont pas encore résolus. Dans cette thèse, nous nous concentrerons sur des approches 2D de reconnaissance de visages en environnements non contraints (contexte de vidéosurveillance). De tels systèmes doivent pouvoir s'affranchir des problèmes suivants :

- Variations de pose
- Variations d'illumination
- Variations d'expression, d'âge
- Occultation partielle du visage
- Image de mauvaise qualité

Ces problèmes sont difficiles car les variations de l'apparence du visage d'une personne en conditions différentes sont souvent beaucoup plus importantes que les variations entre les images de visage de deux individus différents acquis dans les mêmes conditions (les variations intra-classes sont alors plus importantes que les variations inter-classes). Et parmi tous ces problèmes, les variations d'illumination et de pose sont les deux défis les

plus difficiles. En effet, les performances de reconnaissance de visages chutent de manière très importante en cas des conditions d'illumination et/ou de pose variables.

Pour résoudre le problème de ces variations intra-classes, il est communément admis que si un grand ensemble d'apprentissage comprenant l'ensemble des différentes images représentant ces variations pour chaque personne est disponible, on peut augmenter la robustesse du système de reconnaissance par la modélisation explicite de ces variations intra-classes pour chaque personne. Cependant, nous ne pouvons pas nous placer dans ce cas là.

Il est possible de tenir compte de ces variations à plusieurs niveaux du système : au cours du prétraitement, lors de l'extraction des caractéristiques ou lors de la classification. Cette thèse se concentre *directement* sur deux défis les plus difficiles, à savoir les variations de pose et d'illumination. Grace à l'utilisation d'un filtre inspiré du fonctionnement de la rétine, nous résolvons le problème de l'illumination lors de l'étape de prétraitement. Nous allons également présenter un nouveau descripteur facial qui est robuste aux variations de pose, d'illumination, d'expression et d'âge. Ensuite, par un algorithme qui se situe dans l'étape de classification, nous résolvons de manière satisfaisante le problème des variations de pose.

1.4 Principales contributions

L'extraction d'informations robustes et discriminantes est nécessaire dans beaucoup d'applications, pas seulement pour les systèmes de reconnaissance. La contribution la plus importante de cette thèse est la proposition d'un nouveau descripteur appelé POEM (Patterns of Oriented Egde Magnitudes) et destiné à représenter les structures locales d'une image. Ce descripteur est discriminant, robuste aux variations extérieures et *très rapide* à calculer. Les résultats présentés prouvent la robustesse et le caractère discriminant du descripteur proposé non seulement en biométrie faciale mais aussi de manière plus générale pour tout problème de mise en correspondance ou appariement d'images. En biométrie faciale, le descripteur POEM conduit à de très bonnes performances, comparables aux meilleures performances de l'état de l'art en contexte contrôlé ou non, et en même temps il *réduit de manière significative la complexité* en temps et en occupation mémoire. En ce qui concerne la mise en correspondance entre deux images, en comparaison avec le descripteur de référence SIFT, le descripteur POEM conduit à des améliorations de performances en cas de transformations d'images importantes, telles que le changement de point de vue, le changement d'éclairement, le bruit, et la compression.

L'efficacité du nouveau descripteur proposé s'explique d'une part par la quantité d'information sur les objets que le descripteur englobe et d'autre part par le fait que le descripteur POEM est construit en utilisant les auto-similarités sur une région locale.

Le descripteur POEM incorpore à la fois des informations sur la texture et sur la forme de l'objet. Du fait que le descripteur POEM prend en compte les relations entre les intensités des pixels à plusieurs niveaux (au niveau de la cellule qui est un voisinage très local et au niveau du bloc qui est un voisinage de taille plus étendue), le descripteur proposé est résistant aux variations extérieures.

La deuxième contribution de ce travail est le développement d'une nouvelle méthode pour le prétraitement des images de visage en cas de forte variation d'illumination. A partir de l'étude des filtres modélisant le comportement de la rétine humaine développés au laboratoire Gipsa-lab, nous avons développé une nouvelle version adaptée au problème du prétraitement des images de visage en cas d'illumination variable. La méthode proposée d'une part supprime les variations d'illumination et d'autre part renforce les contours tout en conservant le contraste global d'image. Cette méthode est également *très rapide* car sa complexité reste linaire. L'approche proposée permet d'améliorer significativement à la fois les performances des algorithmes de détection de visages et des algorithmes de reconnaissance de visages dans tous les cas qu'il y ait ou non des variations d'illumination.

La troisième contribution est le développement d'un algorithme de reconnaissance de visages en conditions de pose variables, centré sur une modélisation de la façon dont l'apparence du visage change lorsque le point de vue varie. Le modèle proposé repose sur la fusion de plusieurs descripteurs afin d'améliorer les performances finales. Nous examinons le cas le plus difficile pour lequel la reconnaissance de visage en vue non frontale est faite sachant que la base de référence ne contient que des vues de face des individus à reconnaître. Les résultats présentés prouvent les performances de la méthode proposée. Notre algorithme est également beaucoup *moins complexe* que les autres.

La quatrième contribution est d'avoir présenté un nouveau modèle de classificateur basé sur les modelés de Markov cachés (HMMs). En général, dans les systèmes existants basés sur les HMMs, pour chaque individu à reconnaître, un HMM distinct est utilisé pour modéliser les relations spatiales entre les caractéristiques du visage, puis les images de test sont comparées à chacun de ces modèles. De telles méthodes nécessitent plusieurs images de référence pour chaque individu. En revanche, notre approche n'a besoin que d'une seule image de référence pour chacun des individus à reconnaitre et seulement deux HMMs sont nécessaires pour le système complet. A l'inverse de toutes les autres approches basées HMMs pour lesquelles les modèles sont construits directement à partir des caractéristiques du visage, notre classificateur utilise les similitudes entre patches d'images. En particulier, l'utilisation, pour la modélisation des relations spatiales entre composantes faciales, d'une base de données contenant des individus différents donne à nos HMMs la capacité de modéliser différentes variations intra-personnelles, telles que la pose, l'expression,... L'approche proposée a une complexité beaucoup plus faible que celle des méthodes traditionnelles basées HMMs en reconnaissance de visages. Ces propriétés

permettent à notre classificateur d'atteindre des performances élevées, et en particulier d'être le premier classificateur basé HMMs ayant été testé sur une base d'images contenant plus d'un million d'individus. Le classificateur proposé est également très prometteur pour d'autres tâches, telles que l'estimation de la pose de la tête et la reconnaissance de visages dans un contexte de comparaison de vidéos.

1.5 Plan du rapport

Notre travail s'intéresse au problème général de la reconnaissance/vérification de visages en contexte de vidéosurveillance. Nous nous focalisons sur les trois étapes de prétraitement, d'extraction de caractéristiques faciales, et de classification avec comme souci le traitement des deux difficultés les plus importantes dans ce contexte de reconnaissance non contrôlé qui sont la robustesse en conditions d'illumination variables et la robustesse en conditions de pose variables. Pour chaque étape, une solution nouvelle est proposée et évaluée.

Dans le chapitre 2, nous évoquerons l'état-de-l'art en reconnaissance de visages. Nous n'allons pas décrire tous les algorithmes de reconnaissance de visages mais nous nous focaliserons sur les algorithmes les plus populaires et sur ceux les plus adaptés à notre contexte d'étude. La deuxième partie de ce chapitre sera consacrée à la description des deux méthodes d'extraction de caractéristiques faciales les plus connues alors que la troisième partie présentera les bases de données et les protocoles d'évaluation qui seront utilisés dans les parties expérimentales de cette thèse.

Le chapitre 3 commence par un bilan des méthodes existantes pour résoudre le problème des variations d'illumination. Puis, on décrit brièvement la rétine et ses propriétés. Ensuite nous détaillons la méthode proposée ainsi que les tests sur les bases Yale B, FERET et AR destinées à en valider les performances.

Dans le chapitre 4, via un état de l'art sur les représentations faciales, nous montrerons tout d'abord que la plupart des descripteurs existants ne sont pas adaptés pour la reconnaissance de visages en contexte de vidéo-surveillance. Ensuite, de nouvelles représentations faciales basées sur les indices POEM ainsi que les parties expérimentales sur les bases FERET et LFW destinées à en valider les performances seront décrites en détails.

Dans le chapitre 5, nous présenterons un algorithme robuste de reconnaissance de visages en conditions de pose variables, centré sur une modélisation de la façon dont l'apparence du visage change lorsque le point de vue varie. Le modèle proposé repose sur la fusion de plusieurs descripteurs afin d'améliorer les performances finales. La validation de ce modèle sur la base FERET pose sera faite dans la dernière partie de ce chapitre.

Le chapitre 6 va présenter un nouvel algorithme basé sur les HMMs pour modéliser les relations entre les composantes faciales. Les validations sur les bases FERET, AR et LFW de cette stratégie seront faites dans la dernière partie de ce chapitre.

Dans le chapitre 7, nous reviendrons sur l'efficacité du descripteur POEM en montrant ses bonnes performances pour le problème de mise en correspondance d'images. Pour ce faire, nous présenterons également une méthode d'estimation de l'orientation dominante des régions.

Avertissement au lecteur : cette thèse a été écrite pour moitié en français (les chapitres 2,3) et pour moitié en anglais (les chapitres 4–8). En effet, n'étant pas de langue maternelle française, la rédaction du français est très difficile pour moi. Il résulte de cette double écriture quelques éventuelles redondances sur quelques points. En effet, je me suis efforcé de rendre chacune des parties les plus auto-suffisantes possibles.

Chapitre 2

Etat de l'art en reconnaissance de visages

Le but de ce chapitre est de donner un panorama des méthodes les plus significatives en reconnaissance 2D de visages. Tout d'abord, une brève présentation des méthodes les plus populaires utilisées en reconnaissance faciale est proposée puis deux des méthodes les plus connues pour l'extraction des caractéristiques faciales, étape indispensable dans les systèmes de reconnaissance de visages, sont décrites et enfin les bases de données de visages et les protocoles d'évaluation qui ont été utilisés dans ce travail sont présentés.

2.1 Reconnaissance 2D de visages : état de l'art

De nombreuses méthodes de reconnaissance de visages ont été proposées au cours des 30 dernières années. La reconnaissance faciale automatique est un challenge tel qu'il a suscité de nombreuses recherches dans des disciplines différentes : psychologie, neurologie, mathématiques, physique, et informatique (reconnaissance des formes, réseaux de neurones, vision par ordinateur). C'est la raison pour laquelle la littérature sur la reconnaissance de visages est vaste et diversifiée.

La psychologie a montré que les facteurs contribuant à la reconnaissance de visages sont complexes. Un bilan avec 19 résultats très intéressants sur la capacité des hommes en reconnaissance de visages se trouve dans l'article de Sinha *et al.* [119]. Les changements d'expression faciale et de pose affectent la reconnaissance de visages inconnus alors que la reconnaissance de visages connus n'est pas affectée par ces facteurs. Ainsi, les hommes sont capables de reconnaître facilement des gens familiers à partir d'images de mauvaise qualité ou de résolution faible. La figure 2.1 illustre la capacité des hommes en reconnaissance de visages familiers à partir d'images avec des dégradations impor-

tantes, ce qui n'est pas faisable par les systèmes actuels de reconnaissance automatique de visages.

Figure 2.1 – Est-ce que vous arrivez de reconnaître ces personnes ? Les individus dans l'ordre de gauche à droite sont : Bill Clinton, Jaques Chirac, Prince Charles.

Les neurologistes étudient depuis longtemps le mécanisme par lequel le cerveau reconnaît les visages, et beaucoup croient que le cerveau perçoit les visages d'une manière spécifique et très différente des autres objets visuels. Par exemple, les études ont constaté qu'une rotation d'image faciale de 180° dégrade la reconnaissance beaucoup plus qu'une même rotation pour des objets quelconques (voir figure 2.2). Dans un travail innovant [75], Moscovitch *et al.* ont démontré que le cerveau des hommes traite les visages et les objets dans des aires séparées à savoir que les visages sont traités dans une aire spéciale. Cette approche a fait office de référence pendant la dernière décennie jusqu'au travail récent de Jiang *et al.* [54] qui montre que le traitement des visages et des objets ne reposent finalement pas sur des mécanismes différents.

Figure 2.2 – Quand les visages ne sont pas vus dans leur état naturel, la capacité du système visuel humain à les distinguer est dégradée.

D'excellentes synthèses sont présentées dans les articles de Zhao *et al.* [145] et de Tan *et al.* [121]. Dans cette section, on décrit les méthodes automatiques de reconnaissance de visages les plus représentatives et on met en évidence les méthodes adaptées à notre contexte de reconnaissance 2D de visages à partir d'une seule image de référence.

• Les systèmes de reconnaissance de visages sont très souvent classés à partir des conclusions d'études psychologiques sur la façon dont les hommes utilisent les caractéristiques faciales pour reconnaitre les autres. De ce point de vue, on distingue les trois catégories suivantes :

1. **Les méthodes de correspondance globales :** ces méthodes utilisent la région entière du visage comme entrée du système de reconnaissance. L'une des méthodes la plus largement utilisée pour la représentation du visage dans son ensemble est la représentation à partir de *l'image de visages propres* ou *eigenfaces* [123] basée sur une analyse en composantes principales (ACP).
2. **Les méthodes de correspondance locales :** typiquement, ces méthodes extraient tout d'abord des caractéristiques locales, puis utilisent leurs statistiques locales (la géométrie et/ou l'apparence) comme donnée d'entrée du classificateur.
3. **Les méthodes hybrides :** ces méthodes combinent les deux types de caractéristiques locales et globales.

Dans la suite, on classe les méthodes de reconnaissance de visages de ce point de vue et on adopte les termes de méthodes globales et de méthodes locales pour désigner les méthodes de correspondance globales et les méthodes de correspondance locales respectivement. Il convient aussi de souligner la signification et la différence entre les termes globale et locale, notablement pour des représentations globales et locales. Elles se différentient par la manière dont elles ont été calculées. Une représentation est locale si elle prend en compte les relations locales d'image (voir section 2.2). Dans le cas contraire, la représentation est globale.

• Il existe une autre façon de catégoriser les systèmes de reconnaissance de visages en deux classes : méthodes basées sur l'apparence et méthodes basées modèle. Les méthodes basées sur l'apparence tentent de caractériser l'apparence du visage en concaténant les pixels de visage, alors que celles basées modèle créent des modèles spécifiques et identiques en utilisant les caractéristiques choisies sur les visages.

• De notre point de vue, les méthodes automatiques de reconnaissance de visages peuvent être également classées en deux autres catégories : les algorithmes *Généraux (General)* et les algorithmes *Spécifiques (Specific)*. Les "algorithmes généraux" sont s'efforcent de des algorithmes qui sont proposés pour la reconnaissance de visages en général et qui traiter toutes les variations d'images (par example variations de pose, d'illumination,etc.) de la même manière alors que les "algorithmes spécifiques" contiennent des stratégies spécifiques dédiées au traitement d'un type particulier de variations.

2.1.1 Méthodes globales

Le principe de ces méthodes est de représenter une image faciale par un seul vecteur de grande dimension en concaténant les niveaux de gris de tous les pixels de visage.

Cette représentation, appelée description basée sur *l'apparence globale*, a deux avantages. Premièrement, elle conserve implicitement toutes les informations de texture et de forme utiles pour différentier des visages. Deuxièmement, elle peut tenir compte des aspects d'organisation structurelle globaux du visage.

Pour traiter le problème des données de grande dimension, des techniques de réduction de la dimensionnalité peuvent être utilisées. L'une des techniques les plus courantes pour la reconnaissance de visages est la description par *visages propres* [123], qui est basée sur l'analyse en composantes principales (ACP). A partir de ce principe de base, plusieurs variantes ont été développées au cours des dernières années, telles que l'ACP à noyaux et l'ACP de dimension deux [138], les *Fisherfaces* basés sur une analyse discriminante linéaire (ADL) [8, 78], le modèle probabiliste Bayésien [84], les machines à vecteurs de support (SVMs) [100], l'évolution poursuite (Evolution poursuit) [72], ou le visage laplacien (laplacianfaces) [44]. Toutes ces approches conduisent en général à des performances supérieures à celles de la méthode originale basée sur le *visage propre*. Cependant, elles ne fonctionnent plus dès lors qu'une seule image de référence est disponible pour chaque personne à reconnaitre. Dans le reste de ce paragraphe, parmi les méthodes citées ci-dessus, on présentera plus en détails la méthode à base du *visage propre* alors que les autres ne seront évoquées que brièvement en insistant sur les raisons pour lesquelles elles ne sont pas adaptées à la reconnaissance faciale à partir d'une seule image de référence.

2.1.1.1 Analyse en composantes principales (ACP) et visages propres

L'ACP est une méthode mathématique d'analyse de données qui consiste à rechercher les directions de l'espace qui représentent le mieux les corrélations entre des variables aléatoires. C'est une transformation orthogonale linéaire qui transforme les données dans un nouveau système de coordonnées tel que la variance selon la première coordonnée soit la plus grande (appelée première composante principale), la variance selon la deuxième coordonnée soit la deuxième plus grande, et ainsi de suite. L'ACP est souvent utilisée pour réduire la dimension des données en ne gardant que les caractéristiques principales qui contribuent le plus à la variance globale et en ignorant les composantes de petites variances. Les composantes conservées contiennent les informations les plus importantes et les composantes ignorées contiennent les aspects les moins informatifs des données. Soient $\mathbf{x}_i \in R^d, i = 1..n$ n vecteurs aléatoires et $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n\}$. On appelle sa moyenne $\mu = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i$ et sa matrice de covariance $\Sigma_{\mathbf{x}} = \sum_{i=1}^{n}(\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T$.
L'ACP cherche $\mathbf{u}_1 \in R^d, \mathbf{u}_1\mathbf{u}_1^T = 1$ pour que la variance de la projection de $\mathbf{X}$ selon cette direction soit la plus grande. De la même façon, on cherche $\mathbf{u}_k \in R^d, \mathbf{u}_k\mathbf{u}_k^T = 1$ qui maximise la variance de la projection de $\mathbf{X}$ selon cette direction.

Alors, on maximise :

$$\gamma = var(\mathbf{u}_i\mathbf{X}) = \mathbf{u}_i^T\Sigma_\mathbf{x}\mathbf{u}_i = \frac{\mathbf{u}_i^T\Sigma_\mathbf{x}\mathbf{u}_i}{\mathbf{u}_i^T\mathbf{u}_i},$$

Par l'inégalité de Cauchy-Schwartz :

$$\gamma \leq \frac{\|\mathbf{u}_i\|\|\Sigma_\mathbf{x}\mathbf{u}_i\|}{\|\mathbf{u}_i\|^2} = \frac{\|\Sigma_\mathbf{X}\mathbf{u}_i\|}{\|\mathbf{u}_i\|},$$

A partir de cette formule, on voit bien que γ est maximisé si et seulement si $\Sigma_\mathbf{x}\mathbf{u}_i$ et $\mathbf{u}_i$ sont colinéaires, soit $\exists\alpha \in R : \Sigma_\mathbf{x}\mathbf{u}_i = \alpha\mathbf{u}_i$ et $\gamma \leq |\alpha|$. Ceci montre que $\mathbf{u}_i$ est effectivement un vecteur propre de $\Sigma_\mathbf{x}$.

Soit $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_k$, les d valeur propres de $\Sigma_\mathbf{x}$ et $\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_d$ les vecteurs propres orthogonaux correspondant. Comme $\Sigma_\mathbf{x}$ est une matrice symétrique, ses vecteurs propres forment une base d'un espace vectoriel. Alors l'écart des données originales projetées selon la direction $\mathbf{u}_1 = \mathbf{v}_1$ est le plus grand ($\gamma = \alpha = \lambda_1$), l'écart des données originales projetées selon la direction $\mathbf{u}_2 = \mathbf{v}_2$ est le deuxième plus grand ($\gamma = \alpha = \lambda_2$) et ainsi de suite. Une nouvelle base $\{\mathbf{u}_1 = \mathbf{v}_1, \mathbf{u}_2 = \mathbf{v}_2, ..., \mathbf{u}_d = \mathbf{v}_d\}$ est ainsi obtenue et elle satisfait le but de l'ACP.

On peut alors représenter les données originales dans le nouvel espace vectoriel. Effectivement, dans le nouvel espace vectoriel, le vecteur $\mathbf{x}_i$ (en réalité $\mathbf{x}_i - \mu$) est représenté par : $\mathbf{x}_i - \mu = \sum_{j=1}^{d} p_{ij}\mathbf{u}_j$ ou autrement dit, le vecteur $\{p_{i1}, p_{i2}, ..., p_{id}\}$ de taille d contient les coefficients décrivant le vecteur $\mathbf{x}_i$. Pour réduire la dimension des données, on va se limiter dans cet espace aux k premiers axes ($\mathbf{u}_i$) correspondant aux valeurs propres les plus grandes (en pratique, il y a de nombreuses valeurs propres qui sont petites ou très proches de zéro) : $\mathbf{x}_i - \mu \simeq \sum_{j=1}^{k} p_{ij}\mathbf{u}_j$, avec $k \ll d$. On a finalement un espace vectoriel d'ACP $\mathbf{U_{acp}} = \{\mathbf{u}_1, \mathbf{u}_2, ..., \mathbf{u}_k\}^\mathbf{T}$ dans lequel les données $\mathbf{x}_i$ de taille d vont être représentées par les vecteurs $\mathbf{p}_i$ de la taille $k, k \ll d$ et ces vecteurs sont calculés par :

$$\mathbf{p}_i = \mathbf{U_{acp}}(\mathbf{x}_i - \mu), \tag{2.1.1}$$

En utilisant l'ACP, en 1991, Turk et Pentland ont développé la méthode très célèbre de reconnaissance de visage à partir des "visages propres" [123]. En considérant une image faciale de la base d'apprentissage comme un vecteur aléatoire, les visages propres sont les axes principaux obtenus en effectuant l'ACP sur les vecteurs associés. Par rapport aux notations précédentes, les vecteurs $\mathbf{x}_i$ sont les images originales de la base de données, les bases du nouvel espace vectoriel d'ACP $\mathbf{U_{acp}} = \{\mathbf{u}_1, \mathbf{u}_2, ...\}^\mathbf{T}$ sont les visages propres, et les vecteurs $\mathbf{p}_i$ représentent l'image i dans l'espace des visages propres ou encore appelé sous-espace de visages. L'avantage de cette représentation est la réduction de la dimension et de la sensibilité au bruit car le bruit est en général associé aux axes de

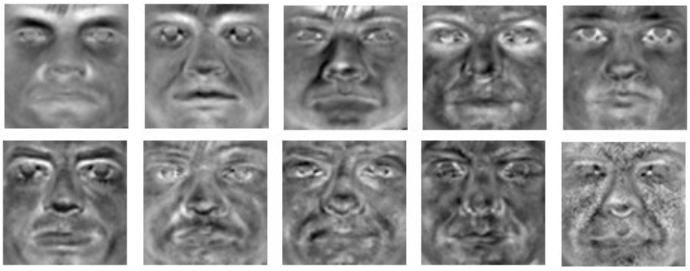


Figure 2.3 – Les 10 visages propres de la base YaleB : dans l'ordre de gauche à droite, de haut en bas, les visages propres sont de moins en moins importants.

variation qui ont été négligés. La figure 2.3 montre les 10 visages propres obtenus en utilisant l'ACP sur la base YaleB (voir section 2.3).

De cette façon, toutes les images de la base de données de référence ainsi que chaque nouvelle image de test sont représentées par un vecteur de poids caractéristiques du visage considéré. L'identification de l'image de test est faite en recherchant l'image de la base de données dont le vecteur de poids est le plus proche de celui de l'image de test. Autrement dit, la méthode de classification consiste à rechercher le vecteur le plus proche. Pour ce faire, des métriques de distance entre vecteurs sont utilisées. Dans leur travail original [123], Turk et Pentland ont utilisé la distance euclidienne pour mesurer la similarité entre deux visages projetés dans le sous-espace de visage. Plus tard, d'autres métriques de distance différentes et des variantes de l'ACP ont été proposées.

Métriques de distances

Dans [98], Perlibakas a testé 14 métriques de distance et a constaté que les performances de la reconnaissance de visages basée sur l'ACP varient beaucoup en fonction des métriques utilisées. La measure de similarité la plus efficace est la distance de Mahalanobis. Mais la mesure de similarité la plus largement utilisée pour la méthode visage propre est la distance cosinus, du fait qu'elle est rapide à calculer et qu'elle donne des performances très proches de celles de la distance de Mahalanobis.

$$d_{maha}(\mathbf{p}_i, \mathbf{p}_j) = -\frac{1}{\|\mathbf{p}_i\|\|\mathbf{p}_j\|} \sum_{r=1}^{k} \frac{1}{\sqrt{\lambda_r}} p_{ir} p_{jr},$$

$$d_{cos}(\mathbf{p}_i, \mathbf{p}_j) = -\frac{\mathbf{p}_i^{\mathbf{T}} \mathbf{p}_j}{\|\mathbf{p}_i\|\|\mathbf{p}_j\|}$$

2.1.1.2 Autres algorithmes

En utilisant une mesure de probabilité de similarité, Moghaddam et Pentland ont étendu l'algorithme standard de "visages propres" à une approche bayésienne [84]. Le nouveau modèle formule le problème de reconnaissance de visages en un problème de classification binaire. Pour ce faire, ils ont proposé d'estimer deux distributions de probabilité, une sur les variations intra-personnelles et l'autre sur les variations inter-personnelles, dans des espaces de visages propres par des lois gaussiennes. A partir d'une image de test et d'une image de référence, ils calculent la différence et ils utilisent les distributions estimées pour décider à quelle classe cette différence appartient : si elle appartient à la classe "intra", l'identité de l'image de test est la même que celle de l'image de référence, et réciproquement. Comme les deux distributions sont estimées dans deux espaces de visages propres, cette méthode est également appelée méthode des doubles visages propres (*double eigenfaces*). Cette méthode a obtenu une des meilleures performances lors de la campagne d'évaluation des systèmes de biométrie faciale utilisant la base FERET (voir section 2.3.1) de 1997. Dans le cas où une seule image pour chaque individu est disponible, la distribution des variations intra-personnelles ne peut être estimée, cette méthode ne fonctionne donc plus.

Une autre méthode très connue est celle basée sur l'ADL (Analyse discriminante linéaire). L'objectif de la plupart des algorithmes basés sur l'ADL [8, 78] est de trouver les directions de projection les plus discriminantes dans l'espace propre, en maximisant le ratio entre les variations inter-personnelles et les variations intra-personnelles. Comme les variations intra-personnelles peuvent être petites (notablement quand il n'y a pas beaucoup d'images par individu), ce ratio est difficile à maximiser puisque il est déjà grand. Ce problème est encore appelé problème *Small Sample Size*. Pour l'éviter, on peut utiliser tout d'abord l'ACP et ensuite l'ADL, et cette méthode est appelée *Fisherfaces*. Voilà pourquoi les méthodes basées sur l'ADL ne fonctionnent bien que lorsque beaucoup d'images par personne sont disponibles dans la base d'apprentissage. En revanche, quand il n'y a pas beaucoup d'images par personne, les méthodes basées sur l'ADL marchent moins bien que celles basées sur l'ACP [8].

2.1.2 Méthodes locales

Les méthodes locales peuvent être classées en deux catégories, les méthodes basées sur les points d'intérêt et celles basées sur l'apparence du visage. Dans le premier cas, on détecte tout d'abord les points d'intérêt et ensuite on extrait des caractéristiques localisées sur ces points d'intérêt. Dans le second cas, on divise le visage en petites régions (ou patchs) sur lesquelles les caractéristiques locales sont extraites directement. En comparaison avec les approches globales, les méthodes locales présentent certains avantages. Tout d'abord, les approches locales peuvent fournir des informations supplémentaires

basées sur les parties locales. De plus, pour chaque type de caractéristiques locales, on peut choisir le classificateur le plus adapté. Les avantages des caractéristiques locales seront décrits en détails dans la section 2.2.

Malgré ces avantages, l'intégration d'informations de structure plus globale est nécessaire. En général, il y a deux façons de faire pour atteindre cet objectif. Premièrement, les informations globales sont intégrées dans les algorithmes en utilisant des structures de données, telles qu'un graphe où chaque noeud représente une caractéristique locale alors qu'une arrête entre deux noeuds représente la relation spatiale entre eux. La reconnaissance de visage apparaît comme un problème d'appariement de deux graphes. Deuxièmement, les algorithmes peuvent utiliser des techniques de fusion de scores : des classificateurs séparés sont utilisés sur chaque caractéristique locale pour calculer une similarité et ensuite les similarités obtenues sont combinées afin d'obtenir un score global pour la décision finale.

2.1.2.1 Méthodes locales basées sur les caractéristiques d'intérêt

Les méthodes les plus anciennes en reconnaissance de visages appartiennent à cette catégorie [58, 76, 20, 108, 64, 134]. Elles s'appuient toutes [58, 20] sur l'extraction de caractéristiques géométriques spécifiques telles que la largeur de la tête, les distances entre les yeux,etc. Ces données sont ensuite utilisées par des classificateurs afin de reconnaitre des individus[1]. Ces méthodes présentent les deux inconvénients suivants : (1) les caractéristiques géométriques sont difficiles à extraire dans certains cas puisque la tâche de détection précise de points caractéristiques n'est pas facile, en particulier dans les cas où des occultations ou des variations (pose, expression) de visages sont présentes, et (2) les caractéristiques géométriques seules ne sont pas suffisantes pour représenter entièrement un visage, et d'autres informations utiles telles que les valeurs des niveaux de gris de l'image sont complètement écartées.

Ces deux limites ont engendré deux directions de recherche. La première se concentre sur les performances des détecteurs de points caractéristiques du visage. Dans [20], Brunelli et Poggio ont proposé d'utiliser un ensemble d'apprentissage pour détecter la position de l'oeil dans une image. Ils ont tout d'abord calculé pour chaque point des coefficients de corrélation entre l'image de test et les images de l'ensemble d'apprentissage et ensuite ils ont cherché les valeurs maximales. Rowley *et al.* [108] ont utilisé plusieurs détecteurs de traits spécifiques correspondant à chaque partie du visage, telles que les yeux, le nez, la bouche, etc. Lanitis *et al.* [64] ont proposé de construire des modèle statistiques de la forme du visage. Malgré toutes ces recherches, il n'existe pas encore de détecteur de points caractéristiques qui soit suffisamment fiable et précis.

Dans la deuxième direction, les méthodes se concentrent sur des représentations plus

1. Ces méthodes sont également appelées méthodes basées modèle (model-based approaches)

élaborées des informations portées par les points caractéristiques du visage, plutôt que simplement sur des caractéristiques géométriques. Manjunath *et al.* [76] ont proposé des algorithmes pour détecter et représenter des caractéristiques faciales à partir d'ondelettes de Gabor. Pour chaque point détecté, deux types d'information sont stockées : sa position et ses caractéristiques (les caractéristiques sont extraites en utilisant le filtre Gabor sur le point considéré). Pour modéliser la relation entre les points caractéristiques, un graphe topologique est construit pour chaque visage.

Basée sur cette idée, Lades *et al.* [62] ont proposé d'utiliser un graphe topologique déformable au lieu d'un graphe topologique fixe comme dans [76] afin de proposer un modèle de représentation du visage appelé *Dynamic Link Architecture* (DLA). Cette approche permet de faire varier le graphe en échelle et en position en fonction des variations d'apparence du visage considéré. Le graphe est une grille rectangulaire localisée sur l'image (voir figure 2.4) où les noeuds sont étiquetés avec les réponses des filtres de Gabor dans plusieurs orientations et plusieurs fréquences spatiales appelées jets. Les bords sont étiquetés par des distances, où chaque bord relie deux noeuds sur le graphe. La comparaison entre deux graphes de visage est réalisée en déformant et en mettant en correspondance le graphe représentatif de l'image de test avec chacun des graphes représentatif des images de référence.

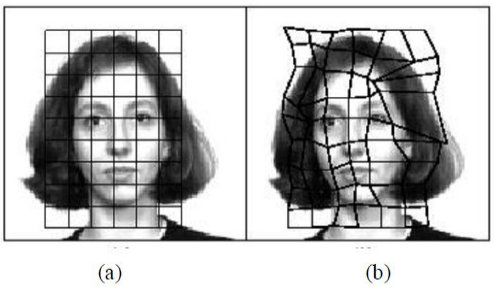


Figure 2.4 – Exemple de grille d'appariement. (a) grille de référence, (b) grille correspondante.

Plus tard, Wiskott *et al.* [134] ont étendu l'utilisation de l'DLA à une méthode très connue appelée Elastic Bunch Graph Matching (EBGM), où les noeuds des graphes sont situés sur un certain nombre de points sélectionnés du visage (voir figure 2.5). De manière similaire à la méthode de [76], Wiskott *et al.* ont utilisé les ondelettes de Gabor pour extraire les caractéristiques des points détectés car les filtres de Gabor sont robustes aux changements d'illumination, aux distorsions et aux variations d'échelle (voir les détails sur les filtres de Gabor dans la section 2.2). De ce fait, l'EBGM fut l'un des algorithmes les plus performants lors de la compétition de FERET en 1996. Le succès de ces méthodes a aussi été une motivation pour certains travaux plus récents [33, 36].

Pour conclure, de nombreuses méthodes basées sur l'extraction de points caractéris-

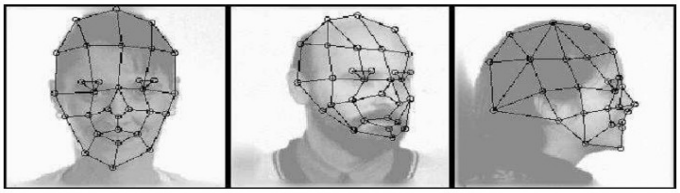


Figure 2.5 – EBGM.

tiques ont été proposées et ces méthodes peuvent être utiles et efficaces pour la reconnaissance de visages dans le cas où une seule image de référence est disponible. Cependant, leurs performances dépendent beaucoup de l'efficacité des algorithmes de localisation des points caractéristiques du visage. Et en pratique, la tâche de détection précise des points caractéristiques[2] n'est pas facile et n'a pas été résolue de manière fiable, en particulier dans les cas où la forme ou l'apparence d'une image du visage peuvent beaucoup varier.

2.1.2.2 Les méthodes locales basées sur l'apparence du visage

De manière générale, les méthodes de reconnaissance de visages basées sur l'apparence locale du visage comportent quatre étapes : le découpage en régions de la zone du visage, l'extraction des caractéristiques, la sélection des caractéristiques et la classification. Bien que ce paragraphe se concentre sur l'extraction des caractéristiques (et ne concerne que les deux premières étapes), on présentera quand même rapidement le principe des autres étapes pour que le discours soit complet et cohérent.

Étape 1 : Découpage en régions. Les deux facteurs qui définissent une région locale sont sa forme et sa taille. La forme peut être rectangulaire, elliptique, etc. (c.f. figure 2.6) mais ce qui est le plus largement utilisé est le découpage rectangulaire. Les fenêtres peuvent être superposées ou non. La taille de la région a une influence directe sur le nombre de caractéristiques et la robustesse de la méthode.

Étape 2 : Extraction des caractéristiques locales. Une fois que les régions locales ont été définies, il s'agit de choisir la meilleure manière de représenter les informations de chaque région. Cette étape est critique pour les performances du système de reconnaissance. Les caractéristiques couramment utilisées sont les valeurs de gris, les coefficients de Gabor [20, 134, 76], les ondelettes de Harr [126], les transformées de Fourier, les caractéristiques basées sur les indices LBP (Local Binary Pattern) [2] ou SIFT (Scale Invariant Feature Transform) [74]. On discutera en détails des méthodes d'extraction de caractéristiques faciales dans le paragraphe 2.2.

Étape 3 : Sélection des caractéristiques. Il est possible de calculer de nombreuses

2. Les détecteurs proposés récemment comme DoG de SIFT ou Harris-Laplace (voir chapitre 7) ne localisent pas les points caractéristiques du visage, telles que les yeux, la bouche,etc.

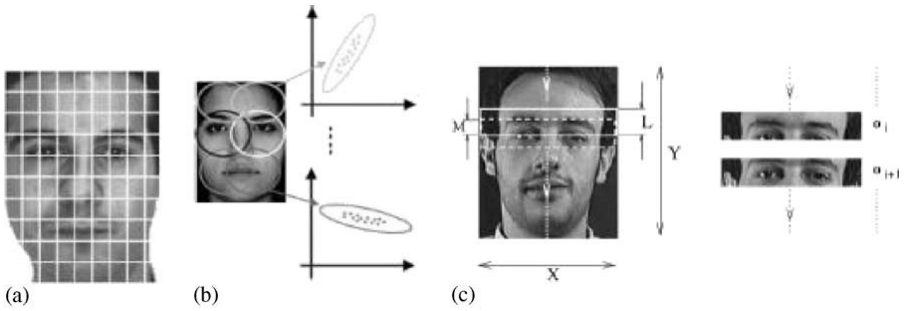


Figure 2.6 – Formes locales typiques des régions ou patchs d'images utilisés par les méthodes basées sur l'apparence locale.

caractéristiques faciales a priori. De ce fait, une étape de sélection des caractéristiques les plus pertinentes peut s'avérer nécessaire pour des questions de rapidité de traitement. L'ACP [123] est une méthode couramment utilisée pour sélectionner des caractéristiques en garantissant une perte minimum d'informations ; l'ADL [8, 78] peut être utilisée pour sélectionner les caractéristiques les plus discriminantes, d'autres techniques comme l'Adaboost [126] sont également possibles pour cette tâche.

Étape 4 : Classification. La dernière étape est bien entendu l'identification de visage. La stratégie la plus courante est de réaliser la fusion par vote à la majorité ou par somme pondérée de l'ensemble des décisions prises par chaque classificateur agissant sur une caractéristique faciale donnée.

Notez que les quatre étapes ci-dessus ne sont pas obligatoires pour toutes les méthodes. En particulier, la troisième étape de sélection des caractéristiques, peut être éliminée ou combinée avec d'autres étapes dans certains cas spécifiques.

Martinez a présenté une approche probabiliste locale pour la reconnaissance de visages qui sont occultés partiellement et avec des variations d'expression dans le cas où une seule image de référence est disponible [77]. Des expériences sur un ensemble de 2600 images montrent que l'approche probabiliste locale ne réduit pas le taux de reconnaissance même lorsque $\frac{1}{6}$ du visage est occulté. Cependant, les complexités de calcul et de stockage ainsi que la procédure de génération des échantillons virtuels sont très compliquées (dans cette méthode *spécifique*, 6615 échantillons virtuels par individu sont générés), en particulier si on a une base de référence avec de nombreux visages.

Comme expliqué précédemment, de manière générale, les méthodes basées sur l'ADL ne sont pas adaptées à la reconnaissance de visage à partir d'une seule image de référence. Chen *et al.* [24] ont proposé une méthode pour rendre l'ADL applicable dans ce cas là. Chaque image de visage est tout d'abord divisée en petites régions non-superposées de même dimension (figure 2.6) afin d'obtenir un ensemble de sous-images pour chaque in-

dividu. Puisque plusieurs échantillons par individu sont disponibles, l'ADL est applicable sur ces nouveaux ensembles. Cette méthode a été testée sur un sous-ensemble de la base FERET contenant 200 personnes avec une seule image de référence et un taux de reconnaissance de 86.5% est obtenu. Cependant, les sous-images générées sont très corrélées et ne devraient pas être considérées comme des échantillons d'apprentissage indépendants, par exemple les variations créées ne sont souvent pas suffisamment importantes pour couvrir celles observées dans la réalité.

La variation de pose est l'un des problèmes les plus difficiles en reconnaissance automatique de visages, en particulier dans le cas où une seule image de référence est disponible. Pour traiter ce problème, Kanade et Yamada [59] ont proposé une méthode probabiliste qui est similaire à celle de Moghaddam et Pentland [84]. Cette méthode *spécifique* sera décrite dans le chapitre 5.

Les méthodes mentionnées ci-dessus ne considèrent pas explicitement les relations existants entre les caractéristiques locales. Il est concevable que l'utilisation de cette information soit bénéfique au système de reconnaissance. Une solution possible est de construire un modèle flexible de géométrie sur les caractéristiques locales comme cela se fait dans la méthode d'EBGM. Motivés par cela, Heisele *et al.* [46] ont proposé une méthode de reconnaissance qui est basée sur les composantes faciales et qui utilise les caractéristiques d'intensité. Malheureusement, cette méthode a besoin de plusieurs images d'apprentissage par individu, prises sous des poses et des éclairages différents, et donc n'est pas adaptée à notre contexte. Une autre façon intéressante d'intégrer l'information globale est d'utiliser le modèle de Markov caché (HMM). Cette approche sera décrite en détails dans le chapitre 6.

2.1.3 Méthodes hybrides

Les méthodes hybrides (ou méthodes de fusion) sont des approches utilisant à la fois des caractéristiques globales et des caractéristiques locales. Les facteurs clés qui influent les performances des méthodes de fusion comprennent le choix des caractéristiques pour la combinaison et la manière de les combiner de telle sorte que leurs avantages soient préservés et que leurs inconvénients soient évités. Ces problèmes sont similaires ceux à des systèmes de classificateur multiple ou *ensemble learning* dans le domaine de l'apprentissage automatique. Malheureusement, même dans ces domaines, ces problèmes ne sont pas encore résolus.

Les caractéristiques locales et les caractéristiques globales ont des propriétés très différentes et peuvent offrir des informations complémentaires utiles à la tâche de classification. Nous notons aussi que d'un certain point de vue, les méthodes locales peuvent être considérées comme des méthodes hybrides car des informations globales sont généralement prises en compte. Dans la méthode probabiliste locale [77] de nouveaux

Variations	Caractéristiques locales	Caractéristiques globales
Petites variations	Pas sensible	Sensible
Grandes variations	Sensible	Très sensible
Illuminations	Pas sensible	Sensible
Expressions	Pas sensible	Sensible
Pose	Sensible	Très sensible
Bruit	Très sensible	Sensible
Occultations	Pas sensible	Très sensible

Tableau 2.1 – Comparaison des propriétés des caractéristiques locales et des caractéristiques globales.

échantillons d'apprentissage sont d'abord produits pour chaque personne par méthode globale, puis une méthode locale est utilisée pour la reconnaissance.

2.2 Extraction de caractéristiques

Dans toutes les méthodes de reconnaissance faciale, le point le plus délicat concerne la définition et l'extraction des caractéristiques faciales les plus pertinentes, à savoir les caractéristiques qui représentent le mieux les informations portées par un visage. Le choix de caractéristiques locales présente plusieurs avantages par rapport à des caractéristiques globales. C'est pour cette raison que les systèmes les plus récents s'appuient sur des caractéristiques faciales locales. Ici on présentera deux des caractéristiques locales les plus performantes dans le contexte de reconnaissance de visages, à savoir les ondelettes de Gabor et les indices Local Binary Patterns. Pour chaque type de caractéristique, il y a plusieurs façons de les utiliser. Nous présentons dans la suite seulement le principe de base. Il convient aussi de noter qu'il existe des variantes de ces caractéristiques originales et qu'il existe également des méthodes qui les combinent, elles seront présentées dans le chapitre 4.

2.2.1 Ondelettes de Gabor

Le filtre de Gabor, défini par Dennis Gabor, est largement utilisé en traitement d'images car les ondelettes de Gabor présentent deux propriétés intéressantes : la localisation fréquentielle et la sélectivité en orientation. Les représentations en fréquence et en orientation du filtre Gabor s'apparentent à celles du système visuel humain [51]. Les articles [94, 115] (le premier est dans *Nature*) indiquent que la représentation par ondelettes de Gabor des images faciales est robuste aux changements causés par des

variations d'éclairement ou par des modifications d'expressions faciales.

Les ondelettes en dimension deux de Gabor ont été introduites dans le domaine de la recherche biométrique par Daugman [31] pour la reconnaissance d'iris. Lades *et al.* [62] ont utilisé les premiers les ondelettes de Gabor en reconnaissance faciale en utilisant la *Dynamic Link Architecture*.

Un noyau de filtre de Gabor est le produit d'une onde complexe sinusoïdale avec une enveloppe gaussienne. Une ondelette de Gabor généralement utilisée dans la reconnaissance faciale est définie comme suit :

$$\psi_{u,v}(z) = \frac{\|k_{u,v}\|^2}{\sigma^2} e^{\frac{\|k_{u,v}\|^2 \|z\|^2}{2\sigma^2}} [e^{ik_{u,v}z} - e^{-\frac{\sigma^2}{2}}] \tag{2.2.1}$$

où $z = (x, y)$ est le point de coordonnées (x, y) dans le plan image. Les paramètres u et v définissent l'orientation et la fréquence des noyaux de Gabor.$\|.\|$ est l'opérateur norme, et σ l'écart-type de l'enveloppe gaussienne.

La représentation en ondelettes de Gabor d'une image résulte du produit de convolution de l'image avec une famille de noyaux de Gabor de fréquence et d'orientation différentes comme définis par l'équation 2.2.1. La convolution de l'image I et d'un noyau de Gabor $\psi_{u,v}z$ est définie par :

$$G_{u,v}(z) = I(z) * \psi_{u,v}(z) \tag{2.2.2}$$

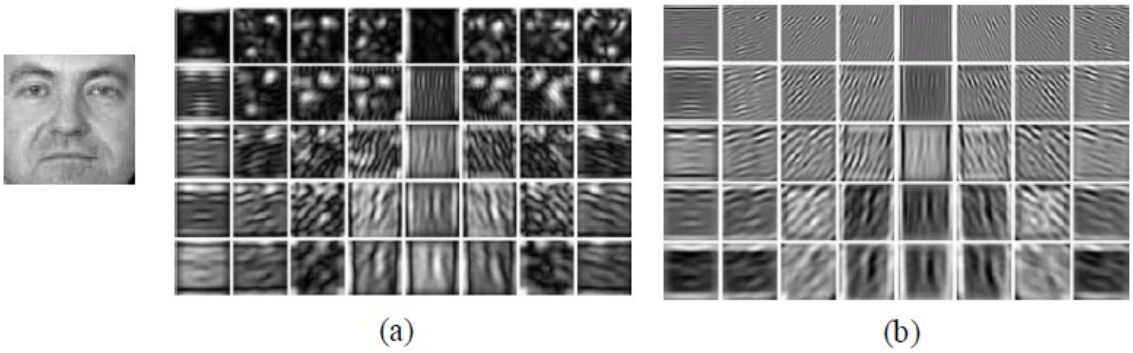


Figure 2.7 – Exemple de représentation faciale en ondelettes de Gabor : les réponses en l'amplitude (a) et en phase (b) d'un visage avec 40 noyaux de Gabor (5 échelles, 8 orientations).

L'intérêt d'utiliser les filtres de Gabor pour extraire des caractéristiques faciales est qu'ils permettent de capturer les informations de visage dans des orientations et des résolutions différentes. De plus, ils sont robustes aux changements d'illumination, aux distorsions et aux variations d'échelle. En effet, la convolution d'une image avec une

banque de 40 noyaux de Gabor (5 échelles et 8 orientations) conduit à 40 cartes d'amplitude et 40 cartes de phase qui sont de même taille que l'image originale, comme illustrée sur la figure 2.7. Par conséquent, si on ne considère que la réponse en amplitude, chaque pixel est décrit par un vecteur de dimension 40. Ce vecteur de dimension 40 est également appelé "Jet" et il a été utilisé largement dans les systèmes les plus anciens, tels que le DLA et EGBM (voir section 2.1.2.1). Notons que ce sont des méthodes basées sur des points caractéristiques qui doivent être détectés très précisément. Dans les méthodes plus récentes telles que [73], les auteurs ont utilisé directement une image de caractéristiques contenant les 40 cartes d'amplitude comme entrée de reconnaissance. Dans [79], Mellakh utilise les réponses en phase comme indice facial. Bien entendu, des méthodes de réduction de dimension ont été utilisées, telles que le sous-échantillonnage ou l'ACP avec ses variantes.

Plusieurs métriques ont été testées pour les caractéristiques basées sur les filtres de Gabor et celle qui est la plus souvent utilisée est la distance cosinus.

2.2.2 Local Binary Pattern

L'opérateur LBP a été proposé initialement par Ojala *et al.* [93] dans le but de caractériser la texture d'une image. Le calcul de la valeur LBP consiste pour chaque pixel à seuiller ses huit voisins directs avec un seuil dont la valeur est le niveau de gris du pixel courant. Tous les voisins prendront alors une valeur 1 si leur valeur est supérieure ou égale au pixel courant et 0 si leur valeur est inférieure (c.f. figure 2.8). Le code LBP du pixel courant est alors produit en concaténant ces 8 valeurs pour former un code binaire. On obtient donc, comme pour une image à niveaux de gris, une image des valeurs LBP contenant des pixels dont l'intensité se situe entre 0 et 255.

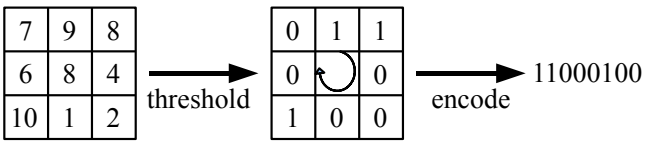
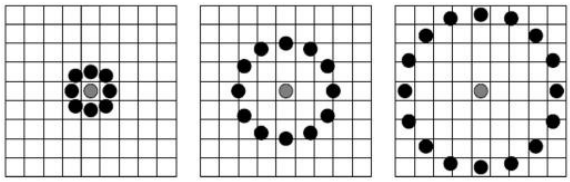
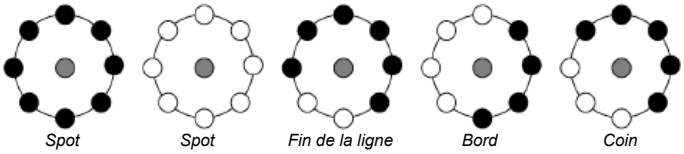


Figure 2.8 – Opérateur LBP.

Le LBP a été étendu ultérieurement en utilisant des voisinages de taille différente. Dans ce cas, un cercle de rayon R autour du pixel central est considéré. Les valeurs des P points échantillonnés sur le bord de ce cercle sont prises et comparées avec la valeur du pixel central. Pour obtenir les valeurs des P points échantillonnés dans le voisinage pour tout rayon R, une interpolation est nécessaire. On adopte la notation (P, R) pour définir le voisinage de P points de rayon R d'un pixel. La figure 2.9(a) illustre trois voisinages pour des valeurs de R et P différentes.



(a)



(b)

Figure 2.9 – (a) :Trois voisinages pour des R et P différents ; (b) : Textures particulières détectées par LBP^{u2}

Soient g_c le niveau de gris du pixel central, $g_p(p = 1..P)$ les niveaux de gris de ses voisins, l'indice LBP du pixel courant est calculé comme :

$$LBP_{P,R}(x_c, y_c) = \sum_{p=1}^{P} s(g_p - g_c)2^{p-1}, \tag{2.2.3}$$

où

$$s(x) = \begin{cases} 1 & if \quad x \geq 0, \\ 0 & if \quad x < 0, \end{cases} \tag{2.2.4}$$

où (x_c, y_c) sont les coordonnées du pixel courant, $LBP_{P,R}$ est le code LBP pour le rayon R et le nombre de voisins P. L'opérateur LBP obtenu avec $P = 8$ et $R = 1$ ($LBP_{8,1}$) est très proche de l'opérateur LBP d'origine. La principale différence est que les pixels doivent d'abord être interpolés pour obtenir les valeurs des points sur le cercle (voisinage circulaire au lieu de rectangulaire).

Une autre extension à l'opérateur d'origine est le LBP uniforme. Un code LBP est uniforme s'il contient au plus deux transitions de bits de 0 à 1 ou vice-versa lorsque la chaîne binaire est considérée circulaire. Par exemple, 00000000, 00011110 et 10000011 sont des codes uniformes. L'utilisation d'un code LBP uniforme, noté LBP^{u2} a deux avantages. Le premier est le gain en mémoire et en temps calcul. Le deuxième est que LBP^{u2} permet de détecter uniquement les textures locales importantes, comme les spots, les fins de ligne, les bords et les coins (c.f. figure 2.9(b) pour des exemples de ces textures particulières). En effet, Ojala *et al.* ont montré que les LBPs uniformes contiennent plus de 90% de l'information d'une image.

La propriété importante du code LBP est que ce code est invariant aux changements uniformes globaux d'illumination parce que le LBP d'un pixel ne dépend que des différences entre son niveau de gris et celui de ses voisins.

LBP pour la reconnaisance de visages

Une fois le code LBP calculé pour tous les pixels de l'image, on calcule l'histogramme de cette image LBP pour former un vecteur de caractéristiques représentant l'image faciale. En réalité, afin d'incorporer plus informations spatiales au vecteur représentant le visage, on divise tout d'abord l'image codée par l'opérateur LBP en petites régions et l'histogramme est construit pour chaque région. Finalement, on concatène tous les histogrammes des régions afin de former un grand histogramme représentant l'image des caractéristiques faciales (voir figure 2.10). L'efficacité du code LBP comme indice facial s'explique par le fait que le LBP permet de caractériser les détails fins d'un visage. Quand seules les LBPs uniformes sont utilisés, toutes les codes LBPs non-uniformes sont étiquetés avec une étiquette unique, alors que chacun des codes uniformes est regroupé dans un histogramme unique. Par exemple, quand $P = 8$, nous avons 58 codes uniformes mais l'histogramme est de dimension 59. De même manière $P = 6$ produit un histogramme de dimension 33.

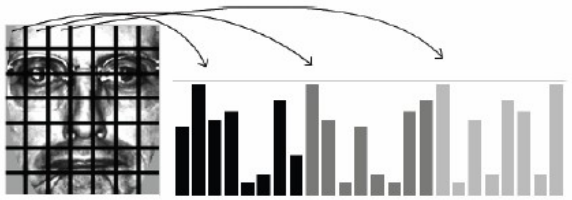


Figure 2.10 – Représentation d'un visage par les histogrammes du code LBP.

Etant donnés deux histograms de LBP H^1, H^2 de deux visages, l'étape suivante est d'utiliser une métrique pour calculer la similarité entre ces deux histogrammes. En testant les trois métriques χ^2, *Histogram intersection* et *Log-likelihood statistic*, Ahonen *et al.* [2] ont observé que la première métrique permet d'obtenir les meilleurs résultats : $\chi_2(H^1, H^2) = \sum_i \frac{(H_i^1 - H_i^2)^2}{H_i^1 + H_i^2}$.

2.3 Bases de données et protocoles d'évaluation associés

De nombreuses bases de visages existent pour les tâches différentes de reconnaissance de visages et le choix de la base de données à utiliser afin d'évaluer des performances spécifiques doit être considéré avec soin. Les bases de données utilisées doivent contenir

des images avec les multiples sources de variations possibles telles que l'éclairage, la pose, l'expression, etc.

Compte tenu de ces contraintes, nous avons tout d'abord choisi deux grandes bases de visages classiques : la base FERET et la base AR qui sont publiquement disponibles et qui contiennent des milliers d'images acquises sous des conditions variables d'illumination, de pose, d'expression, d'âge. Le choix est également motivé par le fait que la plupart des méthodes de reconnaissance utilisent ces bases pour évaluer leurs performances. Utiliser nous aussi ces mêmes bases nous permet donc de positionner nos résultats par rapport aux résultats de l'état de l'art. La base *Extended Yale B* est également utilisée. Bien que cette base ne soit pas très grande en comparaison avec les deux autres, c'est probablement la base la plus difficile à traiter et elle fait référence dans le domaine pour évaluer la résistance des systèmes de biométrie faciale en cas d'illumination variable. Finalement, on a choisi la base LFW (*Labeled Faces in the Wild*) qui a été créée très récemment. Cette base est considérée comme une base d'images réelles ou naturelles car ses images ont été collectées directement à partir des journaux sur Internet en utilisant un détecteur automatique de visage. Cette base est plus difficile car elle comporte de très nombreuses sources de variations intra-personnelles.

Il est nécessaire de noter que des bases comme FRGC (Face Recognition Grand Challenge) [3] et BANCA (Biometric access control for networked and e-commerce applications) [4] sont également intéressantes et utilisées dans la communauté scientifique. Cependant, par manque du temps, nous n'allons pas évaluer nos algorithms sur ces bases. Il également convient de noter que la base de données "réels" pour le projet Biorafale n'est pas encore prise, nous ne pouvons pas donc évaluer les algorithms sur telle base.

Les protocoles d'évaluation sur chacune de ces bases sont différents. Alors que les trois premières bases sont souvent utilisées pour évaluer les performances d'identification, la dernière base est seulement utilisée pour la vérification.

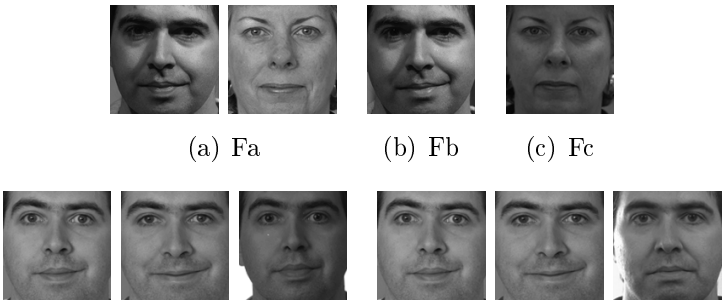
En ce qui concerne l'alignement du visage, dans la majorité des algorithmes, les images faciales (à l'exception des images sur la base LFW) sont détectées et normalisées manuellement en trois temps :

1. Une rotation est effectuée pour que la ligne joignant les deux yeux soit horizontale.
2. Une réduction proportionnelle à la distance entre les centres des deux yeux est appliquée pour que la distance entre les centres des yeux soit fixe.
3. Les dimensions de l'image du visage sont calculées à partir de la distance entre les centres des deux yeux :

$$\text{hauteur}_{visage} = \eta_1 * \text{dis}_{yeux} \tag{2.3.1}$$

$$\text{largeur}_{visage} = \eta_2 * \text{dis}_{yeux} \tag{2.3.2}$$

3. http ://www.frvt.org/FRGC/
4. http ://www.ee.surrey.ac.uk/CVSSP/banca/



(a) Fa (b) Fb (c) Fc

(d) Dup1 (e) Dup2

Figure 2.11 – Exemples d'images de face de la base FERET utilisées dans nos expérimentations.

2.3.1 Base FERET (Face Recognition Technology)

La base FERET a été créée et mise à disposition afin d'évaluer les progrès en reconnaissance automatique de visages. Cette base est administrée par le "National Institute of Standards and Technology" (NIST) aux États-Unis. C'est probablement la base de données la plus connue et la plus utilisée par la communauté de reconnaissance de visages. Elle contient un grand nombre de sujets dont le visage a été acquis dans des conditions différentes d'expression, de pose, d'éclairage et d'âge. A cette base est associé un protocole standard qui définit la stratégie d'évaluation en donnant une norme sur les ensembles d'apprentissage et de test. Par ailleurs, il y a également un sous-ensemble de la base FERET qui contient des images de visages de profil, appelé sous-ensemble FERET pose. La base FERET contient au total 14126 images de 1199 personnes différentes.

Les images de visage de face sont regroupées en 5 catégories : Fa, Fb, Fc, Dup1 et Dup2 (voir des exemples sur la figure 2.11). Les images de Fb ont été acquises le même jour que celles de Fa avec la même caméra, les mêmes conditions d'illumination mais avec des expressions différentes. Les images de Fc ont été acquises le même jour que celles de Fa mais avec des caméras différentes et des éclairages différents. Les images de Dup1 ont été prises au plus tard trois ans après celles de Fa. Les images de Dup2 ont été prises plus de 18 mois après celles de Fa. L'ensemble *gallery* (ou ensemble de référence) est un ensemble d'images d'individus supposés connus. Une image d'un visage inconnu présentée à l'algorithme de reconnaissance est désignée sous le terme d'image requête. Dans le protocole standard, l'algorithme compare la requête à chacune des images de la galerie et étiquette la requête par l'image de référence la plus similaire. Un ensemble d'apprentissage distinct est également disponible, il peut être utilisé pour entrainer des classificateurs. Dans ce protocole, les 1196 photos de Fa sont les images de référence, les 1195 photos de Fb, les 194 photos de Fc, les 722 photos de Dup1 et les 234 photos de Dup2 sont utilisées comme image requête. Dans ces tests, il y a une seule image de

(a) De face (b) $-60°$ (c) $-45°$ (d) $-25°$ (e) $-15°$ (f) $+15°$ (g) $+25°$ (h) $+45°$ (i) $+60°$

Figure 2.12 – Exemples d'images de pose différente de la base FERET.

référence par personne.

L'ensemble FERET pose contient des images de 200 personnes sous 9 poses différentes. La figure 2.12 montre les variations typiques de pose d'un des sujets dans la base FERET.

Les images de visages de face de la base FERET ont été mises à disposition avec les cordonnées des points de caractéristiques de visages, tels que les yeux, la bouche, le nez, etc. Grâce à cela, les chercheurs peuvent couper et aligner les vignettes de visages à leur guise. Afin de pouvoir faire des comparaisons pertinentes entre nos méthodes et les méthodes existantes, les images de face utilisées dans nos expérimentations sont alignées manuellement grâce aux cordonnées disponibles.

Dans les tests sur la base FERET (à l'exception des images de pose), les vignettes d'images sont obtenues en utilisant la procédure d'alignement décrite précédente, avec $\eta_1 = \eta_2 = 2.2$. Comme distance entre les deux yeux dis_{yeux}, nous choisissons à 44 pixels, ce qui conduit à des vignettes de visages de dimension 96×96 pixels [5] (voir figure 2.11). En effet, les méthodes existantes utilisent souvent des vignettes de visages avec une hauteur/largeur comprise entre 88 (comme dans [141]) et 128 pixels (comme dans [90]). Dans [79] Mellakh a montré que les performances de l'algorithme de reconnaissance considéré se stabilisent à partir d'une dis_{yeux} supérieure à 45 pixels.

En revanche, pour les visages de profil, les coordonnées des points caractéristiques ne sont pas disponibles. Il nous a fallu les détecter par nous même. Cependant, comme nous voulons évaluer la robustesse de notre algorithme aux mauvais alignements, ces points ne sont pas détectés parfaitement (au lieu des centres des yeux, nous avons détecté un

5. Il convient de noter que certains algorithmes nécessitent un alignement plus compliqué. Dans [146], Zou *et al.* ont besoin des coordonnées de plusieurs points caractéristiques (centres des deux yeux, de la bouche, du nez, des sourcils, etc.) ; ils ont également coupé le visage en une zone plus grande (η_1, η_2 sont plus grands) et montré que cela peut améliorer les performances de reconnaissance.

point quelconque dans la région de la pupille) et l'alignement est fait de manière semi-automatique. Nous le discuterons en plus détail dans le chapitre 5.

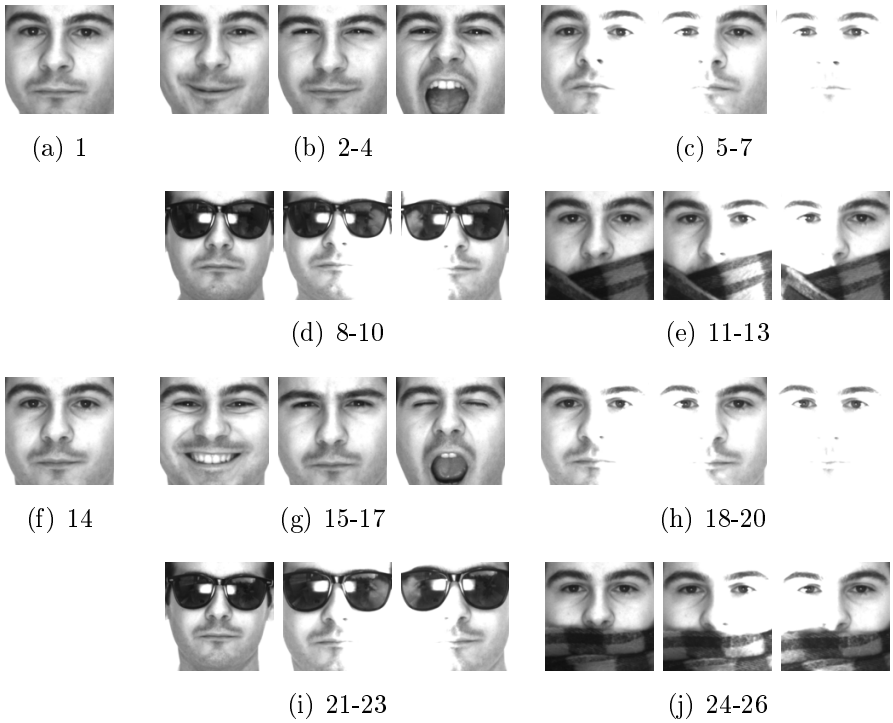
2.3.2 Base Aleix and Robert (AR)

Cette base de données a été créée par Aleix Martinez et Robert Benavente au *Computer Vision Center (CVC)* à l'UAB [78]. Elle contient plus de 4000 images couleur correspondant aux visages de 126 personnes (70 hommes et 56 femmes). Les images sont acquises en vue de face avec des conditions différentes d'expression faciale, d'éclairage, et d'occultation (lunettes de soleil et écharpe). Chaque personne a participé à deux séances, espacées de deux semaines (14 jours). Les mêmes images ont été prises dans les deux sessions. Chaque personne a au total 26 images (13 images par session), comme illustré sur la figure 2.13. Les images sont classées dans des groupes différents : l'image 1 (c.f. figure 2.13(a)) : neutre ; images 2, 3, 4 (figure 2.13(b)) : expressions ; images 5, 6, 7 (figure 2.13(c)) : illumination ; images 8, 9, 10 (figure 2.13(d)) : lunettes + illumination ; images 11, 12, 13 (figure 2.13(e)) : écharpe + illumination. Les images des figures f, g, h, i , j ont été prises lors de la deuxième session avec le même scénario ((f) : neutre, (g) : expressions, (h) : illumination ; (i) : lunettes + illumination ; (j) : écharpe + illumination). De même que pour la base FERET, dans notre expérimentation, les images de la base AR sont coupées et alignées grâce aux coordonnées des deux yeux qui sont disponibles [6], les images sont de dimension 96×96 pixels (voir figure 2.13). Comme les deux sessions ne sont beaucoup espacées (seulement deux semaines), nous n'utiliserons que la première session pour évaluer notre algorithme (notons que les ensembles Fa et Dup2 de la base FERET ont été acquis à au moins 18 mois l'un de l'autre). Dans les tests sur la base AR, les vignettes d'images sont alignées de la même manière que celles de la base FERET ($\text{dis}_{yeux} = 44, \text{hauteur}_{visage} = \text{largeur}_{visage} = 96$).

2.3.3 Base Extended YaleB

La base YaleB créée par l'université de Yale [38] est la base standard pour évaluer la robustesse des systèmes de biométrie faciale en cas d'illumination variable. Elle se compose de 5760 images faciales de 10 individus capturées sous 9 poses et 64 conditions différentes d'éclairage. Récemment, elle a été mise à jour en ajoutant de nouveaux individus pour conduire à la base *Extended Yale B* qui contient des images de 38 individus et est donc plus difficile que la base Yale B. Pour cette base, c'est surtout la partie avec les variations d'illumination qui est utilisée car d'autres bases telles que la base FERET sont beaucoup complètes pour l'étude des variations de pose.

6. http ://personalpages.manchester.ac.uk/staff/timothy.f.cootes/data/tarfd_markup/



(a) 1 (b) 2-4 (c) 5-7

(d) 8-10 (e) 11-13

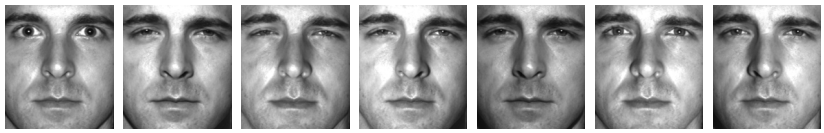
(f) 14 (g) 15-17 (h) 18-20

(i) 21-23 (j) 24-26

Figure 2.13 – Exemples d'images d'un individu de la base AR.

Dans ces bases Yale B, nous ne nous sommes intéressés qu'aux images de face. Pour chaque individu, les images de face ont été divisées en 5 groupes selon l'angle d'éclairage : groupe 1 (0° à 12°), groupe 2 (13° à 25°), groupe 3 (26° à 50°), groupe 4 (51° à 77°) et groupe 5 (plus de 78°). La figure 2.14 montre un exemple d'images de chaque groupe pour un individu donné. Au total, dans la base Yale B, les groupes 1, 2, 3, 4 et 5 contiennent respectivement 70, 120, 120, 140 et 190 images alors que dans la base *Extended* Yale B, ces groupes contiennent respectivement 263, 456, 455, 526, 714 images.

Ces bases sont mises à disposition dans deux formats : les images originales avec les coordonnées de points caractéristiques (les chercheurs peuvent donc faire l'alignement eux-même) et les images déjà alignées. Dans cette thèse, nous utilisons les images déjà alignées par les auteurs. Ces images sont de taille 168 x 192 pixels et dans nos expérimentation, on les réduit par un facteur $\frac{1}{2}$, ce qui conduit aux images de 84 x 96 pixels. Cette taille est plus petite que celle des images des bases FERET et AR. Ce choix nous permet de ne pas introduire de distorsions sur les images en conservant le même ratio hauteur/lageur. Quelques chercheurs ont coupé les images de la base *Yale B* eux-même avec une taille plus grande (200 x 200 pixels dans [25]). Leurs images sont donc moins difficiles que les nôtres.



(a) Groupe 1 (0° à 12°)

(b) Groupe 2 (13° à 25°)

(c) Groupe 3 (26° à 50°)

(d) Groupe 4 (51° à 77°)

(e) Groupe 5 (plus de 78°)

Figure 2.14 – Exemples d'images de la base Yale B pour un individu donné.

Figure 2.15 – Paires de la base LFW *aligned*.

2.3.4 Base Labeled Faces in the Wild (LFW)

Très récemment, une nouvelle base de visages intéressante nommée *Labeled Faces in the Wild* [7] a été présentée. Cette base de données qui contient 13233 images de 5749 personnes est considérée naturelle car les images ont été collectées directement sur le site d'information *Yahoo ! News* en utilisant un détecteur automatique de visage. Cette base est plus difficile pour les chercheurs car aucune contrainte sur les paramètres de prise de vue n'a été imposée (la base contient donc des variations importantes de pose, d'âge, d'expression, des images de mauvaise qualité, etc.). Dans le protocole d'évaluation associé à cette base, la question de la reconnaissance de visages devient une tâche de comparaison de l'identité de deux images dans une paire : étant donnée une paire d'images faciales, il s'agit de déterminer si elles sont issues de la même personne ou pas. Grâce aux étiquettes associées aux images originales, une paire d'images de la base LFW est soit classée dans la catégorie *Same* si les deux images proviennent d'une même personne, soit classée dans la catégorie *Diff* si les deux images proviennent de deux personnes différentes. La figure 2.15 montre quelques exemples de paires d'images de cette base. Cette base est disponible sous trois versions : *original* qui contient les images détectées sans alignement, *funneled* qui contient les images alignées automatiquement par l'algorithme de Huang *et al.* [48], *aligned* qui contient les images alignées automatiquement par l'algorithme de Wolf *et al.* [136]. Toutes les images sont de une taille 250 x 250 pixels. Dans cette thèse, nous avons coupé les vignettes de visage à partir des images *aligned*. Ces vignettes sont de dimension 120 $\times$ 120 et sont centrées sur les images *aligned*. Comme cette base est très difficile et pour pouvoir faire des comparaisons pertinentes entre nos méthodes et les méthodes existantes appliquées sur cette base, nous n'avons pas réduit plus les dimensions des images.

7. http ://vis-www.cs.umass.edu/lfw/

Le protocole d'évaluation associé à la base LFW est unique : deux sous-ensembles *View 1* et *View 2* sont fournis et utilisés pour les différentes tâches. *View 1* est utilisé pour la mise au point des algorithmes, la sélection des éventuels modèles (par exemple, si on veut utiliser le SVM pour la classification, on peut utiliser ce sous-ensemble pour choisir les noyaux SVM, etc.). A cette fin, *View 1* se compose du sous-ensemble *Training* qui contient 1100 paires *Same* et 1100 paires *Diff* et le sous-ensemble *Test* avec 500 paires *Same* et 500 paires *Diff*. Une fois que le modèle est choisi (le type de noyau de SVM, par exemple), on utilise l'ensemble *View 2* pour évaluer les performances (*View 2* n'est utilisé que pour l'évaluation finale des performances). *View 2* se compose de 10 parties différentes. Chaque partie contient 300 paires *Same* et 300 paires *Diff*. Il faut souligner que les images de chacune des 10 parties de *View 2* sont différentes les unes des autres. De même, les images des deux ensembles *View 1* et *View 2* sont différentes. Dans *View 2*, afin d'estimer les performances de reconnaissance, on calcule le taux de classification pour une partie en utilisant les 9 autres comme base d'apprentissage (notons que le rôle de cette base d'apprentissage est tout à fait différent celui du sous-ensemble *Training* de *View 1*, à savoir qu'on peut utiliser cette base d'apprentissage pour choisir le seuil optimal pour mieux distinguer les classes *Same* et le *Diff*, et non pas pour la mise au point des algorithmes). Puis on répète ce calcul en changeant le rôle de chaque partie. Finalement, les performances finales sont rapportées en calculant le taux moyen de classification exacte et l'erreur standard obtenue lors des 10 tests différents[8].

2.4 Conclusions

Dans ce chapitre, nous avons tout d'abord présente une brève bilan des méthodes les plus populaires utilisées en reconnaissance faciale. Plus spécifiquement, nous avons mis en evidence les méthodes qui sont applicables pour la reconnaissance de visages à partir d'une seule image. Puis deux des méthodes les plus connues pour l'extraction des caractéristiques faciales, étape indispensable dans les systèmes de reconnaissance de visages, sont décrites. Finalement, les bases de données de visages et les protocoles d'évaluation qui ont été utilisés dans ce travail sont présentés.

8. $S_E = \frac{\sigma}{\sqrt{10}}$ où σ est l'estimation de la déviation standard.

Chapitre 3

Robustesse aux variations d'éclairement : prétraitement par filtrage rétinien

Il est évident que les variations d'illumination affectent les performances des systèmes de biométrie faciale puisque des conditions différentes d'éclairement peuvent conduire à des images très différentes d'une même personne, comme montré sur la figure 2.14. Dans [1], Adini *et al.* ont observé que les variations d'illumination sont souvent plus importantes que les variations d'identité. Visuellement, on peut voir sur la figure 3.1 que les différences entre deux images de visage de deux individus différents (paire de gauche) peuvent être moins importantes que les différences entre deux images d'un même individu prises avec des conditions d'illumination différentes (paire de droite). Ceci explique la chute des performances des systèmes de reconnaissance faciale en cas de variations d'illumination : les variations inter-personnelles deviennent moins significatives que les variations intra-personnelles.

Figure 3.1 – L'apparence du visage change de manière importante cas d'illumination variable.

Dans ce chapitre, nous discuterons de l'effet des variations d'illumination sur les performances de la reconnaissance de visage et nous proposerons une solution pour remédier à ce problème. Il est possible de résoudre le problème de l'éclairage à plusieurs niveaux du système : au cours du prétraitement, lors de l'extraction des caractéristiques et lors de

la classification. En supposant que les conditions d'illumination sont imposées par l'environnement, non par l'individu, nous proposons de résoudre ce problème lors de l'étape de prétraitement juste après la détection des visages (voir figure 3.2). De plus, nous sommes convaincus qu'un bon prétraitement permettra d'améliorer les performances des étapes suivantes. Dans ce chapitre, nous allons présenter une nouvelle méthode de prétraitement qui s'appuie sur les propriétés de la rétine humaine en vue de normaliser les variations d'illumination. La motivation résulte du fait que naturellement la rétine nous permet de voir des objets dans différentes conditions d'éclairage.

Ce chapitre commence par un état de l'art sur les méthodes existantes. Puis, on décrit brièvement la rétine et ses propriétés. Ensuite nous détaillons la méthode proposée ainsi que les parties expérimentales destinées à en valider les performances. Un bilan et quelques discussions sont proposés à la fin du chapitre.

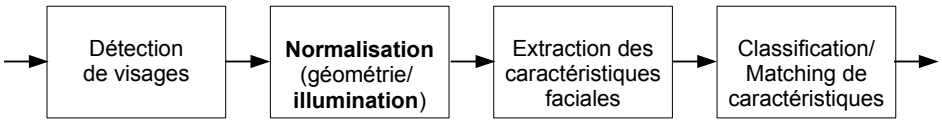


Figure 3.2 – Traitement du problème de l'illumination lors de l'étape de prétraitement.

3.1 Etat de l'art

La robustesse aux conditions d'illumination étant l'un des défis majeurs pour les systèmes de reconnaissance de visage, de nombreux travaux se sont attaqués à ce problème. Ces travaux peuvent être classés en trois grandes catégories que nous allons décrire.

3.1.1 Extraction de caractéristiques invariantes à l'illumination

Les méthodes de cette catégorie essaient d'utiliser des caractéristiques d'image invariantes aux changements d'illumination. Il a été montré théoriquement que dans le cas général, il n'existe pas de fonction invariante à l'illumination [86]. Pour le problème spécifique de la biométrie faciale, certaines représentations d'image ont été considérées comme invariantes à l'illumination, telles que les contours d'images [58], les gradients d'images [20] et les images filtrées par convolution avec des ondelettes de Gabor [20, 76]. Cependant, l'étude empirique d'Adini [1] a montré qu'aucune de ces représentations n'est efficace pour être invariante aux grandes variations d'illumination causées par des directions différentes de la lumière. Cette observation a ensuite été formellement prouvée dans [23], où les auteurs ont montré qu'il n'existe pas de fonction qui soit à la fois discriminante et invariante à l'illumination. Ils démontrent ce résultat à partir du fait que pour

deux images, il existe toujours une famille de surfaces, albédos et sources de lumière qui peuvent les produire. Par conséquent, la recherche d'une méthode de reconnaissance faciale robuste à l'illumination basée seulement sur des caractéristiques faciales a été abandonnée par la communauté (bien que dans des travaux plus récents, on ait trouvé de nouvelles caractéristiques plus robustes à l'illumination, telles que les indices LBP [2]).

3.1.2 Modèle d'illumination

L'idée de ce type d'approche est de construire à partir d'un ensemble d'images de visages acquises sous différentes conditions d'illumination un modèle d'évolution en fonction de l'illumination. Mais modéliser l'incidence des variations d'illumination sur une image est une tâche difficile. Pour s'en convaincre, il suffit de s'intéresser au nombre de degrés de liberté nécessaires pour décrire la lumière. Par exemple, la pose d'un visage par rapport à la caméra a six degrés de liberté, trois rotations et trois translations. L'expression du visage a une dizaine de degrés de liberté si l'on considère le nombre de muscles qui provoquent les changements d'expression. Pour décrire la lumière qui éclaire un visage, il faut connaître l'intensité de la lumière frappant chaque point du visage dans toutes les directions. Autrement dit, la lumière est fonction de la position et de la direction, ce qui signifie que la lumière a un nombre important de degrés de liberté. Beaucoup de travaux ont été faits et il en résulte que les variations d'illumination peuvent néanmoins être modélisées en utilisant moins de 10 degrés de liberté.

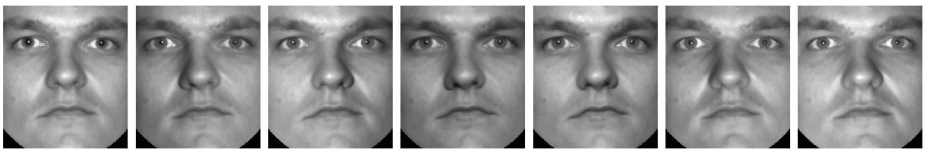
Dans ces méthodes, la phase d'apprentissage est tout d'abord effectuée en utilisant une base contenant plusieurs images d'une même personne acquises sous des conditions différentes d'éclairement afin de construire le modèle de cette personne. Ce modèle est ensuite utilisé pour la tâche de reconnaissance. Ces approches utilisent le modèle de Lambert pour décrire les images : une image est définie à partir de la normale à la surface de l'objet n, de son albédo et de la source lumineuse s qui l'éclaire. Ainsi, la luminance d'un point situé à la position (x, y) dans l'image peut s'écrire :

$$\begin{array}{ll} I(x,y) = \rho(x,y) \cdot n^T(x,y) \cdot s & \text{sans ombre} \\ I(x,y) = max(\rho(x,y) \cdot n^T(x,y) \cdot s, 0) & \text{avec ombre} \end{array} \tag{3.1.1}$$

A partir de cette modélisation lambertienne et avec l'hypothèse que la forme d'un objet est convexe et que l'ensemble des images d'un objet sous des conditions différentes d'illumination forme un sous espace vectoriel, les algorithmes essaient de récupérer les informations d'albédo et de normale à la surface qui sont supposées constantes.

Cône d'illumination

Belhumeur et Kriegman [9] ont montré qu'un ensemble d'images d'un objet dans une même pose, sous des conditions différentes d'illumination forment un cône convexe, appelé cône d'illumination. Ils ont également trouvé qu'un cône d'illumination peut être construit à partir de quelques images acquises sous des éclairages différents si la surface de réflexion peut être considérée comme Lambertienne, et qu'un cône d'illumination peut être bien approché par un sous espace vectoriel de faible dimension. Dans [38], Georghiades *et al.* ont exploité ces observations pour la reconnaissance de visages en conditions variables d'illumination en proposant de construire un cône individuel d'illumination à partir de sept images faciales acquises sous des conditions différentes d'illumination (voir figure 3.3(a)). Ils ont caractérisé le cône convexe d'illumination par un ensemble de rayons extrêmes. Un visage spécifique est donc représenté par les paramètres de forme et d'albédo qui sont estimés en utilisant sept images d'apprentissage pour chaque individu. Plusieurs images sous conditions différentes d'illumination sont ensuite générées en utilisant les paramètres estimés. La figure 3.3(b) montre quelques exemples des images générées par le cône construit à partir des images de la figure 3.3(a).



(a)

(b)

Figure 3.3 – Illustration du cône d'illumination [38] : (a) Sept images en entrée pour la construction d'un cône ; (b) Exemples d'images générées par le cône avec de nouvelles conditions d'illumination.

Ce travail a effectivement produit d'excellentes performances, mais il s'accompagne d'un temps de calcul très élevé. Le cône exact d'illumination est très difficile à calculer à cause du grand nombre de rayons extrêmes, nombre en $O(n^2)$ où n est le nombre de pixels sur le cône.

Harmonique sphérique

Dans [5], Basri et Jacobs ont proposé de représenter l'illumination par un harmonique sphérique, et ont montré que la réflexion lambertienne est produite par une convolution entre la fonction d'illumination et le noyau lambertien. En observant que le noyau lambertien ne contient que des composantes basses fréquences avec plus de 99 pourcents de son énergie réfléchie dans les neuf premières composantes, ils ont prouvé que dans n'importe quelle condition d'illumination, un sous espace vectoriel de dimension 9 linéaire peut capturer au moins 98 pourcents de variabilité de la fonction de réflexion. Autrement dit, un cône d'illumination peut être approché très précisément par un sous espace vectoriel de dimension 9, formé par neuf images harmoniques. Cependant, afin d'obtenir ces images harmoniques, la structure 3D de l'objet doit être connue a priori. De la même manière, Lee et Kriegman [67] ont proposé d'obtenir un sous espace vectoriel de dimension 9 en utilisant neuf images acquises sous neuf directions différentes d'illumination.

Généralement, les méthodes de cette catégorie peuvent conduire à des taux de reconnaissance très élevés. Cependant, elles ont deux inconvénients. Premièrement, plusieurs images d'un individu acquises sous des conditions d'illumination différentes sont nécessaires ce qui n'est pas envisageable en contexte applicatif de vidéosurveillance. Deuxièmement, la phase d'apprentissage du modèle est très complexe et demande beaucoup de temps. Cette approche n'est pas donc adaptée à tout type d'application.

3.1.3 Suppression des variations d'illumination

La troisième catégorie d'approches consiste à transformer l'image en forme canonique dans laquelle les variations d'illumination sont supprimées. Contrairement aux algorithmes de la deuxième catégorie, ces approches requièrent l'acquisition d'une seule image seulement, il n'est pas utile de disposer de plusieurs images acquises sous des conditions différentes d'illumination pour chaque individu. Par ailleurs, ces méthodes de normalisation des variations d'illumination sont moins coûteuses en temps de calcul. Ceci est tout à fait adapté à notre contexte d'étude pour lequel la reconnaissance de visages est faite dans un contexte de vidéosurveillance avec des contraintes sur le nombre d'images de référence disponibles par individu et sur le temps de calcul. Voilà pourquoi nous avons choisi ce type d'approche dans le cadre de nos recherches.

Les méthodes traditionnelles pour supprimer les variations d'illumination sont l'égalisation d'histogramme (HE) et la correction de gamma (GIC).

- L'égalisation d'histogramme consiste à appliquer une transformation sur chaque pixel de l'image, et donc à obtenir une nouvelle image à partir d'une opération indépendante sur chacun des pixels. L'égalisation d'histogramme permet de mieux

répartir les intensités sur l'ensemble de la plage de valeurs possibles.
- La correction de gamma est une technique largement utilisée pour faire de la compression dynamique afin d'améliorer les régions sombres.

Toutes les autres méthodes plus élaborées sont basées sur les propriétés du système visuel humain, mises en évidence par Land et McCann [63].

3.1.3.1 Théorie rétinex de Land

La théorie Retinex (mélange des deux mots "Retina" et "Cortex", pour montrer qu'à la fois l'oeil et le cerveau sont impliqués dans le processus) développée par Land et McCann [63] en 1971 vise à décrire comment le système visuel humain perçoit la couleur/la luminosité d'une scène naturelle. Le système visuel humain est effectivement capable de construire une représentation visuelle des détails et des couleurs dans une gamme importante de variations d'éclairage [55].

Le système visuel humain a également la propriété de constance de couleur et de luminance, ce qui nous permet d'identifier des objets. Les lois de l'optique nous indiquent que l'application d'une lumière rouge sur une pomme verte n'a pas le même effet que l'application d'une lumière blanche (lumière du soleil par exemple) sur la même pomme verte. Cependant, notre vision tente de voir la même couleur, indépendamment de la lumière appliquée. Concernant cette propriété de constance, il est indéniable que les performances de la vision humaine sont meilleures que celles de tous les systèmes artificiels existants. Afin de trouver un modèle de calcul pour simuler cette propriété, la théorie Retinex de Land a exposé deux hypothèses principales :

- Le système visuel humain effectue le même traitement de manière indépendante sur chacun des canaux de couleur.
- Dans chaque canal, à chaque pixel, l'intensité $I(x, y)$ est le produit de la réflectance $R(x, y)$ et de l'illumination $L(x, y)$, soit $I(x, y) = L(x, y)R(x, y)$.

L'obtention de la réflectance R peut être résolue en estimant l'illumination L. Dans la littérature, plusieurs algorithmes ont été présentés pour estimer l'illumination à partir de l'image acquise I, nous choisissons et présentons les méthodes les plus largement utilisées. En pratique, la deuxième hypothèse est utilisée sous la forme :

$$R(x, y) = log\frac{I(x, y)}{L(x, y)} = logI(x, y) - logL(x, y) \tag{3.1.2}$$

L'intérêt de cette formule réside d'une part dans le fait que cette fonction transforme la division en soustraction, donc réduit le coût de calcul et d'autre part dans le fait que la vision humaine traite la luminance selon une loi logarithmique. Il convient aussi de noter que dans notre cas, le processus utilise seulement les valeurs de luminance, car on ne travaille qu'avec des images à niveau de gris.

Dans ce qui suit, nous avons choisi de présenter en détails les deux méthodes les plus connues basées sur cette théorie rétinex : les méthodes SSR/MSR et SQI. Afin de montrer les évolutions dans l'approche, nous allons ensuite décrire la méthode proposée plus récemment, la méthode Rétinex Adaptatif. Finalement, quelques autres algorithms seront présentés brièvement. Avant de commencer, il convient de noter que dans certains cas, il n'est pas facile (et pas juste) de comparer les performances des méthodes de normalisation d'illumination dans le cas général. La meilleure approche est d'évaluer leurs performances dans le contexte considéré, à savoir la normalisation d'illumination pour la reconnaissance de visages.

3.1.3.2 Algorithme du Single/Multi Scale Retinex (SSR/MSR)

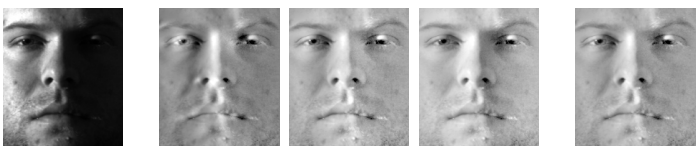
Dans [55], Jobson *et al.* ont proposé l'algorithme du Single Scale Retinex (SSR) où l'illumination L est estimée en utilisant un filtre passe-bas gaussien :

$$L = I * G \tag{3.1.3}$$

où $*$ est la convolution et G est le filtrage Gaussien d'écart-type σ :

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{3.1.4}$$

L'intérêt de l'utilisation du filtre gaussien peut être expliqué par le fait que l'illumination d'une image étant supposée être basse fréquence, une méthode simple pour l'obtenir est donc d'utiliser un filtre passe-bas. Par ailleurs, la valeur de la réflectance dans la formule 3.1.4 est le ratio de l'intensité du pixel par la moyenne pondérée des pixels de son voisinage et Land [63] a déjà montré que le traitement de l'image en termes de ratio de luminance avec les points adjacents dans un voisinage va générer une indépendance à l'éclairement. C'est la raison pour laquelle cette méthode est aussi appelée algorithme du *Center/Surround Retinex*. L'écart-type σ du filtre gaussien est appelé échelle du SSR.



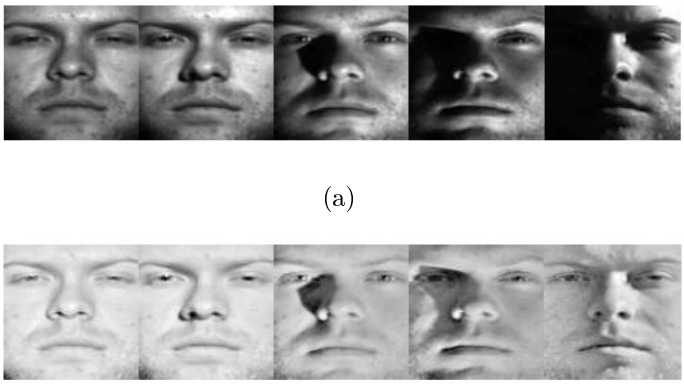
(a) (b) SSR, $\sigma = 3, 15, 40$ (c) MSR

Figure 3.4 – Effet du SSR et MSR.

L'inconvénient du SSR est que l'image obtenue dépend trop de la valeur d'échelle. Sur la figure 3.4.b, on observe que dans l'image obtenue avec une valeur d'échelle faible, la région sombre est grisée et mélangée avec le reste de l'image. A l'opposé, quand l'échelle

s'élève, l'ombre est plus en plus visible (c.f. figures 3.4(c,d)). Autrement dit, l'algorithme du SSR fonctionne bien soit pour les intensités faibles soit pour les intensités élevées.

Jobson *et al.* [55] ont amélioré l'algorithme SSR en proposant le Multi-Scale RetineX (MSR) qui est la combinaison de plusieurs SSR à échelle différente. On applique l'algorithme du SSR sur l'image initiale à plusieurs échelles $(c_1, c_2, ..., c_n)$ et on pondère ensuite les résultats avec les poids $(w_1, w_2, ..., w_n)$.



(a)

(b)

Figure 3.5 – Exemples d'images traitées par l'algorithme du MSR : (a) images originales de la base YaleB ; (b) images traitées.

Contrairement à l'algorithme du SSR, le MSR a des performances plus stables : les niveaux d'intensité élevée et faible sont correctement corrigés. La figure 3.5.b montre quelques images traitées par l'algorithme du MSR [1]. Bien que les images soient éclaircies, les ombres sont toujours visibles.

3.1.3.3 Algorithme du Self-Quotient Image (SQI)

Dans [131], le Self-Quotient Image (SQI) a conduit à une amélioration des performances pour la reconnaissance de visages en conditions variables d'illumination en comparaison avec les algorithmes SSR/MSR. En observant que le filtrage gaussien utilisé par Jobson [55] en étant isotrope, crée des effets de halo autour des régions de contours, les auteurs ont proposé de modifier le noyau gaussien de telle manière qu'il fonctionne comme un filtrage anisotrope. À cette fin, le noyau est pondéré, pour chaque région de convolution. Soit W le poids et G le noyau gaussien, l'algorithme du Single Scale SQI fonctionne de la même façon que l'algorithme du SSR, l'image en sortie est :

$$R_{SSQI}(x, y) = logI(x, y) - log[WG(x, y) * I(x, y)] \tag{3.1.5}$$

1. Dans le reste du paragraphe, les performances des autres algorithms seront illustrées en utilisant les images de la figure 3.5(a).

Wang *et al.* ont ensuite appliqué l'avantage des méthodes MSR en combinant plusieurs Single Scale SQI afin d'obtenir une nouvelle méthode plus performante, le Multi Scale SQI notée finalement SQI :

$$R_{SQI}(x,y) = \sum_{i=1}^{n} w_i\{logI(x,y) - log[W_iG_i(x,y) * I(x,y)]\} \tag{3.1.6}$$

La figure 3.6(a) montre un exemple d'images traitées par le Single Scale SQI (l'image à gauche est l'image en entrée et l'image à droite est la sortie) et la figure 3.6(b) montre des images traitées par le Multi Scale SQI. Sur cette figure, on constate que l'algorithme du SQI traite les ombres mieux que le MSR (voir en particulier les zones à l'intérieur des ellipses et comparer avec la figure 3.5(b)).

(a)

(b)

Figure 3.6 – Exemples d'images normalisées par l'algorithme du SQI. (a) single SQI ; (b) multi SQI.

3.1.3.4 Algorithme du Retinex Adaptatif

En remarquant que les méthodes précédentes ne peuvent pas supprimer complètement les ombres, car elles n'ont pas considéré proprement la discontinuité d'illumination, dans [96], Park *et al.* ont proposé d'estimer l'illumination en utilisant un filtrage qui peut préserver les discontinuités. La méthode est essentiellement basée sur un lissage adaptatif qui utilise des convolutions itératives et deux mesures de discontinuité. Sur la figure 3.7 qui montre des exemples d'images traitées par le retinex adaptatif, on peut voir que les ombres sont encore mieux traitées, mais la méthode introduit du bruit (voir en particulier la zone autour de l'oeil à gauche dans la quatrième image).

Figure 3.7 – Exemples d'images normalisées par l'algorithme du Retinex Adaptatif.

3.1.3.5 Autres algorithmes

Il est facile de voir que l'illumination peut être modélisée par un filtre de type lissage de flou d'une image (le filtre gaussien utilisé dans la méthode SSR par exemple). Le processus de diffusion, qui est basé sur des équations aux dérivées partielles et qui peut produire du flou dans l'image, est aussi utilisé pour l'estimation de l'illumination. Les exemples de telles méthodes sont la diffusion isotrope, la diffusion anisotrope, etc. Quelques exemples d'images normalisées par diffusion isotrope et par diffusion anisotrope sont affichées sur les figures 3.8(a) et (b), respectivement.

(a)

(b)

Figure 3.8 – Exemples d'images normalisées par les algorithmes basés sur la diffusion : (a) isotrope ; (b) anisotrope.

Le filtrage homomorphique, développé par Stockham dans les années 1960 est une technique classique de normalisation pour laquelle l'image d'entrée est tout d'abord transformée en logarithme, puis dans le domaine des fréquences par transformée de Fourier. Les composantes de hautes fréquences sont renforcées et les composantes de basses fréquences sont filtrées. Finalement, l'image est retransformée dans le domaine spatial par transformée de Fourier inverse et en prenant l'exponentielle du résultat. La figure 3.9(a) montre des exemples d'images lissées par ce filtrage. Les méthodes décrites au dessus suppriment les variations d'illumination mais leurs performances pour la reconnaissance de visages sont moins bonnes que celles des méthodes SQI et Retinex Adaptatif.

Une autre méthode qui utilise les ondelettes pour estimer la luminance a été présentée

(a)

(b)

Figure 3.9 – Exemples d'images normalisées par filtrage homomorphique et par la méthode de Zhang [143].

récemment dans [143] et quelques exemples des images traitées par cette méthode sont montrées sur la figure 3.9(b). Comme nous pouvons le voir, cette méthode introduit des effets de distortion sur les images.

Nous venons de voir les méthodes de normalisation d'illumination qui sont largement utilisées dans la littérature. Ces méthodes visent à estimer l'illumination afin de trouver la réflectance. Cependant, l'estimation d'illumination n'est pas une tâche facile. Bien que ces méthodes soient moins complexes que celles de la deuxième catégorie, leur complexité est encore élevée : les méthodes MSR et SQI nécessitent des estimations d'illumination à plusieurs *échelles* différentes (ces échelles sont élevées) ; le poids W du noyau gaussien (équation 3.1.3.3) dans la méthode SQI est lent à calculer ; le Retinex Adaptatif a besoin de cycles itératifs pour estimer l'illumination ; et la méthode dans [143] est basée sur les ondelettes qui ne sont pas rapides à calculer.

Contrairement aux méthodes basées sur la théorie retinex, qui considèrent que la rétine et le cerveau [2] concourent ensemble au processus de normalisation d'illumination, nous proposons une méthode qui s'appuie sur les performances de la rétine uniquement. Notre méthode essaie de trouver directement R au lieu d'estimer l'illumination L. Notre motivation réside dans le fait que nous croyons (et cela peut être discutable) dans la capacité naturelle de la rétine à voir des objets dans des conditions d'illumination différentes alors que le rôle du cerveau est plutôt de reconstruire une représentation visuelle des détails.

2. Le comportement précis du cerveau est toujours un mystère.

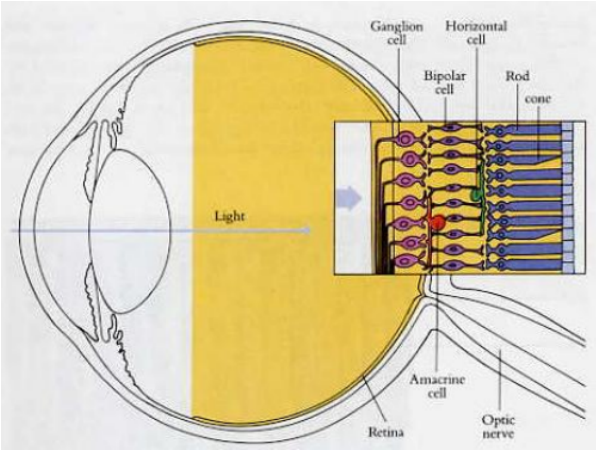


Figure 3.10 – La rétine tapisse le fond de l'oeil. La lumière passe à travers les cellules bipolaires et les cellules amacrines et atteint la couche des photorécepteurs d'où elle repart [http ://hubel.med.harvard.edu/bio.htm].

3.2 La rétine : propriétés et modélisation

La rétine tapisse le fond de l'oeil (c.f. figure 3.10). Elle est constituée de plusieurs couches de cellules : la couche des photorécepteurs (cônes et bâtonnets) , la couche des cellules horizontales, des cellules bipolaires, des cellules amacrines et enfin la couche des cellules ganglionnaires. Ces couches sont séparées par des zones de connexions synaptiques : la couche plexiforme externe (PLE) et la couche plexiforme interne (PLI). Les cellules différentes ont des propriétés particulières et ce paragraphe montrera que les cellules bipolaires sont capables de supprimer les variations d'illumination et le bruit. Il faut aussi noter que la rétine est capable de traiter des signaux spatiaux et temporels mais dans le cadre de ce travail, nous nous limitons à des images statiques et par conséquence, on n'étudiera que les traitements spatiaux de la rétine.

3.2.1 Photorécepteurs : adaptation locale à la lumière

La couche des photorécepteurs est la couche la plus profonde, par rapport à l'arrivée de la lumière. Deux types de photorécepteurs différents se distinguent : les bâtonnets et les cônes. Les bâtonnets sont responsables de la vision de faible intensité lumineuse (vision de nuit) et les cônes sont responsables de la vision des couleurs et des hautes intensités lumineuses (vision de jour). Les photorécepteurs sont ainsi capables d'adapter leur réponse à la dynamique de l'intensité lumineuse qu'ils reçoivent, cette capacité est appelée *compression adaptative* ou *logarithmique*.

L'adaptation réalisée par les photorécepteurs peut être modélisée par l'équation :

$$y = \frac{x}{x + x_0}, \tag{3.2.1}$$

où y est le signal adapté, x est le signal d'entrée, et x_0 est le facteur d'adaptation.

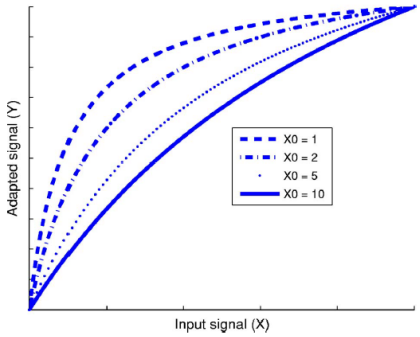


Figure 3.11 – Fonction d'opération non-linéaire pour différents facteurs d'adaption x_0.

La figure 3.11 montre l'allure de cette fonction selon la valeur x_0. Le but de cette fonction est d'étaler les petites valeurs du signal d'entrée. Si x_0 est faible, le signal en sortie augmentera de manière sensible. En revanche, pour les grandes valeurs du facteur d'adaptation x_0, le signal adapté est peu modifié par rapport au signal en entrée.

Pour que la compression soit automatique, plusieurs méthodes sont proposées pour déterminer la valeur optimale du facteur d'adaptation x_0. Une solution est de prendre une valeur x_0 égale à l'intensité moyenne de l'image. Cette solution est correcte si l'intensité lumineuse de l'image est à peu près équilibrée, c'est-à-dire si l'histogramme est relativement plat autour de la valeur moyenne. Cependant, si les zones de l'image ne sont pas éclairées de la même manière, une telle compression égalisera l'intensité de manière identique sur toutes les zones de l'image (c.f. figure 3.12(b)).

Il est donc plus intéressant de trouver un facteur qui s'adapte localement à l'intensité de l'image. Pour cela une valeur différente pour chaque pixel de l'image est calculée. Cette valeur peut être obtenue en faisant une moyenne locale ; pour cela un filtrage passe-bas gaussien est appliqué à l'image d'entrée. Cette image filtrée passe-bas est ensuite appliquée comme facteur d'adaptation sur l'image pour obtenir l'image adaptée à la lumière (figure 3.12(c)). Une autre solution est de combiner ces deux approches : un filtrage passe-bas est appliqué à l'image d'entrée et le facteur d'adaptation est la somme de l'intensité moyenne de l'image d'entrée et de l'intensité de l'image filtrée passe-bas à chaque pixel. La figure 3.12(d) montre les images adaptées à la lumière en utilisant ces facteurs. Sur la figure 3.12, on observe que les zones sombres sont ainsi plus éclaircies que les zones claires car elles sont compressées par un facteur plus petit. Parmi les trois images (b), (c) et (d), on observe que les images (d) sont les mieux adaptées à la lumière. En conclusion, à la sortie des photorécepteurs, l'image a gagné en contraste : les zones sombres ont été éclaircies sans modification des zones claires de l'image.

(a) (b) (c) (d)

Figure 3.12 – Adaptation des photorécepteurs. (a) : images originales ; images obtenues avec le facteur égal à : l'intensité moyenne de l'image (b) ; la moyenne du voisinage du pixel courant (c) ; la somme de l'intensité moyenne de l'image et de la moyenne du voisinage du pixel courant (d).

3.2.2 Couche Plexiforme Externe (PLE)

La couche plexiforme externe correspond à la zone de jonction entre les photorécepteurs, les cellules bipolaires et les cellules horizontales. Chaque jonction appelée triade synaptique permet des interactions entre les signaux de luminance reçus par les photorécepteurs et les signaux porteurs de l'information de luminance locale délivrés par les cellules horizontales. Les cellules bipolaires participent à ces interactions et transmettent ensuite le résultat vers les couches de cellules suivantes.

Le réseau des photorécepteurs ne réalise pas seulement une adaptation de la lumière mais il réalise aussi un filtre passe-bas. Ceci aboutit à une image dans laquelle le bruit haute fréquence est fortement atténué et l'information visuelle basse fréquence préservée. L'information est ensuite filtrée par le réseau de cellules horizontales qui effectue un filtrage passe-bas. Puis, les cellules bipolaires effectuent la différence entre la réponse des photorécepteurs et celles des cellules horizontales. Cela veut dire que le réseau de cellules bipolaires agit comme un filtre passe-bande. Ce filtre a donc deux effets : premièrement, il modélise le rôle des photorécepteurs qui minimisent le bruit haute fréquence. Deuxièmement, il modélise l'action des cellules horizontales pour éliminer l'illumination basse fréquence. La fréquence de coupure des cellules horizontales est donc inférieure à celle des photorécepteurs. En conclusion, à la sortie du réseau des cellules bipolaires, on observe une image dont l'illumination basse fréquence et le bruit haute fréquence ont été supprimés.

Pour modéliser le comportement des cellules bipolaires, on utilise souvent deux filtres passe-bas avec deux fréquences de coupure différentes qui correspondent aux actions des photorécepteurs et des cellules horizontales [11]. La différence de ces filtres est ensuite calculée. Dans notre méthode, deux filtres passe-bas gaussiens sont utilisés pour mo-

déliser les performances des photorécepteurs et des cellules horizontales. Le réseau des cellules bipolaires agit donc comme un filtre différence de gaussiennes (Difference of Gaussian - DOG). Sur la figure 3.13, la courbe bleue modélise l'effet du filtre différence de gaussiennes (différence entre la courbe verte et la courbe rouge). On observe que les fréquences très hautes et très basses sont éliminées alors que les fréquences moyennes sont préservées.

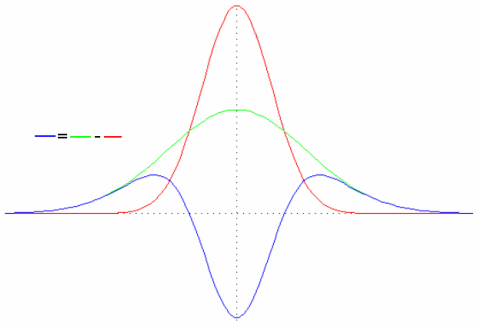


Figure 3.13 – Filtre différence de gaussiennes.

3.2.3 Couche Plexiforme Interne (PLI)

La couche PLI est le dernier étage de traitement au niveau de la rétine avant le nerf optique. L'information entrante résulte du traitement de l'information visuelle par la PLE, elle est transmise par les cellules bipolaires. Au niveau de cette couche, on trouve des interactions entre les cellules bipolaires, les cellules ganglionnaires et les cellules amacrines. Le résultat des interactions au niveau de la PLI est disponible au niveau des cellules ganglionnaires dont les axones forment le nerf optique. Cette couche traite plutôt les informations spatio-temporelles ou de mouvement [11] et donc elle ne nous intéresse pas.

3.3 Méthode proposée

Comme nous venons de le voir, un modèle constitué d'un opérateur non linéaire et d'un filtre passe-bande peut être utilisé pour supprimer les variations d'illumination. Dans notre modèle, *plusieurs* compressions logarithmiques sont appliquées consécutivement pour un filtre adaptation de la lumière plus efficace et une troncature est utilisée après le filtre passe-bande pour améliorer le contraste global de l'image.

3.3.1 Compressions logarithmiques multiples

Dans le travail de Meylan *et al.* [81], les auteurs se sont intéressés à la propriété d'adaptation à la lumière de la rétine et ont modélisé le comportement de la rétine entière par deux compressions logarithmiques, qui correspondent à la PLE et à la PLI, respectivement. Ils ont également observé que ces doubles compressions conduisent à un très bon filtre d'adaptation de la lumière avec une bonne discrimination visuelle. On s'inspire de cette observation en appliquant *plusieurs* compressions logarithmiques dans la première étape de notre modèle. Effectivement, nous avons déjà montré dans [128] que l'utilisation d'une double compression logarithmique conduit à des performances optimales sous réserve que les images de la base de référence aient été acquises sans variation d'illumination (des performances parfaites ont été obtenues sur tous les 5 sous-ensembles de la base Yale B). Nous allons prouver dans la partie expérimentale de ce chapitre que même si des variations d'illumination sont présentes sur les images de référence et sur les images requêtes, l'utilisation de compressions multiples donne toujours de bons résultats et nous déterminerons le nombre optimal de compression à considérer.

Le facteur d'adaptation de la première opération non linaire est calculé comme la somme de l'intensité moyenne de l'image d'entrée et de l'intensité de l'image filtrée passe-bas à chaque pixel :

$$F_1(p) = I_{in}(p) * G_1 + \frac{\overline{I_{in}}}{2} \tag{3.3.1}$$

où p est le pixel courant, $F_1(p)$ est le facteur d'adaptation du pixel p, I_{in} est l'intensité de l'image d'entrée, $*$ dénote l'opération de convolution, $\overline{I_{in}}$ est la valeur moyenne de l'image d'entrée, G_1 est le filtrage gaussien passe-bas 2D d'écart-type σ_1 :

$$G_1(x,y) = \frac{1}{2\pi\sigma_1^2} e^{-\frac{x^2+y^2}{2\sigma_1^2}} \tag{3.3.2}$$

où $x \in [-3\sigma_1, 3\sigma_1]$ et $y \in [-3\sigma_1, 3\sigma_1]$.

L'image d'entrée est ensuite traitée selon l'équation 3.2.1 en utilisant le facteur d'adaptation $F_1(p)$:

$$I_{la_1}(p) = (I_{in}(max) + F_1(p)) \frac{I_{in}(p)}{I_{in}(p) + F_1(p)} \tag{3.3.3}$$

Le terme $I_{in}(max) + F_1(p)$ est le facteur de normalisation où $I_{in}(max)$ est la valeur maximale d'intensité d'image.

La deuxième opération non linaire fonctionne de la même façon que la première, l'image adaptée à la lumière I_{la_2} est obtenue par :

$$I_{la_2}(p) = (I_{la_1}(max) + F_2(p)) \frac{I_{la_1}(p)}{I_{la_1}(p) + F_2(p)} \tag{3.3.4}$$

avec

$$F_2(p) = I_{la_1}(p) * G_2 + \frac{\overline{I_{la_1}}}{2} \tag{3.3.5}$$

et G_2 un filtre gaussien d'écart-type σ_2 différent.

$$G_2(x,y) = \frac{1}{2\pi\sigma_2^2} e^{-\frac{x^2+y^2}{2\sigma_2^2}} \tag{3.3.6}$$

En cas d'utilisation de plus de compressions, les compressions suivantes fonctionnent de la même manière et on obtient finalement l'image I_{la_n} où n est le nombre d'opérateurs non linaires utilisés. La figure 3.14 montre l'effet d'une compression logarithmique de la

(a) I_{in} (b) I_{la_1} (c) I_{la_2} (d) I_{la_2}

Figure 3.14 – Adaptation des photorécepteurs en fonction des paramètres. (a) : images originales ; (b) : images après une opération de compression adaptative ; (c) & (d) : images obtenues après deux opérations de compression adaptative avec des paramètres différents.

luminance sur deux images en fonction des paramètres. Visuellement, on observe que les images après deux opérations de compression (c.f. figure 3.14(c)) sont mieux adaptées à la lumière que celles après une seule opération (c.f. figure 3.14(b)). L'autre avantage de deux compressions logarithmiques consécutives, comme montré dans [81], est que l'image obtenue ne dépend pas de la taille du voisinage utilisé pour calculer le facteur d'adaptation (σ_1 et σ_2) : on ne voit pas de différence entre les images de la figure 3.14(c) ($\sigma_1 = 1, \sigma_2 = 1$) et celles de la figure 3.14(d) ($\sigma_1 = 1, \sigma_2 = 3$).

3.3.2 Filtrage de Différence de gaussiennes (DoG) et troncature

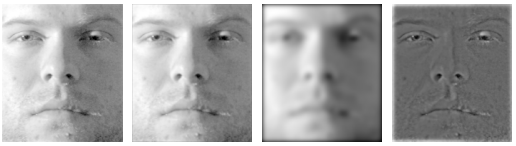
L'image I_{la_n} est ensuite traitée en utilisant un filtrage Différence de Gaussiennes (DoG) :

$$I_{bip} = DoG * I_{la_n}, \tag{3.3.7}$$

où DoG est calculé par :

$$DoG = \frac{1}{2\pi\sigma_{Ph}^2} e^{-\frac{x^2+y^2}{2\sigma_{Ph}^2}} - \frac{1}{2\pi\sigma_H^2} e^{-\frac{x^2+y^2}{2\sigma_H^2}}, \tag{3.3.8}$$

Les termes σ_{Ph} et σ_H correspondent aux écarts-types des filtres passe-bas gaussiens modélisant les performances des photorécepteurs et des cellules horizontales. Effectivement, l'image à la sortie des cellules bipolaires I_{bip} est la différence entre l'image à la sortie des photorécepteurs I_{Ph} et celle à la sortie des cellules horizontales I_H : $I_{bip} = I_{Ph} - I_H$, où I_{Ph} et I_H sont obtenues en appliquant des filtrages passe-bas gaussiens sur l'image I_{la_2} : $I_{Ph} = G_{Ph} * I_{la_n}, I_H = G_H * I_{la_n}$. La figure 3.15 montre l'effet du filtre différence de



(a) I_{la_2} (b) I_{Ph} (c) I_H (d) I_{bip}

Figure 3.15 – Effet du filtre différence de gaussiens.

gaussiennes sur l'image à la sortie de deux compressions logarithmiques (figure 3.15(a)). On observe que les variations d'illumination et les bruits sont bien supprimés (figure 3.15(d)). Comme indiqué précédemment, il faut que la fréquence de coupure des cellules horizontales soit inférieure à celle des photorécepteurs, σ_{Ph} doit donc être inférieure à σ_H. Par défaut, σ_{Ph} et σ_H sont mis à 0.5 et 3.5 respectivement dans le modèle. Un autre avantage du filtre DoG est le renforcement des contours de l'image [11].

Un inconvénient du filtre DoG est la réduction inhérente du contraste global de l'image. Pour pallier cet inconvénient, quelques valeurs extrêmes sont supprimées par une troncature afin d'améliorer le contraste d'image.

Afin de faciliter la troncature, on utilise tout d'abord une normalisation zero-mean pour que la dynamique d'image soit bien étalée. La soustraction de la moyenne $\overline{I_{in}}$ n'est pas nécessaire puisqu'elle est proche de 0 [3] :

$$I_{nor}(p) = \frac{I_{bip}(p) - \mu_{I_{bip}}}{\sigma_{I_{bip}}} = \frac{I_{bip}(p)}{\sqrt{E(I_{bip}^2)}} \tag{3.3.9}$$

Après normalisation, les valeurs d'image sont bien étalées et se situent principalement autour 0, quelques valeurs extrêmes sont supprimées par une troncature par un seuil Th

3. Bien que la moyenne soit déjà nulle, une normalisation est encore nécessaire pour que la dynamique d'image soit bien étalée. Nous remarquons que la soustraction de la moyenne $\overline{I_{in}}$ n'est pas nécessaire pour réduire le temps de calcul.

selon la relation :

$$I_{pp}(p) = \begin{cases} max(Th, |I_{nor}(p)|) & \text{if} \quad I_{nor}(p) \geq 0 \\ -max(Th, |I_{nor}(p)|) & \text{otherwise} \end{cases} \tag{3.3.10}$$

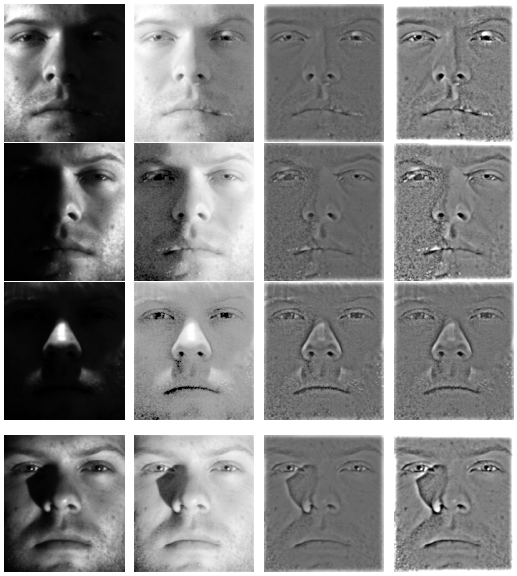
Le seuil Th est réglé de manière à supprimer environ 2-4% des valeurs extrêmes de l'image. La figure 3.16 montre les étapes principales de l'algorithme proposé. On observe qu'après la troncature, le contraste global de l'image est amélioré (c.f. figure 3.16(d)).

3.3.3 Propriété du modèle proposé

Notons qu'au laboratoire Gipsa-lab, les comportements des cellules bipôlaires sont typiquement simulées par une seule compression et un filtre *DoG* [11] alors que notre algorithme se compose de *plusieurs* compressions logarithmiques appliquées consécutivement, un filtre *DoG* et une troncature. Nous allons prouver empiriquement que ce filtre est plus efficace pour la normalisation d'illumination.

Notre algorithme a les avantages suivants :

1. Suppression des variations d'illumination et du bruit de l'image.
2. Renforcement des contours d'image.
3. Conservation du contraste global de l'image.



(a) I_{in} (b) I_{la_2} (c) I_{bip} (d) I_{pp}

Figure 3.16 – Effets des étapes de l'algorithme.

Avant de présenter les résultats expérimentaux, nous décrivons la méthode *Processing Sequence*(PS) [122] qui semble la méthode la plus performante dans le littérature.

La méthode PS

La méthode PS est constituée de trois étapes qui sont la correction de gamma, le filtre DoG et une normalisation. Etant donnée l'image I, elle est traitée par les opérateurs suivants :

1. Correction de gamma : $I(x,y) \leftarrow I(x,y)^{\gamma}$, où $\gamma \in [0,1]$
2. DoG filtre : $I \leftarrow I * DoG$
3. Normalisation : 4 normalisations successives

$$I(x,y) \leftarrow \frac{I(x,y) - \mu_I}{\sigma_I},$$
$$I(x,y) \leftarrow \frac{I(x,y)}{(mean(|I(x,y)|^a))^{1/a}},$$
$$I(x,y) \leftarrow \frac{I(x,y)}{(mean(min(\tau, |I(x,y)|^a)))^{1/a}},$$
$$I(x,y) \leftarrow \tau tanh(I(x,y)/\tau),$$

avec a et τ sont choisis de manière empirique.

Les objectifs de trois étapes de la méthode PS [122] sont d'égaliser la lumière (correction de gamma), de supprimer les variations d'illumination (filtre DoG) et d'améliorer le contraste, respectivement. D'un point de vue strict, les lecteurs peuvent considerer que la méthode PS (publiée en 2007) et notre algorithme (travail à la fin de l'année 2007 et soumis en 2008) présentent des similarités. Notre contribution a été d'améliorer ce filtre avec une augmentation des performances et une réduction du coût de calcul. Cependant, Tan et Triggs [137] n'ont pas dit que leur méthode est basée principalement sur les performances de la rétine alors que dans le travail publié [128], nous avons mis en évidence le lien entre notre méthode et le filtre rétinien. Il convient de souligner que le laboratoire Gipsa-lab est l'un des laboratoires pionniers sur les travaux de modélisation de la rétine.

3.4 Résultats expérimentaux

Nous évaluons les performances de l'algorithme de prétraitement proposé vis-à-vis de l'application reconnaissance de visages. Trois bases de données sont considérées : la base Yale B, la base FERET (les visages de face) et la base AR. Nous choisissons trois méthodes de reconnaissance : l'eigenface, la méthode basée sur le LBP, et la méthode basée sur les filtres de Gabor.

1. Bien que la méthode *Eigenface* soit très simple (beaucoup d'autres méthodes conduisent à de meilleurs taux de reconnaissance), ses résultats en reconnaissance associée à notre méthode de prétraitement sont très intéressants car ils prouvent l'efficacité de notre algorithme.

2. L'intérêt de choisir les méthodes à base de LBP et de filtres de Gabor est que ces deux méthodes utilisent des descripteurs de visage considérés comme a priori invariants aux changements d'illumination. Nous allons montrer que l'utilisation de ce type de caractéristique n'est pas suffisante en cas de grands changements d'illumination et que notre méthode de normalisation d'illumination améliore significativement les performances de ces méthodes.

Dans tous les tests, nous utilisons une classification au *plus proche voisin* pour obtenir les taux corrects d'identification.

3.4.1 Sélection de paramètres

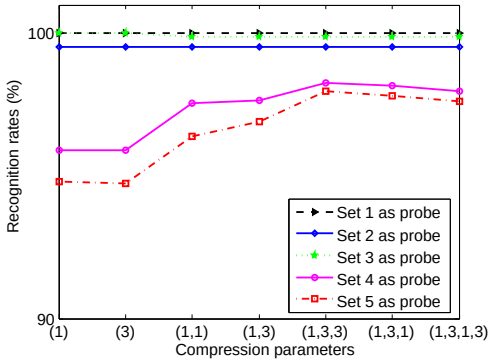
Nous considérons tout d'abord l'influence des paramètres du filtre rétinien. Les paramètres sont le nombre de compressions et les écart-types associés (σ_1, σ_2 dans le cas de deux compressions), les écart-types σ_P, σ_H et le seuil Th. Nous utilisons la base Yale B pour étudier l'influence de ces paramètres. L'algorithme de reconnaissance est la méthode *Eigenface (Whitened PCA)* associée à la distance cosinus.

3.4.1.1 Nombre de compressions et paramètres

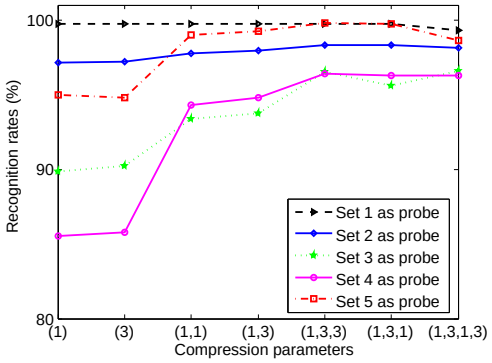
Nous faisons tout d'abord varier le nombre n de compressions utilisées dans la première étape alors que les autres paramètres sont fixés ($\sigma_P = 0.5, \sigma_H = 3.5$, et $Th = 4$). Près de huit cents tests ont été faits :

1. Nous considérons quatre valeurs de n : $n = 1, 2, 3, 4$. Dans la suite, les filtres correspondants sont notés $\mathbf{F}^1, \mathbf{F}^2, \mathbf{F}^3, \mathbf{F}^4$.
2. Pour chaque n, nous faisons varier les écart-types $\sigma = 1, 2, 3$ (qui sont utilisés pour calculer les facteurs d'adaptation). Les filtres correspondants sont notés $\mathbf{F}^n_{(\sigma_1,...,\sigma_n)}$. On considère 12 filtres (3 pour chaque n).
3. Pour chaque filtre, 64 expérimentations différentes ont été réalisées : pour un angle d'illumination donné, les 10 images des 10 personnes dans cet angle sont utilisées comme référence et le reste de la base est utilisé pour le test. Ceci conduit à la réalisation de 64 tests différents (un pour chaque angle d'illumination) pour chaque filtre.
4. Nous calculons ensuite la moyenne des résultats obtenus sur le sous-ensemble à laquelle les images de référence appartiennent.

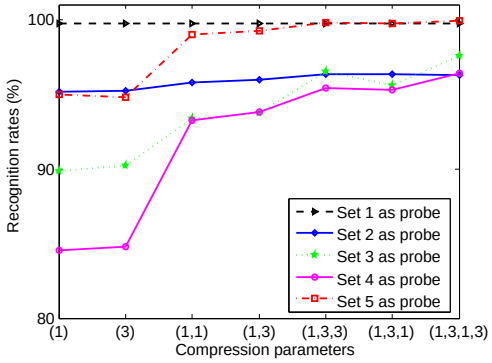
La figure 3.17 montre les taux de reconnaissance moyens obtenus quand les images de référence appartiennent à des sous-ensembles différents avec des filtres différents (nous affichons seulement 7 filtres). Sur les axes horizontaux de la figure 3.17, $(i_1, i_2, ..., i_n)$ signifie $\mathbf{F}^n_{(\sigma_1,...,\sigma_n)}$. Par exemple, (1) correspond au $\mathbf{F}^1$ avec l'écart-type $\sigma_1 = 1$. A partir de cette figure, on peut voir que :



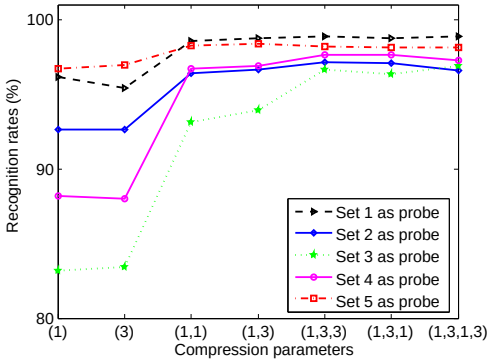
(a) Image in set 1 as reference



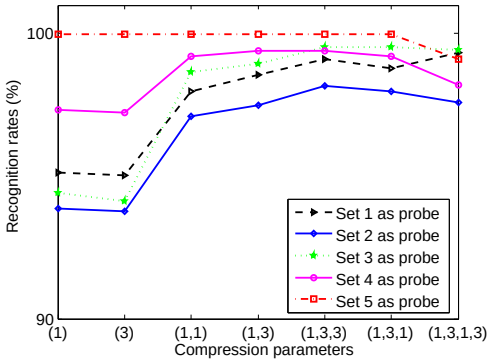
(b) Image in set 2 as reference



(c) Image in set 3 as reference



(d) Image in set 4 as reference



(e) Image in set 5 as reference

Figure 3.17 – Taux de reconnaissance sur la base Yale B pour différents nombres de compressions et écart-types.

1. *Plusieurs* compressions conduisent toujours à de meilleurs taux qu'une *seule* compression.
2. En ce qui concerne les performances en cas de compressions multiples, on observe que tous les $\mathbf{F}^2$, $\mathbf{F}^3$, et $\mathbf{F}^4$ fonctionnent très bien.
3. Les résultats des $\mathbf{F}^n$ $(n = 2, 3, 4)$ sont similaires : *les valeurs de* σ_i *ne sont pas importantes*. En réalité les résultats finaux de $\mathbf{F}^3$ sont légèrement meilleurs que ceux de $\mathbf{F}^2$ et $\mathbf{F}^4$ avec des différences de 0.2 et 0.3% respectivement. Mais pour des contraintes de complexité, nous utiliserons $\mathbf{F}^2$ avec $\sigma_1 = \sigma_2 = 1$.

Il est clair que la méthode proposée produit de très bons résultats dans tous les cas. Quand les images de référence appartiennent aux quatre premiers sous-ensembles, le sous-ensemble 1 est la requête la plus *facile*. En revanche, quand les images de référence appartiennent au sous-ensemble 5, le sous-ensemble 5 est la requête la plus facile. Ces résultats sont logiques. Cependant, de manière surprenante, le sous-ensemble 5 est souvent plus facile à traiter que les sous-ensembles 2, 3, 4 (voir figures 3.17(b),(c) et (d)).

3.4.1.2 Paramètres de DoG et de la troncature

Les paramètres du filtre *DoG* sont les deux écart-types σ_{Ph} et σ_H. Le filtre *DoG* agit comme un filtre passe-bande, σ_{Ph} et σ_H définissent donc les fréquences de coupure basse et haute, respectivement (voir figure 3.13). Une contrainte cruciale est $\sigma_{Ph} < \sigma_H$.

D'un autre côté, les fréquences de coupure devraient dépendre de la qualité des images. Avec une image floue dont les informations se situent principalement en basse fréquence, un filtrage avec σ_{Ph} *trop élévée* va provoquer une suppression des informations utiles. Pour que les choix de ces paramètres soient complètement automatiques, une métrique sur la qualité d'images devrait être utilisée. Si une métrique de qualité d'image est disponible, on peut régler ces paramètres pour que les informations utiles soient bien conservées.

Cependant, par manque de temps, le travail de cette thèse n'adresse pas ce problème. Par conséquent, on choisit les valeurs $\sigma_{Ph} \in \{0.5; 1\}$, $\sigma_H \in \{3; 4\}$. En variant ces valeurs dans ces intervalles, nous trouvons que $\sigma_{Ph} = 0.5$ et $\sigma_H = 3$ donnent des résultats légèrement meilleurs que les autres. Le tableau 3.1 montre les taux moyens obtenus quand les images de référence appartiennent au sous-ensemble 3.

En ce qui concerne le seuil de troncature Th, nous analysons tout d'abord la distribution des valeurs des images I_{nor} (équation 3.3.9). Rappelons que la moyenne de ces valeurs est 0 et que ces valeurs se répartissent principalement autour de 0. Nous choisissons de façon aléatoire 20 images I_{nor} et trouvons qu'avec un seuil $Th \in \{3, 4\}$, on peut supprimer en moyenne 3–4% des valeurs extrêmes dans chaque image. Ensuite, nous

σ_{Ph}, σ_H	Sous-ensembles				
	1	2	3	4	5
0.5 & 3	100	98.3	99.8	98.6	100
0.5 & 3.5	100	98.3	99.4	97.9	100
1 & 3.5	100	98.3	99.0	96.8	99.7
1 & 4	99.8	97.9	96.4	95.7	99.3

Tableau 3.1 – Taux de reconnaissance sur la base Yale B pour différentes valeurs des paramètres σ_{Ph} et σ_H.

évaluons les effets du filtre rétinien en variant Th dans cet intervalle et nous observons que les résultats obtenus sont presque similaires. En revanche, si la troncature n'est pas appliquée, les performances de la reconnaissance sont dégradées d'environ 1–2%, selon les sous-ensembles.

3.4.1.3 Conclusion

Dans toute la suite du travail nous utiliserons les paramètres suivants : $\sigma_1 = \sigma_2 = 1, \sigma_{Ph} = 0.5, \sigma_H = 3$, et $Th = 3.5$.

3.4.2 Résultats expérimentaux sur la base Yale B

Dans la suite, nous comparons les performances de notre méthode avec celles des méthodes de l'état de l'art. Grâce au logiciel "INface tool" [4], nous avons les codes en Matlab de plusieurs méthodes de normalisation d'illumination. Parmi les méthodes disponibles, nous considérons le MSR et le SQI. Nous incluons également les résultats des méthodes qui sont plus représentatives (le code de la méthode PS se trouve sur le site de l'auteur [5]). Nous utilisons comme paramètres de ces méthodes ceux recommandés par les auteurs.

Le tableau 3.2 présente les résultats obtenus sur la base Yale B en utilisant les 10 images d'angle 0° (éclairage de face) comme images de référence et le reste de la base comme images de test. Même si c'est le cas le plus simple, ces résultats sont intéressants puisqu'en général les autres chercheurs ne présentent que les résultats dans ce cas-là. Il est clair à partir du tableau 3.2 que :

1. Quand aucun prétraitement n'est utilisé, les performances de reconnaissance chutent de manière très importante sur les sous-ensembles 4 et 5. Ceci prouve l'importance des méthodes de normalisation d'illumination.

4. http ://uni-lj.academia.edu/VitomirStruc
5. http ://parnec.nuaa.edu.cn/xtan/

Méthodes	Sous-ensembles				
	1	2	3	4	5
Sans prétraitement	100	98.3	64.2	32.9	13.7
Égalisation d'histogramme (HE)	100	98.3	65.8	35	32.6
MSR	100	100	96.7	85	72.1
SQI [131]	100	100	98.3	88.5	79.5
PS [122]	100	100	98.4	97.9	96.7
LTV [25]$^+$	100	100	100	100	100
Filtre rétinien	**100**	**100**	**100**	**100**	**100**
Cone-cast [38]*	100	100	100	100	-
Harmonic image *	100	100	99.7	96.9	-
Gradient angle *	100	100	100	98.6	-

$^+$ *Cette méthode est 500 fois plus lente que notre méthode.*
** Ces méthodes appartiennent à la deuxième catégorie qui vise à modéliser l'illumination. Les auteurs n'ont pas évalué les méthodes sur le sous-ensemble 5.*

Tableau 3.2 – Taux de reconnaissance de différentes méthodes de prétraitement sur la base Yale B en utilisant les images à éclairement idéal comme référence.

2. La méthode proposée a de très bonnes performances : des taux parfaits sont obtenus sur tous les sous-ensembles (bien que cette base est petite, elle est la plus utilisée pour évaluer la robustesse aux variations d'illumination de différentes méthodes).

3.4.3 Résultats expérimentaux sur la base *Extended* Yale B

Nous considérons maintenant les résultats sur la base *Extended* Yale B qui contient plus d'individus. Nous utilisons 38 images de même condition d'illumination dans le sous-ensemble 1 comme référence et le reste de la base comme test. Les résultats présentés dans le tableau 3.3 montrent que la méthode proposée surpasse toutes les méthodes considérées.

Ensuite, nous faisons 30 tests différents sur cette base. Pour chaque expérimentation, la base de référence contient les 38 images des 38 personnes pour un angle d'illumination donné et la base de test contient tout le reste (en réalité, il y a au total 64 tests différents mais on choisit de manière aléatoire 30 conditions d'éclairage différentes : 5 conditions pour chacun des quatre premiers sous-ensembles et 10 conditions pour le sous-ensemble 5). Puis, nous calculons la moyenne des taux de reconnaissance. On observe toujours à partir du tableau 3.4 que la méthode proposée fonctionne très bien même si les variations d'illumination sont présentes sur les images de référence et sur les images requêtes.

Méthodes	Sous-ensembles				
	1	2	3	4	5
HE	98.9	97.6	56.5	23.6	21.4
MSR	100	100	96.7	79.5	65.7
SQI	100	99.8	94.0	85.5	77.0
PS	100	99.8	99.3	99.0	96.6
Filtre rétinien	**100**	**100**	**99.7**	**99.3**	**98.8**

Tableau 3.3 – Taux de reconnaissance de différentes méthodes de prétraitement sur la base *Extended* Yale B en utilisant les images à éclairement idéal comme référence.

Méthode	MSR	SQI	PS	Proposée
Taux	75.5	81.6	97.8	99.1

Tableau 3.4 – Taux moyens de reconnaissance de différentes méthodes de prétraitement sur la base *Extended* Yale B.

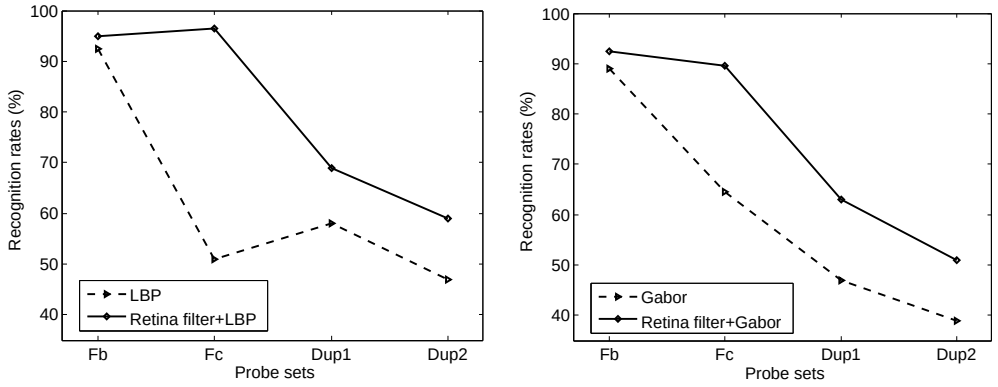
3.4.4 Résultats expérimentaux sur la base FERET

Dans cette partie, on va prouver que la méthode proposée présente les avantages suivants :

1. Elle permet d'améliorer les taux de reconnaissance de méthodes basées sur des caractéristiques qui sont considérées a priori comme invariantes au changement d'illumination.

2. Elle permet en fait également d'améliorer les performances de la reconnaissance de visages dans tous les cas qu'il y ait ou non des variations d'illumination.

Pour ce faire, on choisit deux méthodes de reconnaissance de visages, l'une basée sur le LBP et l'autre basée sur les ondelettes de Gabor car ces deux indices faciaux sont considérées comme a priori invariants aux changements d'illumination. Pour la méthode basée sur le LBP, nous appliquons la méthode décrite dans l'article [2]. Pour la méthode basée sur les ondelettes de Gabor, à partir d'une image en entrée, nous calculons les 40 images de convolution avec 40 noyaux différents (seules les réponses en amplitude sont utilisées). Avant de concaténater toutes ces 40 images dans une seule image, nous utilisons la technique *down-sampling* pour réduire la dimension (le facteur de down-sampling est 8). Cette image est stockée sous la forme d'un vecteur. La distance cosinus entre deux vecteurs est utilisée pour mesurer la similarité entre les deux images originales. On suit le protocole standard associé à la base FERET pour rapporter les performances : le sous-ensemble Fa est la base de référence alors que les sous-ensembles Fb, Fc, Dup1 & Dup2 sont les requêtes.



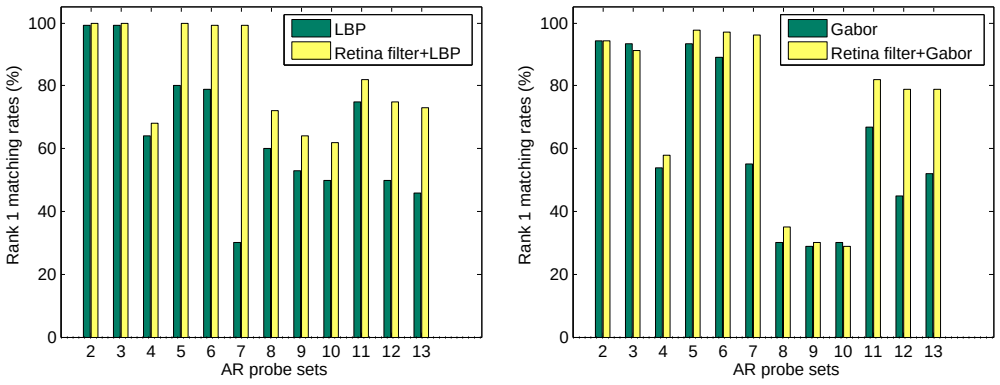
(a) (b)

Figure 3.18 – Performances du filtre rétinien sur la base FERET en utilisant différentes méthodes de reconnaissance : (a) LBP ; (b) : Gabor

Il est évident sur les courbes de la figure 3.18 que le prétraitement par filtre rétinien améliore significativement les performances des deux méthodes considérées sur tous les quatre sous-ensembles de requête. Les améliorations importantes obtenues sur le sous-ensemble Fc confirment que si une bonne méthode de normalisation d'illumination est appliquée au préalable, le robustesse des caractéristiques faciales sera augmentée même si ces caractéristiques sont considérées a priori comme invariantes au changement d'illumination. En ce qui concerne les résultats obtenus sur les sous-ensembles Fb, Dup1, & Dup2, on constate que le prétraitement proposé permet d'améliorer les performances de la reconnaissance de visages dans tous les cas qu'il y ait ou non des variations d'illumination. La raison est que notre filtre ne se contente pas seulement de supprimer les variations d'illumination mais il renforce les contours d'image qui sont des indices importants pour distinguer les individus.

3.4.5 Résultats expérimentaux sur la base AR

Afin de renforcer l'étude des performances de la méthode de *prétraitement* proposée, nous répétons les même expérimentations sur la base AR. Nous utilisons toutes les 126 images "AR-01" (une pour chaque individu) comme référence. Les résultats de reconnaissance sont évalués sur 12 ensembles de requête, de "AR02" à "AR13" et sont présentés sur la figure 3.19. A partir de cette figure, on peut voir que la méthode proposée conduit toujours à de très bons résultats. Les résultats les plus intéressants et surprenants sont ceux sur le test "AR07" (voir Figure 3.20). Dans ce test, les deux méthodes basées sur le LBP et les filtres de Gabor conduisent à de mauvaises performances. Quand les images sont traitées par notre algorithme au préalable à la reconnaissance, les taux de reconnaissance augmentent de manière impressionnante. Ceci prouve encore les performances excellentes de notre méthode. La figure 3.20 montre les images de cet ensemble et celles



(a) (b)

Figure 3.19 – Performances du filtre rétinien sur la base AR en utilisant de différentes méthodes de reconnaissance : (a) LBP ; (b) : Gabor

traitées par notre filtrage rétinien.

A partir de cette figure, nous observons que les variations d'illumination des images sont très bien normalisées par le filtre rétinien. Voilà pourquoi on a obtenu des améliorations très impressionnantes sur cet ensemble requête.

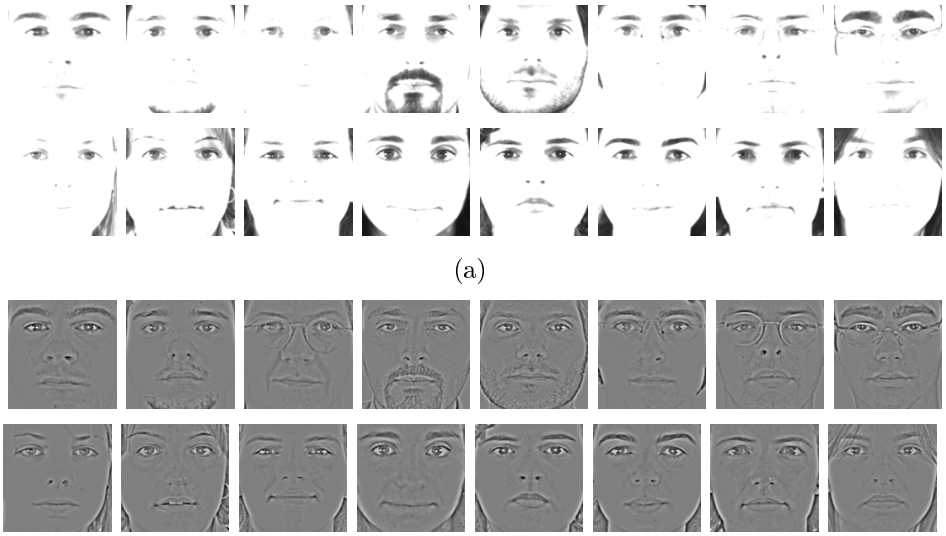
3.4.6 Temps de calcul

Dans cette partie, nous étudions le temps de calcul de différentes méthodes de normalisation d'illumination. Les méthodes considérées sont mises en oeuvre en Matlab. Donc, les temps de traitement ne sont pas optimaux mais ils nous permettent de faire une comparaison entre les diverses méthodes. Pour une estimation plus fiable, c'est le temps pour traiter 10000 images de taille 192x168 pixels qui est considéré (1000 images de la base *Extended Yale B* sont utilisées et on fait les tests 10 fois). Ensuite, un temps moyen par image est calculé [6]. Nous essayons d'optimiser au mieux les codes de chacune des méthodes considérées [7].

Les temps de calcul sont affichés dans le tableau 3.5 ou plus visuellement sur la figure 3.21. Notre méthode est de complexité très faible, on peut traiter environ 65 images de

6. La machine utilisée est un PC Duo 2.4 GHz, 2Gb Ram.

7. Il convient de noter que l'étape la plus longue dans notre méthode sont les convolutions. Soient mn les dimensions d'image, w^2 la taille du masque. En pratique, pour réduire le temps de calcul, au lieu d'utiliser directement un masque 2D, qui conduit à une complexité de $O(mn \times w^2)$, on fait deux convolutions 1D successives, ceci conduit à une complexité de $O(mn \times 2w)$. Dans le modèle, $w = 3\sigma$ où σ est l'écart-type du filtre Gaussien. Comme nous utilisons des écart-types petits ($\sigma_1 = \sigma_2 = 1, ...$), le temps de calcul des convolutions n'est pas important. A l'opposée, pour que les performances des méthodes MSR et SQI soient bonnes, les écart-types utilisés devront être beaucoup plus grands (par exemple, $\sigma_1 = 7, \sigma_2 = 13, \sigma_3 = 20$).Voilà pourquoi notre méthode est très rapide.



(a)

(b)

Figure 3.20 – Images de l'ensemble AR-07 : (a) images originales ; (b) images traitées

Méthodes	LTV$^+$	SQI	MSR	PS	Proposée
Temps (second)	7.3$^+$	1.703	0.126	0.0245	0.0156

$^+$ *Comme le code de cette méthode n'est pas disponible, nous n'avons pas pu mesurer ce temps mais dans [122] les auteurs ont montré que la méthode LTV est environ 300 fois plus lente que la méthode PS, on estime alors que la méthode LTV est environ 500 fois plus longue que notre méthode.*

Tableau 3.5 – Temps de calcul de différentes méthodes sur une image de taille 192x168 pixels.

taille 192x168 pixels pendant chaque seconde. La méthode PS est d'environ 1.57 fois plus longue que notre méthode alors que les autres méthodes sont beaucoup plus lentes que notre méthode.

3.4.7 Normalisation d'illumination pour la détection de visages

Il est clair que les variations d'illumination affectent également les performances des algorithmes de détection de visages. Bien que cette thèse ne se concentre pas sur la détection de visages, nous discutons brièvement comment on peut utiliser la méthode présentée pour améliorer les performances de la détection de visage. En regardant les effets de chaque étape du modèle proposé, nous observons que l'utilisation de l'image en sortie du filtre d'adaptation à la lumière peut améliorer la détection. Pour s'en convaincre, nous calculons les taux de détection de visages sur 640 images originales de la base Yale B en utilisant le détecteur de visage proposé dans [37] associé à différentes méthodes de normalisation d'illumination. La figure 3.22(b) montre un exemple de détection cor-

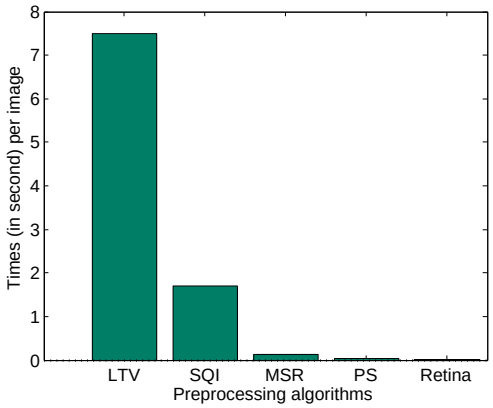


Figure 3.21 – Temps de calcul moyen de différentes méthodes sur une image de taille 192x168 pixels.

(a) Sans prétraitement, le détecteur ne fonctionne pas

(b) Avec prétraitement, le détecteur fonctionne de manière fiable

Figure 3.22 – Illustration de performance de l'algorithme proposé pour la détection de visages.

rect alors que le tableau 3.6 permet une comparaison des performances des différentes méthodes de normalisation considérées. Le tableau 3.6 montre que la méthode proposée produit les meilleurs résultats.

3.5 Conclusions

Dans ce chapitre, nous avons tout d'abord discuté de l'effet des variations d'illumination sur les performances des méthodes de reconnaissance de visages et nous avons vu que celles-ci chutent de manière importante lorsque l'éclairage varie. Nous sommes convaincus par la capacité d'adaptation à l'éclairement du système visuel humain. Cependant, contrairement à beaucoup de méthodes existantes qui sont basées sur la théorie rétinex et qui considèrent que la rétine et le cerveau concourent ensembles au processus de norma-

Méthodes	Sans prétraitement	HE	MSR	Doubles compressions
Taux (%)	12	98	99.0	99.5

Tableau 3.6 – Taux de détection de visage en fonction de la méthode de normalisation sur les images originales de la base Yale B.

lisation d'illumination, nous proposons une méthode qui s'appuie sur les propriétés de la rétine uniquement. Notre motivation réside dans le fait que nous croyons dans la capacité naturelle de la rétine à voir des objets dans des conditions d'illumination différentes alors que le rôle du cerveau est plutôt de reconstruire une représentation visuelle des détails. En d'autre termes, au lieu d'estimer l'illumination comme dans d'autres méthodes, nous essayons de la supprimer directement sur l'image. C'est la première fois qu'un filtre basé sur le comportement de la rétine est utilisé comme normalisation d'illumination pour la reconnaissance de visages. Par rapport au filtre rétinien développé antérieurement au Gipsa-lab, nous avons apporté des modifications qui ont rendu le filtre plus adapté à notre contexte d'application. D'un point de vue théorique, nous avons prouvé que notre algorithme a deux avantages importants :

1. Il supprime les variations d'illumination et le bruit de l'image.
2. Il renforce les contours en conservant les contrastes.

Grâce à de nombreuses expérimentations rigoureuses, nous avons tout d'abord déterminé les paramètres optimaux de notre filtre et puis nous avons prouvé son efficacité. Effectivement, notre méthode de prétraitement :

1. fonctionne de manière très efficace en toutes conditions d'illumination, même s'il y a des variations d'illumination sur les images de référence ou sur les images requêtes ;
2. donne de meilleurs résultats en comparaison avec les méthodes de l'état de l'art ;
3. est plus rapide que toutes les méthodes considérées. Notre algorithme fonctionne en temps réel ;
4. permet d'améliorer les performances des méthodes de reconnaissance qui utilisent des descripteurs de visages considérés comme a priori invariants aux changements d'illumination ;
5. permet d'améliorer les performances de la reconnaissance de visages en général, même s'il n'y a pas de variations d'illumination sur l'image ;
6. peut être utilisée pour améliorer les performances d'algorithmes de détection de visages.

Une nouvelle méthode de prétraitement très efficace viens d'être présentée, nous allons voir dans le chapitre suivant l'étape suivant dans le schema de reconnaissance de visages : extraction des caractéristiques. Il également convient d'avertir au lecteur que *le reste de cette thèse sera écrite en anglais.*

Chapitre 4

Patterns of Oriented Egde Magnitudes : a novel efficient facial descriptor

Once the face image is preprocessed, by using the retina filtering presented in Chapter 3, the next important step in face recognition system is the feature extraction (see Figure 4.1). This involves characterizing/describing the face area by a vector which refers to face representation/descriptor. A good face representation is one which minimizes intra-person dissimilarities whilst enlarging the margin between different people. This is a critical issue, as variations of pose, age and expression can be larger than variations of identity. For real-world face recognition systems, such as surveillance applications, a good representation should also be both fast and compact : if one is testing a probe face against a large database of desirable (or undesirable) target faces, the extraction and storage of the face representation have to be fast enough for any results to be delivered to the end user in good time. Unfortunately, to the best of our knowledge, any existing face description algorithms balance these criteria : the features which can produce high-quality recognition results are computationally intensive in both terms of the stockage requirement and the computational time, whereas low complexity algorithms do not perform reliably enough. It is both a challenge and our motivation to find a descriptor satisfying three criteria, namely the *distinctiveness*, the *robustness*, and the *computationally inexpensive cost*. In this chapter, we propose novel features so-called Patterns of Oriented Edge Magnitudes (POEM) for robust face recognition, a descriptor which we argue satisfies these criteria.

In the rest of this chapter, we first discuss related work in Section 4.1. The POEM feature extraction is described in Section 4.2. Section 4.3 details the use of POEM features for face recognition. Experimental results are presented in Section 4.4 and conclusions are given in Section 4.5.

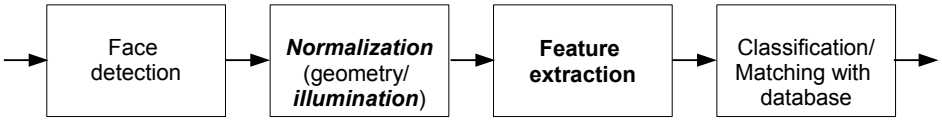


Figure 4.1 – General pipeline of face recognition. ***Bold italic face*** indicates problems already solved ; **Bold-face** indicates problem addressed in this chapter.

4.1 Related literature

In this section, we first present the state-of-the-art algorithms for face representation and explain why most of them are not suitable for our context, surveillance applications where there is crucial constraint upon the processing time. Historically, face recognition algorithms have been developed using classic test sets such as the FERET, AR databases involving high quality images taken under controlled conditions. However, recently large databases of natural images such as LFW have come to prominence, gathered from the Internet using a standard face detector. These latter collections of images are more challenging to vision researchers as they include greater variations in terms of lighting, pose, age, and indeed image quality (unconstrained images of celebrities downloaded from the site *Yahoo News*). In order to show "how challenging yet interesting" this dataset is, **the first** *up-to-date* survey on the systems applied on this set will be given in the second part of this section. By this study, we will point out that even with these very recent methods, we can not find an algorithm which is suitable for our context.

4.1.1 Face representation

Face recognition is an established field with a long history. As one of its crucial issues, face representation has also attracted much work. Face representations are mainly divided into two classes : subspace based holistic features and local appearance features. Heisele *et al.* [46] compare local and global features and observe that local representations outperform global representations for recognition rates larger than 60%. Due to increasing interest, in recent surveys stand-alone sections have been specifically devoted to local features. In Chapter 2, we have already highlighted the advantages of the local approaches and pointed out that they are suitable for our context. Also, we have presented the way of extracting two of the most successful single local face representations, Gabor wavelets and LBP features (see Section 2.2).

In our consideration of face representations, we make a distinction between *elementary* descriptors which returns the feature vector using only the image itself and those requiring a training set for learning the descriptors, which we refer to *learned* descriptors. Learning techniques range from the classical dimensionality reduction algorithms such as

PCA, LDA or feature selection methods such as Adaboost to the more recently proposed ones such as *Bag of Feature* [92]. It is widely known that learning techniques enhance the descriptor performance (boosting the performance or at least speeding up the process). However, it is not really fair to compare directly these two types of descriptors because *elementary* descriptors capture the *primitive* feature of an object which can be useful for tasks of detection or registration.

Gabor features, which are spatially localized and selective to spatial orientations and scales, are analogous to the receptive fields of simple cells in the mammalian visual cortex [73]. Due to their robustness to local distortions, Gabor features (or variants, such as truncated Gabor filters or Gabor *jets*) have been successfully applied to face recognition [144, 141, 73, 146, 137, 102, 103]. Indeed, the FERET evaluation and the FVC2004 contests have seen top performance from methods based upon Gabor features. Typically, Gabor features are calculated by convolving images with a family of Gabor kernels at different scales and orientations, which is a costly stage. For example, applying 40 convolutions which are widely used in Gabor-based representations on a 96×96 image would result in a vector of $40 \times 96 \times 96 = 368640$ features. To address the problem of high dimension, dimensionality reduction techniques can be employed, from the simple *downsampling* [73] or Kernel PCA techniques [71] to automatic feature selection mechanism for selecting only the important features such as Adaboost [117]. Recently, Choi *et al.* [27] attempt to speed up the Gabor feature extraction process by proposing the Simplified Gabor wavelets ; Mellakh [79] uses 32 convolutions for Gabor-based representation extraction. Despite these attempts, the computational cost of these features is still prohibitive for real-time applications.

More recently, the spatial histogram model of LBP has been proposed to represent visual objects, and successfully applied to texture analysis [93], human detection [87] and face recognition [2]. LBP is basically a fine-scale descriptor which captures small texture details, in contrast to Gabor features which encode facial shape and appearance over a coarser range of scales. The process of extracting LBP features is much faster than that of extracting Gabor-based features. Using LBP, Ahonen *et al.* [2] have reported impressive results on the FERET database. Some boosting LBP algorithms have been proposed, such as [142].

Other local descriptors, e.g. SIFT and its dense version HOG, have been commonly employed for many real-world applications because they can be computed efficiently, are resistant to partial occlusion, and are relatively insensitive to changes in viewpoint. Although SIFT and HOG have been widely accepted as two of the best features to capture edge or local shape information, they have not seen much use in face recognition. Bicego *et al.* [15], Rosenberger and Brun [107] evaluated the use of SIFT for face authentification and achieved strong results on the BANCA and AR databases respectively, but no further results on bigger databases were given. It was shown in [80] that HOG performs worse

than LBP and Gabor filters on the FERET database.

Effectively, current state-of-the-art face representation algorithms are those combining both LBP and Gabor features. The motivation for this combination is that the Gabor and LBP features have correlated properties which can be easily incorporated : LBP features capture fine-scale details whereas Gabor filters capture shape and appearance over coarser scales. Zhang *et al.* [144] introduced a combination approach extending LBP to LGBP (Local Gabor Binary Pattern) by applying multi-orientation and multi-scale Gabor filtering as a preprocessing step of LBP. They first calculated the 40 Gabor magnitude images by convolving the original image with 40 Gabor kernels and then applied the LBP method upon these resulting magnitude images. This additional stage greatly improves performance when compared with the pure LBP algorithm. In a similar vein, they proposed HGPP (Histogram of Gabor Phase Pattern) [141] combining the spatial histogram and the Gabor phase information encoding scheme. Unlike the LGBP method encoding the Gabor magnitude images, the HGPP algorithm encodes both real and imaginary images. Global Gabor phase pattern (GGPP) and local Gabor phase pattern (LGPP) are proposed to encode the phase variations : GGPP captures the variations derived from the orientation changing of Gabor wavelets at a given scale, while LGPP encodes the local neighborhood variations. They also used 40 Gabor kernels of 5 different scales, resulting in 5 real GGPPs, 5 imaginary GGPPs (GGPP encode), 40 real LGPPs, and 40 imaginary LGPPs. This means that HGPP has to encode 90 *images* of the same size as the original one.

In [90], Nguyen *et al.* applied the Whitened PCA technique upon LGBP and showed very high recognition rates on the FERET database. To the best of our knowledge, the HGPP descriptor has not been re-evaluated by researchers other than the original authors, although it has been presented for several years. This is probably due to its dramatically high complexity in terms of both time and memory.

Zou *et al.* in [146] compared LBP, Gabor and PCA features using a subset of face regions (eyes, nose, mouth) extracted through manual labelling, and show that in this context Gabor features perform best on the FERET and AR datasets. Gabor features were combined using the Borda Count method. Another fusion method proposed by Tan and Triggs [137] uses Kernel PCA to reduce first the dimension of the LBP and Gabor features, and then applies Kernel DCV (Discriminant Common Vector) to combine them. They also present impressive results on the FERET database (similar to KPCA Gabor in [71] or WPCA LGBP in [90], this is a *learned* representation).

The terms *combination* and *fusion* have slightly different meanings : combination approaches refer to those which try to incorporate the advantages of stand-alone features into an unique one, such as LGBP and HGPP, while fusion methods aim at fusing the output of some features, as in [137].

Many representation approaches have been presented in the literature. Unfortunately, the high-performing algorithms, such as HGPP and LGBP, are computationally intensive and are therefore impractical for real-time applications.

4.1.2 Algorithms applied on LFW dataset

We now turn to techniques applied on the more challenging Labeled Faces in the Wild dataset for unconstrained face recognition. Within this dataset and methodology, the task of face recognition becomes face verification. This typically consists of representation in some kind of feature space (for example, by using local descriptors to create a histogram of oriented edge values), and of classification as *same face* or *different face* (for example, using an SVM).

In our consideration of work on the LFW dataset, we make a distinction between systems which follow this pipeline strictly using a pair matching protocol, and those which use additional information when making the decision. This additional input ranges from systems which use human annotation such as [146] to systems which incorporate additional negative examples such as [135]. Our discussion here of prior work is therefore divided into two subsections : methods which are learned from training set of labelled image pairs (same/not same) and tested in a pair decision (same/not same) context, and those which use additional information (such as identity, additional databases, or multiple images of each face) and which we call *extended* methods.

Wolf *et al.* in [135] provide results for four different descriptor-based techniques in addition to an extended one-shot learning approach which will be described in the following section. The descriptors they cover are LBP, Gabor filters, and a novel Patch based LBP in which the similarities between neighboring patches are used to encode local information. The same/not-same decision is made by training a linear SVM. They also present combined results which are obtained by concatenating all of the descriptors, and training an SVM on the resultant 16 element vectors. Guillaumin *et al.* [42] use a similar approach concatenating several different descriptors (LBP, three-and four-patch LBP and also SIFT) and then applying a metric-learning approach.

The randomized tree approach of Nowak [91] has been applied to LFW by Huang *et al.* [50]. This involves learning randomized trees from simple image features (SIFT & geometric features), optimizing the trees to generate same/not-same distinctions. This has also been combined with the MERL classifier (which was presented by Jones and Viola and then called by the laboratory's name [56]), also described in [50] (as these classifiers appear to capture different qualities of the face, a simple average of the two outputs improves results).

Pinto *et al.* in [102, 103] use *simple* features in combination with SVMs and later

Multiple Kernel Learning SVMs (MKL-SVMs) and present good results on LFW. Their feature sets include pixels, V1-like features made by truncating Gabor wavelets, and what they call V1-like+, which consist of the V1-like features concatenated with various ad-hoc features such as image histograms and a scaled down version of the original image. They do not discuss computational cost or runtime, but as they use many Gabor filters (around 1000), this suggests us that Pinto *et al.*'s technique would be too slow to be applicable to surveillance applications.

Multi-region probabilistic histograms (MRH) are introduced by Sanderson and Lovell in [114]. This technique is inspired by *Bag-of-Word* models (such as [92]) in which a training set is used to cluster features into a dictionary of *visual words*, and during subsequent processing, the closest visual words are used rather than the input features themselves. In [114] the features are extracted by dividing the image into small patches, and using DCT decomposition. Within the training set, these features are then clustered using a Gaussian mixture model (GMM) to create a dictionary of visual words. In testing, the face image is divided into regions and each region is represented by a probabilistic histogram, representing the probability that each of the visual words is present in that region. By using fairly large regions (just 9 per face) and probabilistically modeling the presence or absence of a particular visual word, this method achieves a certain amount of robustness to noise and misalignment. The authors also present a method with improved results by using a normalized distance which relies upon the existence of a cohort of *negative faces* ; according to our categorization this variant is an extended method.

Also related to the *Bag-of-Word* technique is the very recent article published last June by Cao *et al.* [21]. In this technique pixel-level sampling in ring patterns is converted into a descriptor by first clustering using PCA-tree techniques, then dimensionality is reduced with a joint PCA-normalization step. The authors argue that this method learns descriptors (they call the descriptor the LE descriptor) optimized for face recognition, and their results are very impressive. They couple this descriptor with a pose-estimation technique and facial feature-level matching and gain some of the best results to date on the LFW dataset. This work is in a similar vein to recent work from Winder, Brown and Hua on the learning of more general descriptors (SIFT/HOG style representations of images) as outlined in [133, 132, 19].

In this work, the PCA-normalization step plays a very important role in boosting the descriptor performance with an improvement of 6% points on the LFW set. Whilst the authors argue this gain *surprising*, we wish to find the *explanations*. The reason is quite simple : once two PCA-based dimension-reduced vectors are normalized to unit-length vectors, the $L2$ distance between them (in the LE method, the authors use $L2$ distance) is exactly equal to the cosine distance between the two dimension-reduced vectors without normalization. And in the prior work carried out in 2004 [98], Perlibakas has empirically shown that cosine distance is much better than Euclidean ($L2$) distance for PCA-based

face recognition. This observation was reinforced in [32, 90]. The LE algorithm belongs to *learned* descriptors, according to our categorization.

4.1.2.1 Extended methods

Wolf *et al.* [135] present an one-shot-learning method in which they use Linear Discriminant Analysis to learn a same/not-same model for each of the images in a test face pair. This relies upon a large set of images of people who are definitely not either of the test faces ; and is achieved by using one of the 9 LFW training splits as negative examples. (The structure of the LFW dataset ensures there is no overlap in identity between the 10 splits.) The use of one-shot learning in this way improves the results remarkably. However, in real-life applications, it may be difficult to obtain a set of face images that are definitely *not* going to be in the test domain. In [136], the method is extended to include 2-shot learning and also a ranking of target faces against the background set of negative faces, further improving results. These techniques are effectively learning the relationships between the target faces and the distribution of face images in general. At the time of writing, Wolf *et al.*'s 2009 method [136] presents the best performance on the LFW benchmark. Taigman *et al.* in [120] use multiple one-shots using more than one image per individual (by making use of image labels during training - this is the LFW dataset's image unrestricted training protocol). They also present results using a hybrid descriptor (multiple patch based LBP with SIFT) and one-shot learning in a pair-matching context. Kumar *et al.* in [61] also present excellent results on the LFW dataset, using attribute and simile classifiers. Attribute classifiers are learned from a labeled dataset of faces – not merely labeled with the name of the person (as in LFW), but labeled with attributes such as "Bags under eyes" or "Asian". This stage requires the manual annotation of over 65000 attributes using the Amazon *Mechanical Turk* system. Simile classifiers do not have this costly labelling stage but are learned from automatically extracted facial features (mouth, nose, eyes etc.). Each simile classifier is trained on a particular region with positive examples coming from a set of images of the same person, and negative examples coming from the same face region but from different people. Simile classifiers capture the intuitive idea that whilst we might not be able to describe facial features, we can say whether they are similar (He's got eyes like Brad Pitt, for example). Importantly, the training set for the simile classifiers cannot be in the set of images to be tested. So, this technique relies on a large external database of labeled faces (60 reference people, up to 600 positive face images per reference person, and ten times as many negative images). An SVM is used on the output of the attribute or simile classifiers for the pairs of images in the test set, in order to determine whether the images are from the same person or different people.

4.1.3 Conclusions

This section first presented the state-of-the-art algorithms for face representation and explained why most of them are not suitable for our context. Then, by giving an *up-to-date* survey on the systems applied on the challenging but interesting LFW set, we pointed out that even with these very recent methods, we can not find an algorithm which is suitable for our context. That is our motivation for the proposition of a novel face descriptor in the next section.

4.2 Proposed algorithm : POEM features

The current state-of-the-art face representations are those combining or fusing different single features : the LBP and Gabor features as in the LGBP, HGPP methods ; LBP, Gabor, and SIFT as in [136, 42]. These algorithms try to bring the advantages of different single features : the LBP method is a "micro-pattern" capturing the image details at small scale whereas the Gabor filters are capable to characterize the image details over coarser scales and through different orientations. Whilst the LBP algorithm is a good choice for describing texture information, SIFT and HOG have been widely accepted as the best features to capture edge or local shape information.

In order to build a feature which can inherit these properties without suffering the drawback coming from the Gabor filters, i.e. the computationally expensive cost, we propose to apply the idea of the self-similarity calculation from the LBP-based structure on the distribution of local edge through different orientations. The resulting features are referred to Patterns of Oriented Edge Magnitudes (POEM). This is done as we find that combining both the edge/local shape information and the relation between the information in neighboring regions can better characterize object appearance.

In order to build the POEM features for one pixel, the intensity values in the calculation of conventional LBP are replaced by gradient magnitudes, computed by accumulating a local histogram of gradient directions over all pixels of a spatial patch ("*cell*"). Additionally, these calculations are done across different orientations.

1. We use the terms *cell* and *block*, as in [30], but with a slightly different meaning. Cells (large squares in Figure 4.2) refer to spatial regions around the current pixel where a histogram of orientations is accumulated and assigned to the cell's central pixel. Blocks (circular in Figure 4.2) refer to more extended spatial regions, upon which the LBP operator is applied.
2. Our use of oriented magnitudes is also different from that in [30] where HOG features are computed in dense grids and then is used as the representation of cell. On the contrary, in our POEM algorithm, *for each pixel*, a local histogram of

gradients over all pixels of the cell, centered on the considered pixel, is used as the *representation of that pixel*.

3. The term *pattern* in POEM is not as *local* as in the conventional LBP based methods. LBP methods often calculate the self-similarity within a small neighborhood while the block used in POEM is rather extended (see details in Section 4.2.1).
4. In combination approaches of Gabor and LBP features such as [144], Gabor filters are first used for capturing large scale information and LBP operator is then applied for encoding small details. On the contrary, POEM first characterizes object details at small scale and then uses the LBP based structure to encode information over larger region.

Similar features have seen increasing use over the past decade [60, 74, 30] ; the fundamental idea being to characterize the local object appearance and shape by the distribution of local intensity gradients or edge directions. We further apply the idea of self-similarity calculation from LBP-based structure on these distributions. As can be seen in Figure 4.2, once the gradient image is computed, the next two steps are assigning the cell's accumulated magnitudes to its central pixel, and then calculating the block self-similarities based on the accumulated gradient magnitudes by applying the self-similarity operator.

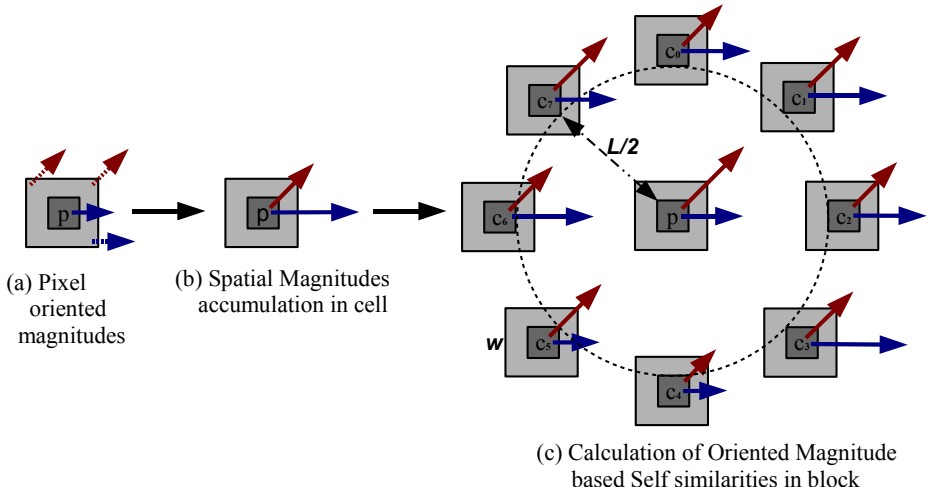


Figure 4.2 – Main steps of POEM feature extraction.

4.2.1 POEM feature extraction in detail

The first step in extracting the POEM feature is the computation of the gradient image. The gradient orientation of each pixel is then evenly discretized over 0-π (*unsigned*

representation) or 0-2π (*signed* representation). Thus, at each pixel, the gradient is a 2D vector with its original magnitude and its discretized direction (the blue continuous arrow emitting from pixel **p** in Figure 4.2(a)).

The second step is to incorporate gradient information from neighbouring pixels (the discontinuous arrows in Figure 4.2(a)) by computing a local histogram of gradient orientations over all cell pixels. At each pixel, the feature is now a vector of m values where m is the number of discretized orientations (i.e. number of bins). Vote weights of each pixel's contribution can either be the gradient magnitude itself, or some functions of the magnitude : we use the gradient magnitude at the pixel, as in [30]. To increase the importance of the central pixel, a weighted window can be used, such as a Gaussian filter or a binomial kernel. However, similar to the CS-LBP method [45], we find this does not improve the discriminative power of the POEM feature. This is, of course, a good news since we can accelerate the process using the integral image trick as used by Viola and Jones [126].

Finally, we encode the accumulated magnitudes using the LBP operator within a block. Remind that the original LBP operator labels the pixels of an image by thresholding the 3×3 neighborhood surrounding the pixel with the intensity value of central pixel, and considering the sequence of 8 resulting bits as a number (as shown in Figure 2.8 – Chapter 2). Only uniform patterns are typically considered to reduce the number of patterns and then accelerate the method (using 8 binary bits for encoding LBP results in 256 LBPs, among these only 58 LBPs are *uniform*).

We apply this calculation on the accumulated gradient magnitudes and across different directions to build the POEM features. Firstly, at the pixel position p, a POEM feature is calculated for each discretized direction θ_i :

$$POEM^{\theta_i}_{L,w,n}(p) = \sum_{j=1}^{n} f(S(I^{\theta_i}_p, I^{\theta_i}_{c_j}))2^j, \tag{4.2.1}$$

where I_p, I_{c_j} are the accumulated gradient magnitudes of central and surrounding pixels p, c_j respectively ; $S(.,.)$ is the similarity function (e.g. the difference of two gradient magnitudes) ; L, w refer to the size of blocks and cells, respectively ; n is number of pixels surrounding the considered pixel p ; and f is defined as :

$$f(x) = \begin{cases} 1 & if \quad x \geq \tau, \\ 0 & if \quad x < \tau, \end{cases} \tag{4.2.2}$$

where the value τ is slightly larger than zero to provide some stability in uniform regions, similar to [135].

The final POEM feature set at each pixel is the concatenation of these unidirectional POEMs at each of our m orientations :

$$POEM_{L,w,n}(p) = \{POEM^{\theta_1}, ..., POEM^{\theta_m}\}, \tag{4.2.3}$$

4.2.2 Properties of POEM feature

We discuss here the interesting properties of this novel feature set for object representation postponing the question of complexity until Section 4.4.4. For each pixel, POEM characterizes not only local object appearance and shape, but also the relationships between this information in neighboring regions. It has the following properties :

1. POEM code is an oriented feature. Since the number of discretized directions can be variable, POEM has the ability to capture image information in any direction and is adaptable for object representation with different levels of orientation accuracy.
2. Computed at different scales of cells and blocks, POEM is also a spatial multi-resolution feature. This enables it to capture both local information and more global structure.
3. Using gradient magnitudes instead of the pixel intensity values for the construction makes POEM robust to lighting variance. In [23, 74], edge magnitudes have been shown to be largely insensitive to lighting.
4. The oriented magnitude based representation contains itself the relation between cell pixels. POEM further calculates dissimilarities between cells and therefore has the ability to capture *multi-scale self-similarities* between image regions. This makes POEM robust to exterior variations, such as local image transformations due to variations of pose, lighting, expression and occlusion that we frequently find when dealing with faces.

Patch-based or multi-block LBPs [135] also consider relationships between regions in a similar way to our POEM features. However the richer information coming from the use of gradients at multiple orientations gives us greater descriptive power, and a greater insensitivity to lighting variations.

4.3 Face recognition based on POEM features

Once every image pixel is characterized with the POEM features, meaning that m POEM images (see Figure 4.3) are obtained from one input image [1], the next stage is to use these feature values in some ways to represent the given face. Since the final step of POEM feature extraction is based on the LBP operator which is widely known as a texture detector (refer to Section 2.2.2), and histogram serves as a good description tool for representing a texture image [2], we exploit the spatial histogram to model the encoded POEM. This algorithm called POEM Histogram Sequence (POEM-HS)

1. Since some pixels at the image border have not enough neighbors, the POEM feature extraction procedure (also the original LBP method) is not carried out. One possible solution is using interpolation. However, this is not really necessary as we can easily crop the facial images. Thus, we carry out here experiments without interpolation; and our POEM images are slightly smaller than the original one.

belongs to the *elementary* descriptor, according to our categorization, and is illustrated in Figure 4.3 and will be discussed in detail in Section 4.3.1. We further propose to apply the PCA technique, followed by a whitening process, on the POEM-HS descriptor in order to obtain a more discriminative and compact descriptor. The obtained representation, a *learned* algorithm, will be detailed in Section 4.3.2. Finally, we will describe some techniques for normalizing the descriptor and for calculating the distance between two feature vectors.

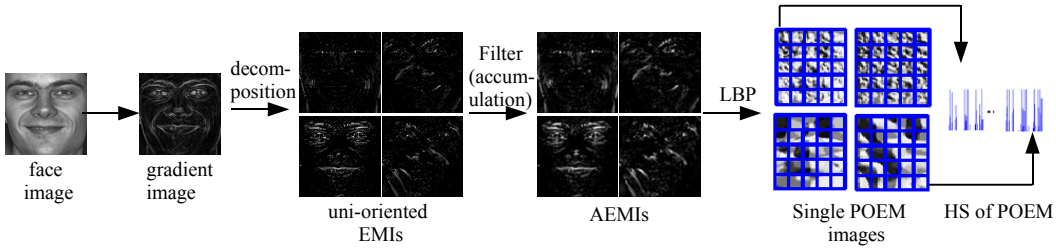


Figure 4.3 – Implementation of POEM histogram sequence for face description.

4.3.1 POEM Histogram Sequence

The Oriented Edge Magnitude Image (oriented EMI) is first calculated from the original input image (Section 4.2.1) and divided into m uni-oriented EMIs through gradient orientations of pixels. Note that the pixel value in uni-oriented EMIs is gradient magnitude. For every pixel on uni-oriented EMIs, its value is then replaced by the sum of all values in the cell, centered on the current pixel. These calculations are *very fast* (using the advantage of integral image [126]). Resulting images are referred to accumulated EMIs (AEMIs). LBP operators are applied on these AEMIs to obtain the POEM images (Figure 4.3). In order to incorporate more spatial information into the final descriptor, the POEM images are spatially divided into multiple non-overlapping regions, and histograms are extracted from each region. Similar to [2, 135], for the goal of descriptor dimension reduction, only *uniform* POEM codes are used, meaning that all non-uniform POEM codes are labeled with a single label, while each uniform POEM code is cast into a unique histogram bin according to its decimal value (for example, when using 8 neighbors, there will be 58 *uniform* codes, the histogram dimension is therefore of 59). Finally, all the histograms estimated from all regions of all POEM images are concatenated into a single histogram sequence (POEM-HS) to represent the given face.

4.3.2 Dimensionality reduction with Whitened PCA

Although the POEM-HS descriptor has significantly speeded up the face recognition process when compared to similar high performing descriptor (see Section 4.4.4), its dimension is still relatively high : $m \times \#uniform\_codes \times \#patches$ where # x extends for the number of x (note that # uniform_codes depends on n, the neighbor number of a cell within a block, see Section 4.4.1.3). In order to address this problem, we apply the PCA dimensionality reduction technique, followed by a whitening process, on POEM-HS. Note that PCA technique is a natural choice for dimensionality reduction in face recognition from one sample. Also, the PCA technique, followed by a whitening process, which is called Whitened PCA, can increase the discriminative power and robustness of feature sets, as proved in [98, 32, 90]. In the rest of this chapter, we will refer the final representation to WPCA-POEM or PCA-POEM.

In order to make the English writing portion of this thesis more self-contained, we succinctly discuss here the advantages of WPCA. The PCA technique has two shortcomings : (1) the leading eigenvectors encode mostly illumination and expression, rather than discriminating information ; and (2) Mean-Square-Error (MSE) principle underlying PCA favors low frequencies and thus loses the discriminating information contained in the high frequency components. The whitening process normalizing the PCA based feature can directly counteract these disadvantages. Specifically, the PCA based feature, $\mathbf{p}$ (which is calculated as in the equation 2.1.1.1 : $\mathbf{p} = \mathbf{U}_{\mathbf{pca}}^{\mathbf{T}}\mathbf{x}$) is subjected to the whitening transformation and yields yet another feature set $\mathbf{w}$:

$$\mathbf{w} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{p} = \mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}_{\mathbf{pca}}\mathbf{x}, \tag{4.3.1}$$

where $\mathbf{\Lambda}^{-\frac{1}{2}} = diag\{\lambda_1^{-\frac{1}{2}}, \lambda_2^{-\frac{1}{2}}, ...\}$ and λ_i is eigenvalue. The integrated projection matrix $\mathbf{\Lambda}^{-\frac{1}{2}}\mathbf{U}_{\mathbf{pca}}$ treats variance along all principal component axes as equally significant by weighting components corresponding to smaller eigenvalues more heavily and is arguably appropriate for discrimination. Consequently, the negative influences of the leading eigenvectors are reduced while the discriminating details encoded in trailing eigenvectors are enhanced.

4.3.3 Descriptor normalization and distance measure

In the pioneer work using histograms of micro patterns (LBP) for face recognition [2], Ahonen *et al.* did not apply any normalization techniques on histogram values and argued that using the χ^2 distance gives better performance than $L2$ distance and *Intersection histogram*. Very recently, Wolf *et al.* [135] proposed two variants of LBP, TPLBP and FPLBP, and applied a *Two-step* normalization technique [2] on histogram values. Also,

2. This technique is originated in SIFT algorithm, which is used for image matching [74].

they reported the recognition rates on the LFW set using $L2$ & Hellinger distances (Hellinger distance is obtained as the $L2$ distance after taking the square root of histogram values).

4.3.3.1 POEM-HS descriptor

With respect to the *elementary* POEM-HS descriptor, we investigate the effect of different normalization methods and different distance metrics by considering the 4 cases :

1. POEM-HS + χ^2
2. POEM-HS + $L2$
3. POEM-HS + Hellinger, this is similar to POEM-HS + *Square root* + $L2$
4. POEM-HS + *Two-step* normalization + $L2$

4.3.3.2 *Learned* WPCA-POEM descriptor

When using the *learned* WPCA-POEM descriptor, we employ the angle-based distance (cosine distance) for calculating the similarity of two representations as Perlibakas [98] has pointed out that it is the best metric distance for Whitened PCA based face recognition. With respect to descriptor normalization methods, taking the square root of histogram values can also be considered as a normalization step. We have therefore 4 normalization scenarios before applying the whitening PCA process :

1. *Original* : any normalization step is employed,
2. *Two-step* : before concatenating all patch histograms to build one complete POEM-HS, patch histogram is normalized to unit length, their values truncated at a threshold, and then once again normalized to unit length as in SIFT,
3. *Square root*,
4. *Two-step* + *Square root*,

Table 4.1 shows all considered test cases.

4.4 Experiments and Discussions

We first present preliminary experiments and details of parameter choice of *elementary* POEM feature using the FERET database, in which we use the nearest neighbor classifier to determine facial identity. Then we present classification results of both POEM-HS and PCA-POEM representations on FERET and LFW showing that our method outperforms the descriptor-based techniques and compares well with much more

Feat. Extraction	Normalization	Dim. reduction	Distance
POEM-HS	No		χ^2
	No		$L2$
	Square root		$L2$
	Two-step		$L2$
POEM-HS	No	WPCA	Cosine
	Square root		
	Two-step		
	Two-step + Square root		

Tableau 4.1 – Different scenarios for evaluating techniques of descriptor normalization and distance measure.

complex systems. According to our categorization, PCA-POEM is a *learned* representation whose performance depends on the dataset used for learning. Therefore, for a fair comparison with other *learned* algorithms, for each considered target database, before reporting the performance of PCA-POEM, several experiments are conducted in order to find the optimal parameters, e.g the feature normalization method and the PCA dimension.

4.4.1 Parameter evaluation

In this section, we study how the parameters of POEM-HS descriptor influence the final performance. Parameters include the number m and the type (unsigned or signed) of orientations, the cell size $(w * w)$, and the block size $(L * L)$, and the neighbor number n of a cell within a block. As for the cell/block geometry, two main geometries exist : rectangular and circular. We use here circular blocks including bilinear interpolation since they provide the relation between equidistant neighboring cells [2]. For a simple implementation, square cells are used, meaning that pixel information is calculated using its neighborhood in a square patch. Throughout the first experiments, we use the default algorithm whose parameters are : neighbor number $n = 8$, original values of histogram without normalization, and χ^2 distance. The simple "nearest neighbor" is used as the classifier.

The experiments checking the effects of parameters are conducted on the FERET face database, following the standard evaluation protocol : Fa is used as gallery, while Fb, Fc, Dup1 & Dup2 are the probe sets. All images are divided into 8x8 non overlapping patches, meaning that there are 64 region histograms of POEM per image.

4.4.1.1 Determining the optimal number of orientations and signed/unsigned representation.

Nearly six hundred cases are considered, recognition rates are calculated on 3000+ FERET face images with different parameters :

1. the block size $L = \{5, 6, 7, 8, 9, 10, 11\}$,
2. the cell size $w = \{3, 4, 5, 6, 7, 8\}$,
3. the number of discretized orientations is $m = \{2, 3, 4, 5, 6, 7\}$ in the case of unsigned representation, and is doubled to $m = \{4, 6, 8, 10, 12, 14\}$ in the case of signed representation.

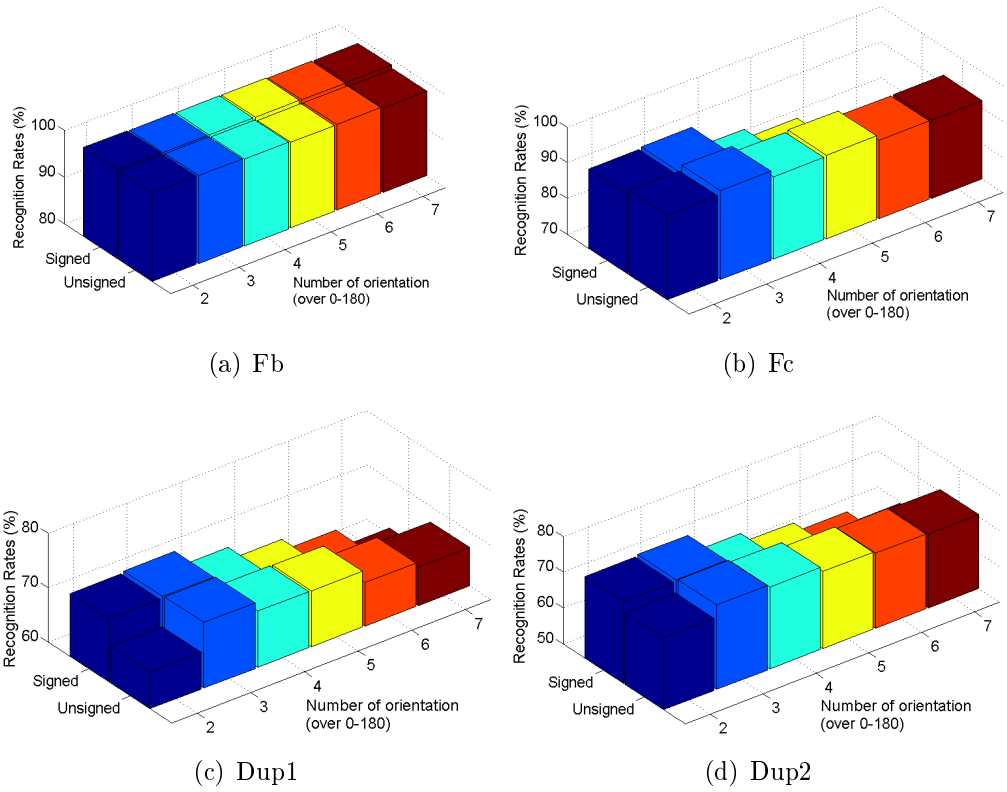
Cells can overlap, notably when blocks are smaller than cells, meaning that each pixel can contribute more than once. For each probe set, the average rates are calculated over different numbers and types of orientation. Figure 4.4 shows the recognition rates obtained on probe sets Fb, Fc, Dup1, and Dup2. The average recognition rate obtained on probe set Fb (shown in Figure 4.4(a)) is around 96.5%, suggesting the number m and signed/unsigned orientations do not affect the recognition performance in the case of expression variations.

Considering the question of using a signed or an unsigned representation, similar to [30] we find that including signed gradients decreases the performance of POEM even when the data dimension is doubled to preserve more original orientation resolution. For face recognition, POEM provides the best performance with *only 3 unsigned bins* (see Figures 4.4(b,c,d). This should be noted as one advantage of POEM since the data dimension for face description is not greatly increased as in LGBP or HGPP [144, 141]. It is clear from Figures 4.4(c,d) that using too many orientations degrades significantly the recognition rates on Dup1 and Dup2 sets. This can be explained by the fact that increasing the number of orientation bins makes POEM more sensitive to wrinkles appearing in face with time.

Summarizing : the number m and signed/unsigned orientations do not affect the recognition performance in the case of expression variations (see results obtained on Fb set – Figure 4.4(a)) ; the unsigned representation is more robust than signed representation, notably to lighting (see results obtained on Fc set – Figure 4.4(b)) ; using few orientations (1, 2) is not enough to represent face information but too many (more than 3) makes POEM sensitive to aging variations (refer to Figures 4.4(c,d)). Thus, ***the best case is 3 unsigned bins***.

4.4.1.2 Determining the optimal cell and block size

Average recognition rates of all 4 probe sets are first calculated with different sizes of cells and blocks with 3 unsigned bins of orientation discretized. As can be seen from



(a) Fb (b) Fc

(c) Dup1 (d) Dup2

Figure 4.4 – Recognition rates obtained with different numbers of orientations on probe sets : (a) Fb, (b) Fc, (c) Dup1, and (d) Dup2. These rates are calculated by averaging recognition rates with different sizes of cell/block.

Figure 4.5, using POEM built on 10x10 pixel blocks with histogram of 7x7 pixel cells provides the best performance. To verify the correctness of these parameters, we further calculate the average rates across cell sizes and across block sizes, meaning that these parameters are now considered independent. Also, in this test, both 10x10 pixel block and 7x7 pixel cell perform the best. This procedure has been repeated with *different numbers of orientation bins*, and the same optimal parameters have been obtained.

The following tests are conducted with ***3 unsigned bins, cell size*** $w = 7$ and ***block size*** $L = 10$.

4.4.1.3 Determining the optimal neighbor number

Using the optimal parameters so far, we calculate the average recognition rates for all 4 probe sets by varying the neighbor number n of a cell from 4 to 12, $n = \{4, 6, 8, 10, 12\}$. For higher confidence, these experiments are also carried out with different distance metrics : χ^2, $L2$ & Hellinger. As can be seen from Figure 4.6, for all three considered distance metrics, the performance changes quite slightly when n ranges from 6 to 12,

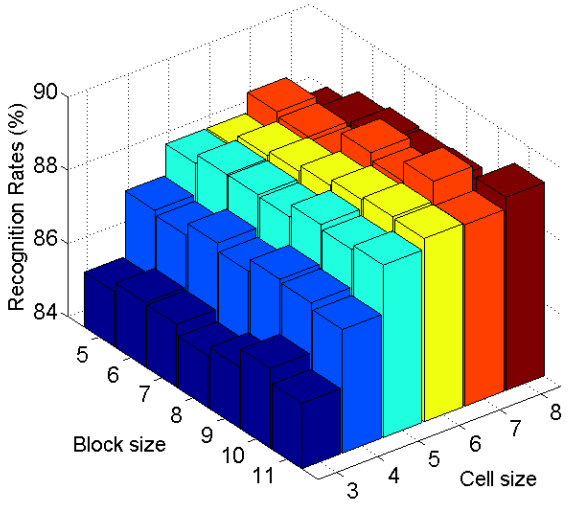


Figure 4.5 – Recognition rates as the cell and block sizes change.

while it drops 2.5% when n decreases to 4. For the goal of speeding up the process without scarifying the performance, ***we therefore choose*** $n = 6$.

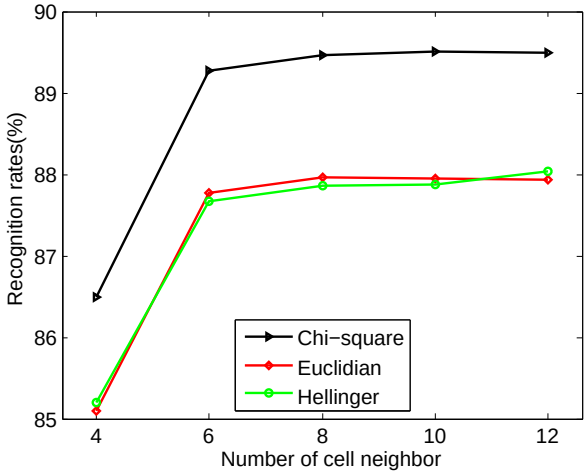


Figure 4.6 – Recognition rates as cell's neighbor number varies.

Conclusion : the optimal parameters of POEM feature for face recognition are : *unsigned representation with 3 bins, built on 10x10 pixel blocks and 7x7 pixel cells*, and there should be *6 neighbors for each considered cell.*

4.4.2 Results on the FERET database

In this section we compare the performance of POEM-based representations with other well-known results on the FERET dataset. We first consider the FERET'97 results [99], and results obtained with the state-of-the-art *elementary* facial feature sets including LBP, HOG, and Gabor filter based methods, including HGPP & LGBP-HS. We then consider the more recent results in [146, 137, 80], which are obtained by using *learned* descriptor or a more complex classification step. These results, to the best of our knowledge, are the state-of-the-art with respect to the FERET dataset.

4.4.2.1 Performance of POEM-HS descriptor

Before comparing our algorithm with other *elementary* descriptors, it is worth noting that among considered distance metrics, χ^2 measure performs the best (see Figure 4.6) (we also evaluated the *Two-step* normalization associated with $L2$ but this gives slightly worse results than χ^2). Therefore, we use χ^2 distance associated with the POEM-HS descriptor.

As can be seen from Table 4.2, in comparison with the conventional LBP and HOG methods (the performance of HOG for face recognition is reported in [80]), our POEM descriptor is much more robust to lighting, expression & aging, illustrated by significant improvements in recognition rates for all probe sets. When compared with LGBP and HGPP, reported as being the best performing descriptors on the FERET database, POEM provides comparable performance for the probe sets Fb and Fc. When we consider the more challenging probe sets Dup1 and Dup2, POEM outperforms LGBP and is comparable to HGPP.

Table 4.2 also shows that the retina filter preprocessing enhances the performance of POEM, especially for the probe set Fc. It is worth re-emphasizing that our retina filter preprocessing considerably improves the performance of LBP features (and also Gabor filters, as in Figure 3.19). The recognition rates shown in two last lines of Table 4.2 have been obtained by using the available code from authors. These results are slightly worse than those presented in the original paper (see the first line in Table 4.2). This is probably due to the fact that authors use an other mask for face cropping.

Summarizing :

1. The high-quality performance of the *elementary* POEM-HS algorithm without any preprocessing clearly proves its strength for face recognition : it is robust to the variations of expression (Fb set), illumination (Fc set), and aging (Dup1 and Dup2 sets).
2. The retina filter preprocessing is always useful, it also enhances the performance of POEM-HS descriptor.

Methods	Fb	Fc	Dup1	Dup2
LBP [2]	93.0	51.0	61.0	50.0
LGBP [144]	94.0	97.0	68.0	53.0
HGPP [141]	97.6	98.9	77.7	76.1
HOG [80]	90.0	74.0	54.0	46.6
POEM-HS*	**97.6**	**95**	**77.6**	**76.2**
Retina filter + POEM-HS*	**98.0**	**99**	**79.2**	**78.8**
LBP	*92.5*	*51*	*58*	*47*
Retina filter + LBP	*94.9*	*96.5*	*67.2*	*59*

* These rates are slightly worse than those presented in our previous work [129] where we used 8 neighbors for each cell, the results on Fb, Fc, Dup1 & Dup2 were 98, 99, 77.8, 76.5% for **POEM-HS** and were 98.1, 99, 79.6 and 79.1% **Retina filter + POEM-HS**, respectively.

Tableau 4.2 – Recognition rate comparisons with state-of-the-art *elementary* descriptors tested with FERET evaluation protocol.

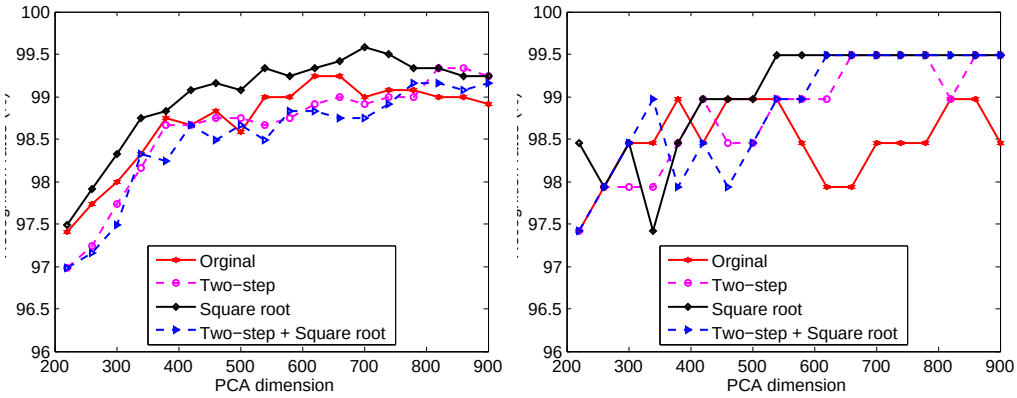
4.4.2.2 Performance of *learned* POEM descriptor

We first investigate the effects of different normalization methods and PCA dimensions. Figure 4.7 presents the recognition rates obtained with 4 normalization methods and different PCA dimensions which vary from 200 to 900. It is clear that *Square root* algorithm outperforms the others. Considering the effects of PCA dimension numbers, the recognition rates obtained on the Fb & Fc probe sets is really stable when the PCA dimension varies from 600 to 900 (see Figures 4.7(a,b)). On the Dup1 & Dup2 probe sets, the performance reaches its maximal value when the 700 first principle components are used (c.f. Figures 4.7 (c,d)). For the average performance on four probes, we obtain the *highest results when PCA dimension ranges from 600 to 900*, and in particular the ***700 dimension square-root representation performs the best***. We thus use this representation in the rest of this section.

We now compare our results with the state-of-the-art. We consider the results of WPCA Gabor [32], WPCA LBP (the experiments were conducted by ourselves) [3], and WPCA LGBP [90] since similar to WPCA POEM, these algorithms apply a whitening PCA process on feature sets. Due to its dramatically high complexity in terms of both time and memory, we do not conduct the experiment for WPCA HGPP. To the best of our knowledge, the HGPP descriptor has not been re-evaluated by researchers other than the original authors, although it has been presented for several years.

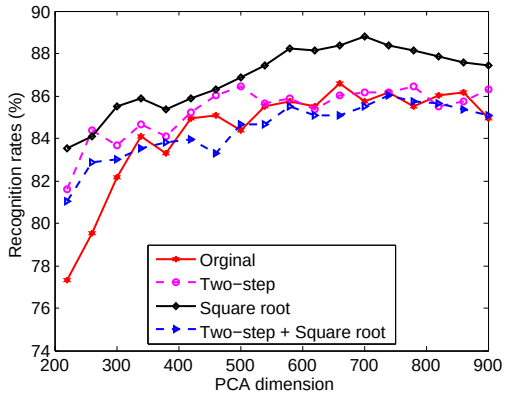
Generally, different facial components are of different importance for face recognition.

3. For a fair comparison, we repeat the experiments described above (images are preprocessed using retina filter ; normalization methods and PCA dimensions are variable) and report the best performance. Similarly, taking the square root of data values gives a small improvement, and the 500 dimension representation performs the best.

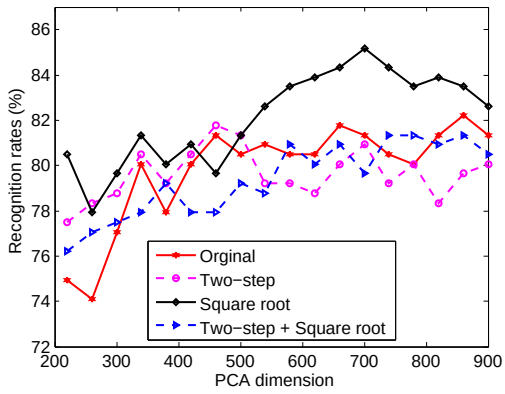


(a) Fb

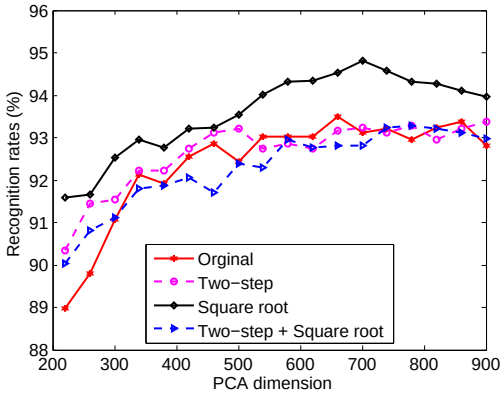
(b) Fc



(c) Dup1



(d) Dup2



(e) Average

Figure 4.7 – Effects of the PCA dimension when applying different normalization methods on the FERET probe sets.

Methods	Fb	Fc	Dup1	Dup2	Note
WPCA Gabor [32]	96.3	99.5	78.8	77.8	
WPCA LBP	98.5	99.0	81.0	75.0	
WPCA LGBP [90]	98.1	99.0	83.8	81.6	
Weighted LBP [2]	97.0	79.0	66.0	64.0	
Weighted LGBP [144]	98.0	97.0	74.0	71.0	
Weighted HGPP [141]	97.5	99.5	79.5	77.8	
Results of [146]	99.5	99.5	85	79.5	Gabor & hand labeling features
Results of [137]	98	98	90	85	LBP, Gabor, KPCA, KDCV
Retina + WPCA-POEM	**99.6**	**99.5**	**88.8**	**85.2**	

Tableau 4.3 – Recognition rate comparisons with state-of-the-art results on the FERET database.

Therefore, different weights can be set to different patches when comparing two faces. This is the principe behind *Weighted* methods which are included in our consideration. We also include the other state-of-the-art results, such as those presented in [146, 137]. It is clear from Table 4.3 that our algorithm outperforms significantly and systematically all competing algorithms.

4.4.3 Results on the LFW database

We duplicate the same experiments on the challenge LFW dataset. Similarly, our results are classified into two parts, the performance obtained by POEM-HS and WPCA-POEM algorithms. We follow the standard procedure described in [49] and report the mean classification accuracy ± standard error computed from 10 folds of the "Image-Restricted View 2" portion of LFW set.

- Concerning the POEM-HS algorithm, we use its optimal parameters : *3 unsigned bins,* $w = 7, L = 10, n = 6, \chi^2$ *distance.*
- When considering the performance of WPCA-POEM method, we use *View 1* of the LFW dataset for the PCA dimension choice.

Throughout the experiments, we use the POEM-based descriptor in a simple threshold-on-descriptor-distance classification context, meaning that for each test fold, an optimal threshold giving the highest separation score on the 5400 examples of the training set is chosen and then is used to calculate the classification accuracy for the 600 examples of the test set.

Reference	Descriptors (similarity measure)	Performance
Pinto2008 [102]	V1-like	0.6421 ± 0.0069
	V1-like+	0.6808 ± 0.0045
Wolf2009 [136]	LBP Euclidean/SQRT	0.6824/0.6790
	Gabor Euclidean/SQRT	0.6849/0.6841
	TPLBP Euclidean/SQRT	0.6926/0.6897
	FPLBP Euclidean/SQRT	0.6818/0.6746
	SIFT Euclidean/SQRT	0.6912/0.6986
	All combined	0.7521 ± 0.0055
This work	**POEM-HS***	**0.7369 ± 0.0059**
	POEM-HS* Flip	**0.7522 ± 0.0073**

* These rates are slightly worse than those presented in our previous work [129] where we used 8 neighbors for each cell, and the results of **POEM** and **POEM Flip** were 0.74 ± 0.0061 and 0.7542 ± 0.0071, respectively.

Tableau 4.4 – Recognition results of different face representations on the LFW set, Image-Restricted Training, View 2.

4.4.3.1 Performance of POEM-HS descriptor

In this section, we compare our results with other descriptor-based ones. Within the LFW dataset, there is significant pose variation. In order to address this, we first flip image 1 of each pair on the vertical axis, and then calculate two histogram distances : the first one is obtained from two original images of the considered pair ; the second one is obtained from the image 1 "flip" and the image 2. Finally, we take the smallest of the two histogram distances as our measure. This simple pre-processing step improves recognition rates and is refereed to POEM-HS Flip in the following results.

Because of the poor quality of the images in the LFW dataset, retina filtering does not improve the recognition results (but it does not reduces the recognition rates on LFW dataset). So the recognition rates related to **Retina filter + POEM-HS** are not presented in table 4.4. Note that :

- The retina filter removes illumination variations and enhances the image contours but may also enhance image artifacts, such as those arising from compression which are numerous with low quality images in the LFW database.
- In Section 3.4.4 – Chapter 3, we have already pointed out that two parameters σ_{Ph} and σ_H should be tuned according to the image quality, in particular with blurry images where information lies in low frequency space. But in this work, we fixed the parameters for all images, although they are of different qualities.
- A fully automatic retina filter where parameters are tuned according to the image quality is also perspective of future work.

It is clear from Table 4.4 that POEM-HS method outperforms all other competing

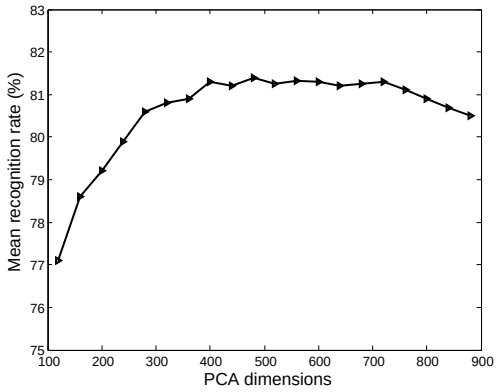


Figure 4.8 – Effects of the PCA dimension on LFW "View 1".

descriptors : LBP, TPLBP, FPLBP, Gabor filters and SIFT. When compared with these descriptors, the POEM-HS method presents around 20% reduction in classification error. Our POEM-HS Flip mean recognition rate 75.22% is comparable to that of the "combined" method of [136]. It is worth noting that Wolf et al. [136] combine 10 descriptor/mode scores using SVM classification. The results in [102], based upon Gabor filters, are much worse than ours.

4.4.3.2 Performance of *learned* POEM descriptor

Similar to Section 4.4.2.2, this section details firstly the PCA dimension choice. Since the *Square root* normalization produces better performance than other techniques (see Figure 4.7), it is used in the following tests. In order to determine the optimal number of PCA dimension on LFW dataset, the following procedure is carried out :

1. We use 1000 images of *training set* from View 1 to calculate the covariance matrix and the PCA projection axes.
2. All images of View 1 are then projected on the obtained PCA axes. Thus, within View 1, the *training set* has 2200 projected pairs (1100 *Same* and 1100 *Diff*), while the *Test set* has 1000 projected pairs (500 *Same* and 500 *Diff*).
3. Similar to experiments conducted on View 2 portion for final performance reporting (refer to Section 4.4.3.1), an optimal threshold giving the highest separation score on the 2200 projected pairs of the *training set* of View 1 is chosen.
4. The optimal threshold is then used to calculate the classification accuracy for the 1000 projected pairs of the *test set* of View 1.

This procedure means that we follow strictly the protocole associated with the LFW dataset : we can only use images from View 1 for parameter choice, which involves the number of PCA axes in our case.

The classification accuracies obtained when PCA dimension varies are shown in Figure 4.8. As can be seen in this figure, the classification rates are very high and quite similar when the PCA dimension ranges from 400 to 700. Out of this range, the performance drops. The optimal range of about [600, 900] obtained on FERET dataset and the optimal range of about [400, 700] on LFW base suggest that with respect to the performance of PCA-POEM descriptor, the PCA dimension should be chosen between 600 and 700 because those representations perform well in both constrained and unconstrained context. However, in the following experiments on LFW database, we will use 400 dimension representation. The motivation for this choice is that this representation yields a feature vector which has the same dimension as the LE descriptor [21], a *learned* representation which produces one of the best results on the LFW dataset.

After being projected on PCA axes, the discriminative information of low quality images (with noise, compression effect) will mainly lie in few components (the eigenvectors associated with the biggest eigenvalues) and the remaining components contain rather useless information. On the contrary, the discriminative information of a high quality image when projected on PCA axes will lie in more components. That is why fewer PCA components are necessary on the LFW database.

We now project all 12000 POEM-HS vectors representing 12000 images of View 2 on the 400 obtained PCA projection axes in order to obtain 12000 WPCA-POEM descriptors. Using a simple threshold-on-descriptor-distance classification context, we calculate the performance of WPCA-POEM on View 2 which is of $81.02\% \pm 0.57$. Using the simple *Flip* pre-processing, a performance improvement of 1.6% points is gained, resulting in the classification rates of WPCA-POEM Flip of $82.63\% \pm 0.63$.

Further experiment

Thanks to the structure of the LFW dataset, we further carry out the next experiment. We propose to apply the LDA algorithm upon the WPCA-POEM feature vectors in order to obtain more discriminative features. The LDA algorithm needs to calculate two covariance matrices : the intra-class and between-class. Within View 1, we use the *Same* and the *Diff* pairs to calculate the intra-class and between-class covariance matrices, respectively. The LDA projection axes are then calculated using these two matrices. All the WPCA-POEM feature vectors are now projected on these LDA axes. Note, all the WPCA-POEM feature vectors are normalized using the $L2$ norm before projected. By varying the number of LDA axes, we obtained the highest separation rate on View 1 when 200 LDA components are considered. Therefore, we use the 200 "LDA + WPCA-POEM" representations to report the final performance in View 2, which is shown in Table 4.5. (the term "LDA + WPCA-POEM" means that the WPCA-POEM feature vectors are projected on the LDA space.)

Table 4.5 presenting increasing recognition results of all published methods on LFW set, Image-Restricted Training, View 2 shows that our algorithm reaches the state-of-art result with a considerable reduction in the complexity (see also next section for more details).

4.4.4 Runtime and storage requirements

4.4.4.1 POEM-HS descriptor

We compare the complexity of the POEM-HS representation with two of the most widely used descriptors for face recognition : LBP and Gabor wavelet based methods.

• **Runtime of feature extraction :**
Considering the pure one-LBP-operator method, POEM based face recognition requires a computational complexity which is 1.8 [4] times higher (the calculation of integral gradient image is very fast when compared to the calculation of POEM features and the construction of POEM-HS) but at the same time, there are remarkable improvements in recognition rates on the FERET database (+4.6%, +44%, +16.6% and +26.2% for the probe sets Fb, Fc, Dup1 and Dup2, respectively). And on the LFW set, POEM method also outperforms LBP and its variants, TPLBP and FPLBP.

When we consider Gabor filter based descriptors, only the runtime required for the convolution of the image with the family of Gabor kernels (8 orientations and 5 scales) is necessary. For a stronger comparison, we use 1000 images with size of 96x96 pixels. The average runtime per image are shown in Table 4.6 where we see that the computation of the whole POEM descriptor is about 28 times faster than the computation of just the first step of Gabor feature extraction.

We do not calculate here the time required to extract SIFT descriptor and do not compare directly it to POEM. But as argued in [45], SIFT is about 3 times slower than 3×3 grid Center-Symmetric LBP (CSLBP), a variant of LBP (3×3 grid CSLBP means that the descriptor is obtained concatenating the histogram of CSLBP features over grid of 3×3). Thus it seems that POEM and SIFT have similar time complexity. However, for face recognition, POEM clearly outperforms SIFT, representing about 20% reduction in classification error on the LFW set.

• **Storage requirements :**
Considering data storage requirements, for a single face, the size of a complete set of POEM descriptors is 13 and 27 times smaller than that of LGBP and HGPP (LGBP calculates LBP on 40 convolved images while HGPP encodes both real and imaginary

4. In the previous work [129], we used $n = 8$ which yielded a descriptor of 3.3 time more complex than the LBP method

Reference	Perf.	Std. Err.	Notes
1. Pixels and linear SVM [101]	0.5995	0.0064	
2. Eigenfaces, original [123]	0.6002	0.0079	
3. Gabor (Euclidean) [135]	0.6293	0.0047	
4. V1-like linear SVM [101]	0.6421	0.0069	
5. LBP (Hellinger) [135]	0.6782	0.0063	
6. V1-like+ linear SVM [101]	0.6808	0.0044	
7. Pixels/MKL [103]	0.6822	0.0041	
8. FPLBP (Euclidean) [135]	0.6865	0.0056	
9. TPLBP (Hellinger) [135]	0.6890	0.0040	
10. 3x3 MRH [114]	0.7038	0.0048	
11. MERL [50]	0.7052	0.0060	
12. Combined lbp-gabor-tplbp-fplbp [135]	0.7062	0.0057	
13. Nowak, original [50]	0.7245	0.0040	
14. 3x3 Multi-Region Histograms (normalised) [114]	0.7295	0.0055	E : needs cohort of negative faces
15. Nowak, funneled [50]	0.7393	0.0049	
16. MERL+Nowak, funneled [50]	0.7618	0.0058	Combination of 11. and 13.
17. One-shot learning on lpb-gabor-tplbp-fplbp [135]	0.7653	0.0054	E : Uses 1 shot learning
18. Hybrid descriptor-based, funneled [135]	0.7847	0.0051	E : Combination of 12. and 17.
19. LDML, funneled [42]	0.7927	0.0060	Uses methods 5. 8. 9. and SIFT features
20. V1-like+/MKL [103]	0.7935	0.0055	Around 1000 gabor filters + ad-hoc features
21. WPCA-POEM	**0.8102**	**0.0057**	
22. Single LE + holistic [21]	0.8122	0.0053	
23. WPCA-POEM Flip	**0.8263**	**0.0063**	
24. Attribute classifiers [61]	0.8362	0.0158	E : big training set (65000 hand labelled)
25. Hybrid, aligned [120]	0.8398	0.0035	E : 1 shot learning
26. Simile classifiers [61]	0.8414	0.0131	E : big training set (hundreds of images of 60 additional reference people)
27. Multiple LE + comp [21]	0.8445	0.0046	
28. LDA + WPCA-POEM Flip	**0.8496**	**0.0043**	
29. Attribute and Simile classifiers [61]	0.8529	0.0123	E : Combination of 24. and 26.
30. Combined b/g samples based methods, aligned [120]	0.8683	0.0034	E : uses 1 shot/2 shot learning+ranked distances

Tableau 4.5 – Increasing recognition results of (as far as we are aware) all published methods on LFW set, Image-Restricted Training, View 2. **Bold-face** indicates methods introduced in this thesis, *italics* and the letter E : in the description column indicates extended methods. Note that the methods to date which outperform our method are all either *extended* methods, or are methods which use multiple feature sets with large numbers of Gabor features.

Methods	Times (ms)
LBP extraction*	9.0
POEM extraction	16.1
Convolution with 40 Gabor kernels	434.9

Calculated using the Matlab implementation, these times are only suitable for rough comparisons of computational complexity.
**This algorithm performs considerably less effectively than ours.*

Tableau 4.6 – Runtime required to extract the **whole** LBP-HS & POEM-HS descriptors, and runtime of **only the initial step** of Gabor based feature extraction.

Methods	Dimension^{+} (per patch)
LBP-HS [2]	59
POEM-HS	99
LGBP-HS [144]	10240
HGPP-HS [141]	20480
Combined* [136]	262

** We do not consider yet the dimension of the Gabor based features used in this algorithm.*
$^{+}$ dimension in "values"

Tableau 4.7 – Comparison of the stockage requirements of different face representations.

images, requires 80 convolutions, and operates on 90 images). Note that these comparisons are roughly done considering all 256 patterns of our POEM features. However, we use only 33 uniform POEMs, meaning that the size of POEM descriptors used here is *103 and 232 times* smaller that of LGBP and HGPP (these ones use all 256 feature values). When compared to the "combined" method of Wolf et al. [136], the space complexity of POEM descriptor is considerably smaller. For one patch, the size of POEM-HS is 33×3 [5], while the size of method in [136] is $59 \times 2 + 16$ (the size of LBP, TPLBP, and FPLBP per patch are 59, 59 & 16, respectively) + 128 (dimension of SIFT) + size of Gabor based descriptor (which is equal to the size of patch × number of scales × number of orientations as in [136, 80]).

4.4.4.2 PCA-POEM descriptor

All *learned* descriptors first require to extract the whole feature vector. In the rest of this section, we adopt the term 1^{st} *step* to refer this step. As consequence, all *learned* Gabor-based representations suffer from a computationally intensive feature extraction.

5. In the primarily work [129], we used 8 neighbors for each cell, thus the size of POEM-HS is 59×3 features per patch – see explanations in Section 4.3.1.

Methods	Extraction(1)	Dimension^{+}	Comp. 2img(2)	1000 pairs(3)	(1) + (3)
LBP-HS	9.0	3776	0.213	213	222
POEM-HS	16.1	6336	0.323	323	348.1
LGBP-HS	795.0	655360	51.186	51186	51981
HGPP-HS	1185.1	1474560	102.613	10261	11446
LE	39.8	400	0.022	22	61.8
PCA-POEM	20.2	400	0.022	22	40.2

(1,2,3) times in ms, $^{+}$ *dimension in "values",*
(2) time required to compare two facial vectors, (3) time required to compare 1000 pairs of facial vectors, (1) + (3) the total time required to represent a novel face image and compare it with 1000 other images.

Tableau 4.8 – Comparison of the complexity of different face representations.

In this section, we detail the comparison of our PCA-POEM representation with the LE descriptor [21], which results one of the best results on the LFW database. The PCA-POEM and LE algorithms have the equal size but the computational time of 1^{st} *step* of WPCA-POEM algorithm is smaller than that of LE descriptor. By re-implementing the presented LE algorithm and trying to optimize the best possible in Matlab, we find that the 1^{st} *step* of the LE descriptor is about 2.2 times slower than ours. Note that, before applying the PCA dimensionality reduction step, the LE descriptor requires 256 histogram bins per patch (POEM requires 99 bins). Moreover, besides finding the PCA projection axes, the LE algorithm also needs to construct the Bag-of-Word model.

Before ending this section, let see Table 4.8 for the comparison of the complexity of different methods. Assume that the face images are divided into 64 patches and we need to compare a test image with 1000 images stocked in database for a "classification task". It is clear that our methods including POEM-HS and PCA-POEM are of low complexity. Moreover, the PCA-POEM algorithm is an real-time method.

4.4.4.3 Conclusion

With above considerations, we argue that the POEM-based descriptor is a good candidate for high performance real-time face recognition. Low complexity descriptors provide worse results ; whilst representations based upon multiple feature types can achieve similar performance but are too slow for real-time systems.

4.5 Conclusions

This chapter addresses the second step in the face recognition pipeline : feature extraction. By a detailed consideration on the existing face representation algorithms, we have pointed out that most of them do not balance the three criteria : the discriminative power, the robustness and the low complexity in both terms of runtime and stockage requirements. We then developed a novel feature set, referred to Pattern of Oriented Edge Magnitudes (POEM), by applying the self-similarity operator on accumulated edge magnitudes across different directions, and proved these features satisfy all these criteria. By carefully designed experiments, we have also determined the optimal parameters of POEM features for face recognition : *unsigned representation with 3 bins, built on 10x10 pixel blocks and 7x7 pixel cells, with 6 neighbors for each considered cell*. Using these features, we presented the first face representation algorithm, the POEM-HS descriptor, with several desirable properties. It is robust to the variations of pose, expression and aging, and is very fast to compute.

We further proposed to apply the PCA dimensionality reduction technique, followed by a whitening processing, upon the POEM-HS descriptor, in order to build the WPCA-POEM descriptor. Compared with the POEM-HS descriptor, the WPCA-POEM descriptor is even more discriminative and robust, notably more compact : *our representation dimension is only between 400 and 700*, depending on the context. We have proved the strength of our descriptors by presenting the *state-of-the-art results* on both constrained and un-constrained face recognition tasks. This very high performance coupled with the high speed of extraction and low dimension size suggests that our POEM based representations (POEM-HS and WPCA-POEM) are good candidates for use in real-world face recognition systems. We have also proposed to apply the LDA technique upon the WPCA-POEM feature vectors to obtain the LDA-WPCA-POEM descriptors. Since the LDA algorithm is not naturally suitable for face recognition in one sample circumstance (of course, some work has been carried out to make LDA applicable to face recognition from one sample), we only evaluated the performance of LDA-WPCA-POEM on the LFW database (this was done thanks to its special structure).

Before ending this chapter, we wish to highlight that a more detailed analysis on false recognition cases will be considered in future work.

Up to now, we have built efficient pre-processing and feature extraction algorithms. By these, the first challenge of face recognition, illumination variation, is well solved. The next chapter will adresse the second challenge, pose variations.

Chapitre 5

Statistical model for pose-invariant face recognition from one reference sample

Once the problem of illumination is solved (by using either our retina filter in the first pre-processing step or the POEM algorithm in the feature extraction stage [1] or both), one of the remaining challenges for face recognition is pose variations [2]. This is the topic of this chapter. Face recognition algorithms perform very unreliably when the pose of the probe face is different from that of the enrolled face because feature vectors typically vary more with pose than with identity. In fact, among intra-personal variations, pose variation is the factor affecting the recognition performance the most importantly.

Similar to all other parts of this dissertation, we examine here the most difficult scenario in which the task is to recognize a person from previously unseen view using only one frontal image. We propose a novel statistical model for face recognition across poses. The principe behind our algorithm is to treat the relationship between frontal and non-frontal images as a statistical learning problem. Although this framework is *specifically* [3] designed and evaluated for face ***identification*** across poses, it is potentially be applied to other types of intra-personal variations.

In the remainder of this chapter, we first provide an overview of algorithms proposed

1. In Chapter 4, we have shown that the POEM-based representations are more robust to lighting change than many other considered features.

2. It is widely accepted that the two greatest challenges for face recognition are the variations of illumination and pose [145].

3. Face recognition systems can be divided into two categories : *general* and *specific* algorithms. *General* algorithms are designed for general purpose of face recognition equally handing all image variations, e.g. they do not contain specific tactics on handing pose or illumination variations, whereas *specific* algorithms are those with particularly designing mechanisms that can eliminate or at least compensate the difficulties brought by one or some types of particular image variations (e.g. pose variations here). According to this categorization, the retina filter and the POEM features are *general* algorithms since they can be applied for general face recognition systems.

in the literature for the specific problem of face recognition across poses in Section 5.1, then we detail the proposed algorithm in Section 5.2. Experimental results are presented in Section 5.3. Conclusions are finally given in Section 5.4.

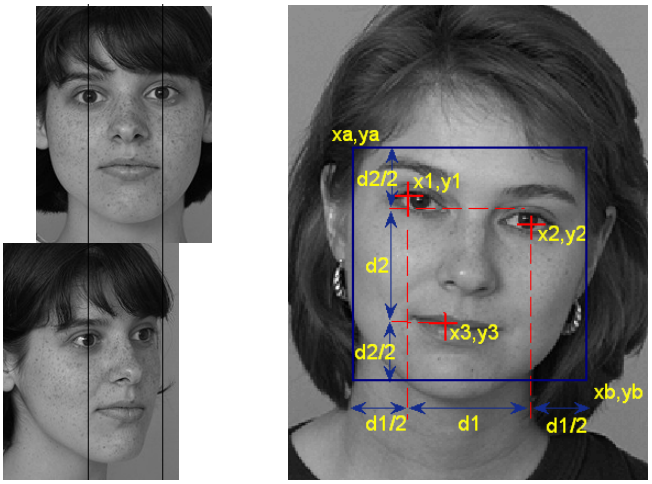
5.1 Related literature

5.1.1 Face alignement/registration in pose-invariant face recognition

As already mentioned at the beginning of this dissertation, face alignment/registration itself has emerged as an important research problem, but our work does not deal with this problem. However, the performance of many current *specific* state-of-the-art pose-invariant face recognition systems depend importantly on face registration. Therefore, we consider briefly this topic in this section.

In fact, all face recognition algorithms require some degrees of alignment so as to normalize for unwanted shape variations. Most of *general* algorithms presented in the literature use/advocate aligning images with respect to 2 or 3 facial landmarks such as center of eyes, nose tip etc. Faces aligned with 2-3 facial landmarks points are also called sparsely registered faces. On the contrary, some systems identify more than 80 landmarks (densely registered) per face. In the recent work [41], Gross *et al.* demonstrated that improved face recognition performance can be attained using dense registration (39-54 fiducial points depending on the pose) rather than sparse registration, ***especially in pose-invariant face recognition systems***. Similarly, Blanz and Vetter [18] demonstrated good performance using extremely dense offline registration (75972 vertex points on laser scan 3D images) and medium density registration (at least 7-8 fiducial points depending on pose) with the online 2D images. The contribution of increasing the landmark number used for face matching when pose varies is easy to understand since the pose change causes corresponding changes in the position of facial features. For example, when a face rotates from the frontal view to the left or right view, the distance between two eyes consequently reduces. If we fix the position of one eye, the position of the other eye will be changed, as shown in Figure 5.1(a) (of course, we consider faces on original images, without warping effect).

One potential solution to the alignment problem is to use a face representation robust to misalignment. In fact, several evaluations have been carried out [116, 40] and Gabor wavelet and histogram based descriptors, such as histogram of LBP, have been seen robust to *small misalignment*. The term *small misalignment* is used in the sense that the face area can be displaced of some pixels (translation of $\pm$ 2 pixels) and the rotation angle can be of some degrees (4°). Since our POEM based representations are histogram-



(a) (b)

Figure 5.1 – (a) : Pose change causes corresponding changes in the position of facial features ; (b) : Non-perfect alignment technique for cropping our face area.

based descriptors, they are consequently believed to be robust to small misalignment, at least as the LBP based representation. Note that while evaluating the performance of the POEM descriptor on the challenging LFW dataset (refer to Chapter 4), we already used the facial images aligned automatically using the algorithm of Wolf *et al.* [136]. Again, in this chapter, for the goal of re-validating the robustness to misalignment of our approach, all face areas used in our experiments are cropped in a non-perfect way, as shown in Figure 5.1 [4]. Mathematically, given the position of the eye and mouth regions, we calculate the coordinates of two pixels (top left pixel with coordinates (x_a, y_a) and bottom right pixel with coordinates (x_b, y_b)) of the face area : $x_a = x_1 - d_1/2; x_b = x_2 + d_1/2; y_a = y_{12} - d_2/2; y_b = y_3 + d_2/2$ where $d_1 = x_2 - x_1$ is the horizontal distance between the two eye regions ; $d_2 = y_3 - y_{12}$ is the vertical distance between the eye and mouth regions ($y_{12} = (y_1 + y_2)/2$). The term "position of eye regions" means that we need only the coordinates of any pixel in this region. If the angle between the line joining both eyes and the horizontal one is bigger than 4°, a rotation transform is used in order to eliminate the *serious* misalignment.

5.1.2 Illustration of the difficulties of pose variations

It is widely known that face recognition performs very unreliably when the pose of the probe face is different from the enrolled face (see Figure 5.6 in Section 5.3). In order

4. Commonly, face area is manually cropped so that the two eye centers are aligned at the same position (the line joining the centers of eyes needs to be horizontal and the distance between the eyes is nominal). To do this, the coordinates of the eye centers must be located very accurately which is not easy.

to re-validate this remark, we carry out the following *experiment* :

1. Using 900 images of 100 different individuals in the FERET database (9 images per person at 9 different poses), we first crop 900 "Eye" patches, the term "Eye" means that the considered patch is centered on the eye (in fact the left one).
2. We then consider the 100 frontal patches as gallery patches and the 800 other patches as probe.
3. For each probe and gallery patch, the similarity values are calculated using χ^2 distance of two LBP histograms and of two POEM histograms (the motivation for consideration both features is to illustrate the robustness of POEM features, see next paragraph for comparisons).
4. Two values obtained for a patch pair are then used as the coordinates of a pixel depicted on a 2D space. We classify those pixels according to the pose of the gallery patch. Also, a pixel belongs to the *same* class if two considered patches come from the same individual, and vice versa.

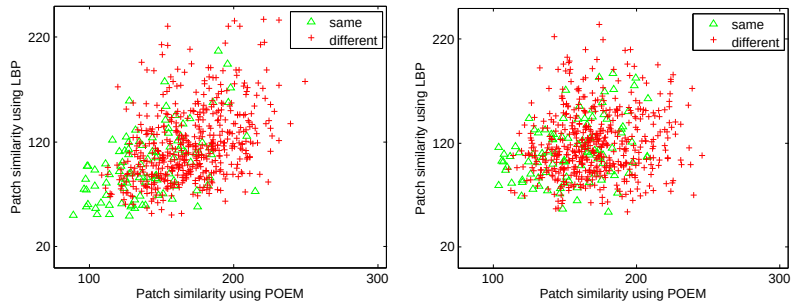
Given a pose, there are in total 100 *same* pixels and $\frac{100\times99}{2}$ *dif* pixels. We depict all 100 *same* pixels and only 500 *dif* pixels. The motivation for considering only local patches, not the whole face, is that the feature vector extracted from small region is less sensitive to pose changes than that from the whole face.

More overlapping the two *same* and *dif clouds* are, more similar the distance between two patches coming from same person and that of two patches coming from different persons is. And more difficult the recognition task is. There are therefore two main conclusions coming from Figure 5.2 analysis :

1. Recognizing a face from a wide range of pose variations is challenging : it is very difficult to separate the two *same* and *dif clouds* in Figures 5.2(a,b).
2. However, it is easier to separate the two *same* and *dif clouds* along the vertical direction than along the horizontal direction. Therefore, POEM features are more robust to pose than LBP features. This remark will be empirically proved by our experiments.

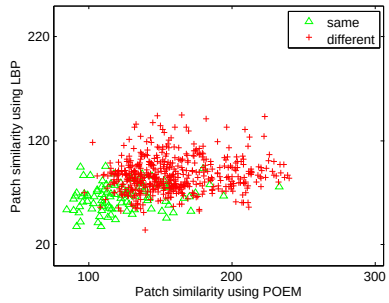
5.1.3 Algorithms for face recognition across poses

In the literature, many methods have been proposed to solve the problem of pose variations. The first and simplest method is to record each subject at each possible angle [12, 97, 69]. Beymer and Poggio [12, 13] are probably the first researchers to specifically handing pose variations in face recognition. Their algorithm in [12] requires 15 gallery face images to cover a range of pose variations with approximately $\pm$ 40° in yaw and $\pm$ 20° in tilt, as shown in Figure 5.3. The recognition process is a typical template

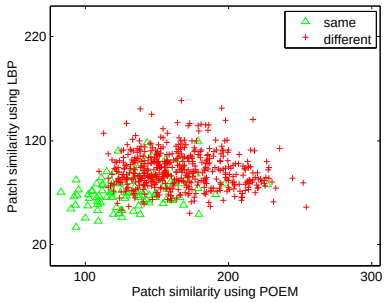


(a) $\phi_p = -60°$

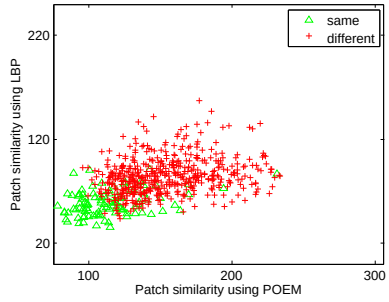
(b) $\phi_p = 60°$



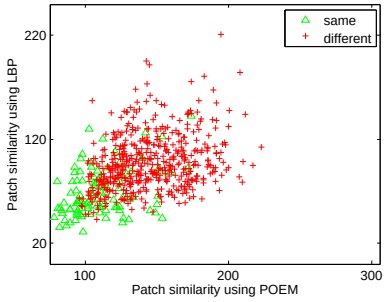
(c) $\phi_p = -40°$



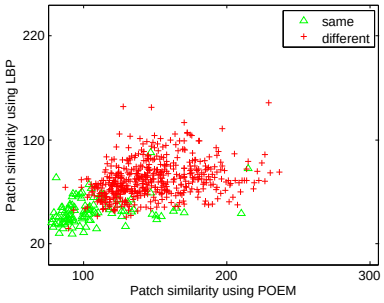
(d) $\phi_p = 40°$



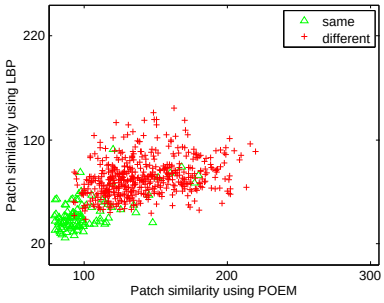
(e) $\phi_p = -25°$



(f) $\phi_p = 25°$



(g) $\phi_p = -15°$

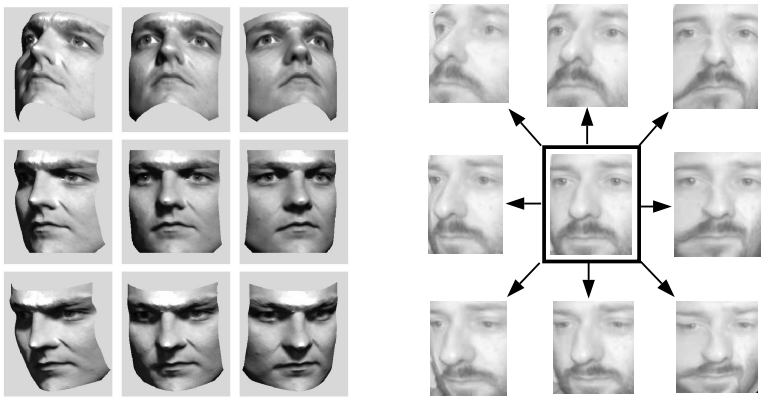


(h) $\phi_p = 15°$

Figure 5.2 – Distribution of similarities between two *Eye* patches of two different images, one frontal image and the other is with pose ϕ_p of : (a) -60° ; (b) 60° ; (c) -45° ; (d) 45° ; (e) -25° ; (f) 25° ; (g) -15° ; (h) 15° (given 2 patches, we calculate 2 values : χ^2 distances of LBP histograms and of POEM histograms. We then use these 2 values as coordinates of a pixel which is depicted on 2D space).

matching algorithm where the only difference is that it matches a probe face image with gallery face images in similar poses. A related approach is to take several images of the subject and to use them to build a statistical model that can interpolate unseen views [125]. Another method [38] makes explicit use of geometric information and use several photos to create a 3D model of the head, which can then be re-rendered at any given pose to compare with a given probe. In [38], using 7 facial images per individual taken under different illumination conditions (see Figure 3.3(a) – Chapter 3), Georghiades *et al.* reconstruct a face cone which is then used to generate a face at other views, as shown in Figure 5.4(a). All of these methods are valid, and some produce high-quality results. However, they all require the cooperation of the user, multiple images, or special capture methods. They are consequently unsuitable for face recognition in one sample circumstances.

Figure 5.3 – Multiple face images taken in different poses of the same person [12].



(a) (b)

Figure 5.4 – (a) Synthesized images under variable pose generated from the training images shown in Figure 3.3(a) ; (b) : Surrounding images are synthesized from the center image [13].

In order to overcome the difficulty of collecting multiple images in different poses,

a feasible solution is to synthesize virtual views to substitute the demand of real views from a limited number of known views. Beymer and Poggio [13] proposed a parallel deformation to generate virtual views covering a set of possible poses from a single example using feature-based 2D warping. Figure 5.4(b) shows eight virtual views (surrounding images) which have been synthesized from an image of frontal view. This algorithm can generate the face at novel views, covering -30° to 30° rotations in yaw and -15° to 15° rotation in tilt. Another approach is presented in [39] where the well known Active Shape Models (ASM) [29] are used to generate the face at novel views. However, these algorithms can generate virtual view at *quite* small angles and they are therefore unsuitable for face recognition with wide range of pose variations. For example, in [39] the recognition performance drops to 67.5% and about 20% for rotations of 45° and 65°, respectively.

The second category of approaches takes a single probe image at one pose and creates a full 3D head model for the considered subject based on only one image, including parameters representing pose and illumination. Face recognition can be performed in two distinct ways. The first method is to directly compare the parameters representing the shape and the texture of the 3D model [17, 106]. In the second approach, the 3D model can be used to re-render the face at a new pose, and 2D methods can be used [16]. There does not seem to be very much empirical difference between these methods [16]. These methods, though producing good results, need a dense registration of the probe and requires a costly iterative optimization of the model parameters. These algorithms take the order of tens of minutes to create a 3D model from an image. This problem is partly mitigated if the second recognition style (re-rendering) is employed, as the models are built for the gallery images offline, but the registration of a new individual to the system is still slow. As a consequence, it is not suitable for applications such as video surveillance or Internet search for faces.

Another category of approaches has been developed : the aim is to look for features which are invariant to viewpoint. Though also producing promising results, such algorithm requires manual detection of a significant number of points, e.g in [104], 21 manual landmarks per face are needed.

The fourth and most common approach to face recognition across poses is the statistical approach. Here, domain specific information about the 3D world is eschewed, and the relationship between frontal and non-frontal images is treated as a statistical learning problem. The first idea is to model the statistical relationship in order to *transform* non-frontal faces into frontal views, or vice versa, so that standard 2D face recognition methods can be used [13].

The second idea aims at *directly* modeling how face appearance changes due to pose, across same subjects and among different subjects. The most common example of a statistical approach is probably the Bayesian analysis of Moghaddam and Pentland [84]

(this pioneer work is a *general* algorithm). Sanderson *et al.* [113] developed a Bayesian classifier based on mixtures of Gaussians and transformed the parameters of the model for non-frontal views. A further example of the statistical approach is the "eigen light fields" work of Gross *et al.* [41]. These models can be considered as global statistical methods [104] since they build models relating the whole face image at one pose to the whole face image at other poses. Inspired by the idea that local regions of a face change differently in appearance as the viewpoint varies, other authors [59, 3] developed local statistical models that relate patches of the frontal image to their counterparts in non-frontal images. Generally, the local statistical models outperform the global statistical ones.

Statistical approaches such as [59, 3, 127] are particularly attractive because of their low computational complexity. They automatically solve the one reference image problem since the appearance models are learned effectively from an offline database.

In this contribution, we propose a novel model based on local robust features for face recognition across poses. Our model is made of two stages : an offline learning stage and a recognition stage. In the learning stage, using an *independent* facial dataset, probabilistic models describing the joint probability distribution of face patches of a gallery and probe images at different poses are learnt. To do this, similarities between extracted features of face patches at frontal and other poses are computed. In the recognition stage, the distribution parameters obtained in the learning stage are used to compute the probability of two faces at different poses of being from the same or from different individuals.

In our previous work [127], we built the models upon two features : Gabor wavelets and LBP. That statistical model was referred to an *efficient* algorithm since it produced very high-quality results, and at the same time its complexity is considerably lower compared to other Gabor-based algorithms. Together with the proposition of the more *efficient* and *compact* POEM features, we present in this chapter a novel face recognition system where POEM features are integrated.

5.2 Proposed model

As in [84, 59, 127], the principle behind our algorithm is to model how facial appearance changes as the viewpoint varies. The novelty is in the building of the model upon *multiple* local features instead of luminance values. We also propose and compare two strategies for combining various descriptors : decision-level fusion and feature-level fusion [53]. Simply speaking, the decision-level fusion strategies combine several classifiers to make a stronger final classifier and feature-level fusion methods combine several feature sets into a single one which is then used in a classifier. In our model, the incoming

features are the similarity value between two face patches and the decision-fusion is the sum rule strategy. Our model is made of two stages : an offline learning stage in which the prior distributions are calculated, and the face recognition stage.

5.2.1 Obtaining prior distributions

At the training stage, we aim at computing the probability distribution for similarity between two face patches, given the pose of the probe :

$$P(S_r|w, \phi_p), w \in \{same, dif\}, \tag{5.2.1}$$

where S_r is the similarity between the rth patch of gallery and probe images (the similarity can be a scalar in the case of using decision-level fusion or a vector in case of using feature-level fusion), ϕ_p is the probe viewpoint and w defines whether the gallery and probe images are from the same subject or from different subjects. In other words, we need to model the distributions of each of the two *clouds* in Figure 5.2.

In [3], a log-normal distribution has been argued to be more efficient than the normal distribution in the case of using the sum of squared differences (SSD) as the similarity measure between two patches. However, by experimental results, we find that employing a normal distribution results in better fit when using the POEM, LBP, and Gabor features. This can be observed in Figure 5.2. Therefore, we detail in the rest of this section the use of the normal distribution.

• In the case of a decision-level fusion strategy where the similarity is a scalar, distributions in Equation 5.2.1 are calculated for each considered feature as :

$$P(s_r^{f_i}|w, \phi_p) = \frac{1}{\sqrt{2\pi}\sigma_r^{w,f_i}} exp\left[-\frac{1}{2}(\frac{s_r^{f_i} - \mu_r^{w,f_i}}{\sigma_r^{w,f_i}})^2\right], \tag{5.2.2}$$

where f_i refers to the considered feature, e.g. POEM or LBP ; μ_r^{w,f_i} and σ^{w,f_i} are the similarity mean and standard deviation of class w using the feature f_i.

• In the case of a feature-level fusion strategy where the similarity between two patches is a vector, distributions in Equation 5.2.1 are calculated as :

$$P(\mathbf{s}_r|w, \phi_p) = \frac{1}{2\pi\|\mathbf{\Sigma}_r^w\|^{\frac{1}{2}}} exp\left[-\frac{1}{2}(\mathbf{s}_r - \boldsymbol{\mu}_r^w)^T \mathbf{\Sigma}_r^{w^{-1}} (\mathbf{s}_r - \boldsymbol{\mu}_r^w)\right], \tag{5.2.3}$$

where $\mathbf{s}_r^w = \{s_r^{w,f_i}\}$, $\boldsymbol{\mu}_r^w = \{\mu_r^{w,f_i}\}$ are vector of similarity and vector of mean of class w, and $\mathbf{\Sigma}_r^w = \begin{bmatrix} (\sigma_r^{w,f_1})^2 & 0 & 0 \\ 0 & (\sigma_r^{w,f_2})^2 & 0 \\ \dots & & \end{bmatrix}$ is the covariance matrix.

5.2.2 Recognition across poses

In the recognition [5] stage, given a probe image I^p at pose ϕ_p, for each image in the gallery set I^g, we compute the posterior probability in order to decide if this probe image comes from the same subject.

• At patch level, given the similarity value, the posterior probability that the probe patch and the gallery patch are of the same identity is calculated using Bayes rule :

$$P(same|S_r, \phi_p) = \frac{P(S_r|same, \phi_p)P(same)}{P(S_r|same, \phi_p)P(same) + P(S_r|dif, \phi_p)P(dif)}, \tag{5.2.4}$$

• In the decision-level fusion (sum rule) strategy, the image-level posterior probability is computed as the summation of all patch-level posterior probabilities obtained over all considered features :

$$\begin{aligned} P(same|I^g, I^p, \phi_p) &= \sum_r P(same|r^g, r^p, \phi_p) \\ &= \sum_r \sum_{f_i} P(same|s_r^f, \phi_p), \end{aligned} \tag{5.2.5}$$

where $P(same|s_r^{f_i}, \phi_p)$ is calculated using Equations 5.2.2 and 5.2.4.

• In the feature-level fusion strategy, the image-level posterior probability is computed as :

$$\begin{aligned} P(same|I^g, I^p, \phi_p) &= \sum_r P(same|r^g, r^p, \phi_p) \\ &= \sum_r P(same|\mathbf{s}_r, \phi_p), \end{aligned} \tag{5.2.6}$$

where $P(same|\mathbf{s}_r, \phi_p)$ is calculated using Equations 5.2.1 and 5.2.4 (note that $\mathbf{s}_r$ is a vector).

• When the pose ϕ_p is not known, Equation 5.2.4 can not be used directly. We therefore use the marginal distributions : $P(S_r|same) = \sum_p P(\phi_p)P(S_r|same, \phi_p)$ and $P(S_r|dif) = \sum_p P(\phi_p)P(S_r|dif, \phi_p)$. Then, instead of employing Equation 5.2.4, we use :

$$P(same|S_r) = \frac{P(S_r|same)P(same)}{P(S_r|same)P(same) + P(S_r|dif)P(dif)}, \tag{5.2.7}$$

5.2.3 Properties

Our model is made of two stages : an offline learning stage and the face recognition stage. The first stage aims at modeling the relationships between the face patch

5. Throughout this chapter, we consider only the identification task.

appearance at frontal view and that at profile view by some probability distribution functions. In the recognition stage, using the distribution parameters previously obtained, we compute the probability of two faces at different poses of being from the same or from different individuals. The output of our model is new *distance metric*.

Similar to [59, 127], as the appearance models are learned effectively from an offline database, our model automatically solves the one reference image problem. Another advantage is that adding a new person's image in the gallery does not require re-training existing models.

Considering the complexity, the proposed model is very fast. The training stage is carried out in advance whereas the recognition stage requires only the calculations of probability in addition. Those are arithmetic calculations and very fast (in fact, it takes us only 0.003 ms to calculate these probability values for a patch or in other words, the complexity of system is not too much modified).

5.3 Experiments and Discussions

5.3.1 Experiment setup

In our experiments, we use the FERET pose database which consists of images from 200 subjects. For each subject, images have been captured at viewpoints $ba, bi, bh, bg, bf, be, bd, bc, bb$ which correspond to viewpoint angles of $0\,^\circ$ (frontal), $-60\,^\circ$, $-40\,^\circ$, $-25\,^\circ$, $-15\,^\circ$,$+15\,^\circ$, $+25\,^\circ$, $+40\,^\circ$, $+60\,^\circ$, as illustrated in Figure 5.5. In our experiments, all facial images have been cropped in a non-perfect alignement way, as shown in Figure 5.1(a), then resized to 80x104 pixels and preprocessed by using the retina filter. The face area is divided into 8x8 non-overlapping patches [6].

When using LBP as facial features, each face patch is characterized by a vector of 59 values corresponding to histograms of the LBP^{u2} ($u2$ refers to *uniform* LBP). When using the POEM features, each face patch is characterized by a vector of 99 values corresponding to histograms of the *uniform* POEM codes. The similarity between two patches is also computed using the X^2 distance in both cases.

When using Gabor filters [7] for feature extraction, for one input image, we calculate the 40 convoluted images using 40 Gabor kernels, as commonly used. Each output image

6. In the previous work [127], we normalized face areas to 100x120 pixels and then divided them to 10x10 non-overlapping patches, and this results in slightly better performance. But here we divide the face areas into 8x8 non-overlapping patches as all other tests in this dissertation

7. Due to the high computational cost, this algorithm is not recommended to use, at least in the context of the Birofale project. However, for the clarity of experimental results, we present the results of this algorithm here.

Figure 5.5 – Example images in the FERET database : cropped images for 0°, −60°, −40°, −25°, −15°,+15°, +25°, +40°, +60° views (left to right). Upper row : original cropped images. Lower row : images preprocessed using the retina filter.

after the convolution is down sampled to give a 8x8 pixel image. Since the facial image is also divided into 8x8 non-overlapping patches, the descriptor of every face patch is a vector of 40 values. The cosine distance is used to calculate the similarity between two patches.

The 200 persons in the FERET [8] pose dataset are divided randomly into two groups of equal size : group A and group B. We first learn our model using images from Group A and conduct the recognition experiments on Group B. To cross-validate, our model is then learnt on Group B and experiments are conducted on Group A. For each recognition experiment, only frontal images are used as gallery and all non-frontal images are used as test set. The final recognition rates are calculated as the mean of recognition rates obtained with respect to probe pose.

The pose of probe image can be one of 8 values, thus we set $P(\phi_p)$ to 1/8 and class priors $P(same) = 1/8$, $P(dif) = 1 - P(same)$.

5.3.2 Robustness to pose changes of facial features

In this experiment, we compare the robustness to pose changes of face representations, *not yet the performance of the presented algorithm*. We use all 200 frontal images as reference and other 1600 images at 8 profile poses as test to calculate the identification rates (the Nearest neighbor classifier is used). Four different face representation algorithms are included : POEM-HS, LBP-HS, Gabor and *Eigenface*.

There are two main conclusions coming from Figure 5.6 analysis :

8. We did evaluate the performance of the proposed model on the LFW dataset–to do this, we used the image pairs in *View 1* to choose the threshold of probability and then reported the verification rates on *View 2*. The performance increases from 75.2 to 76.1, meaning that the gain is not very great. It is not very disappointed since this real-world database is very difficult–it contains not only the pose variations. Readers are referred to the section 6.4 in the next chapter for more details about these experiments.

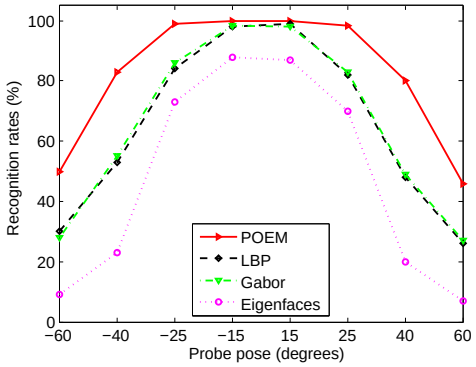


Figure 5.6 – Performance of different features when pose varies (the nearest-neighbor classifier associated with the χ^2 distance when using POEM/LBP histogram features or associated with the cosine distance when using Gabor/Eigenfaces methods).

1. This figure first shows the importance of designing algorithms for pose-invariant face recognition : the classical *Eigenface* method fails to work from the pose angle of 40°.
2. Among all other algorithms which are more robust to pose changes than *Eigenface*, POEM-HS performs the best. The performance of LBP-based and Gabor-based algorithms drops to about 50% and 30% when the pose of probe face is 40° and 60° whereas the performance of POEM-HS is still higher than 80% and 50%, respectively.

5.3.3 Performance of probability distributions

In the following experiments, except the last one, we assume probe pose to be known. We compare the performance of using different probability distributions for approximating the patch similarity distributions (or for modeling the *clouds* in Figure 5.2). Three probability models are considered :

1. ***Log-normal distribution*** : in [3], authors show that log-normal distributions give better results than normal distributions when the sum of squared difference (SSD) is used to measure the similarity between two face patches.
2. ***Normal (Gauss) distribution***
3. ***Gaussian Mixture Models (GMMs)*** : we use the EM (Expectation Maximization) algorithm for estimating the parameters of GMMs. In fact, the normal distribution is a special case of GMMs (one Gauss function is used).

By varying the number n of GMMs ($n = 2, 3$), we find that 2 GMMs results in similar performance to the normal distribution. The recognition rates performance even drop when 3 GMMs are employed. This is probably due to the fact that we do not have enough

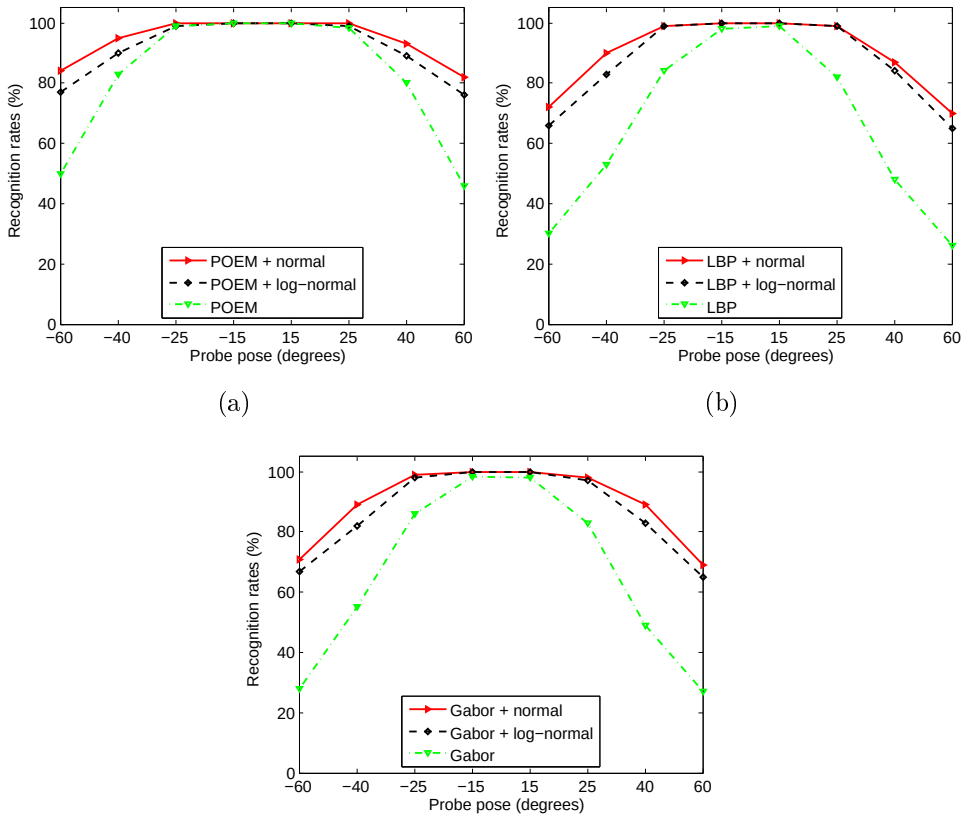


Figure 5.7 – Performance of different probability distributions when using with different patch features : (a) POEM ; (b) : LBP ; (c) : Gabor wavalets.

data for training GMMs (there are only 100 *same* pixels per pose). However, we believe that if more training data is available, GMMs will give better results. The performance of GMMs is not shown in Figure 5.7 since it is similar to the normal distribution. It can be seen from Figure 5.7 that the normal distribution outperforms the log-normal distribution for all POEM, LBP and Gabor features.

Figure 5.7 proves the ***efficiency of our algorithm*** : it enhances significantly the performance of all considered face representation methods.

5.3.4 Performance of fusion strategies

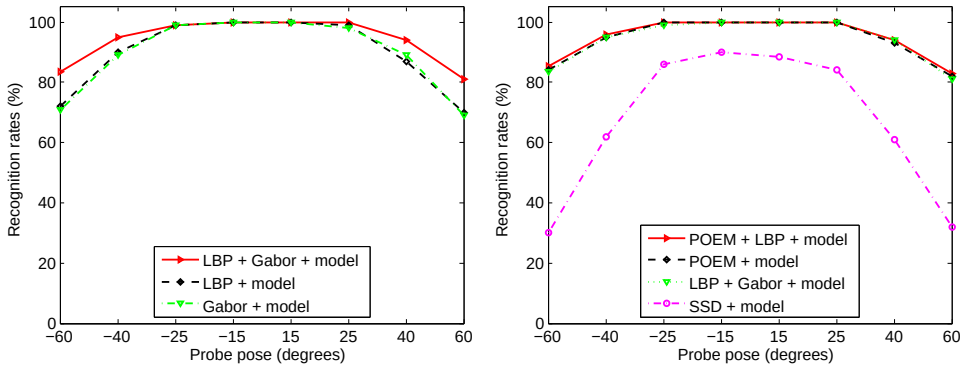
We calculate the recognition rates in several cases : we first combine LBP and Gabor features, as in our previous work [127], and then combine POEM with each of them and both. By these tests, we have several main remarks :

1. In all cases, feature-level and decision-level fusion strategies result in similar rates. Therefore, we depict only the results of the decision-level fusion in Figure 5.8.
2. When using the LBP and Gabor features, we obtain similar results, and fusion strategies combining them provide a significant performance gain, notably for the wide range of pose variations (refer to Figure 5.8(a)).
3. The performance of using only POEM features is as good as the fusion strategies combining both LBP and Gabor algorithms (see Figure 5.8(b)). This again shows the *strength of the POEM features*.
4. When the POEM algorithm is combined with other features, there is only a slight improvement (see Figure 5.8(b)). Therefore, in the rest of this chapter, we consider the results obtained by joining our POEM features with the proposed model.
5. We also include the results obtained when SSD is used as the patch similarity (SSD is used by Kanade and Yamada [59], and by Ashraf *et al.* [3] [9]). Figure 5.8(b) shows that SSD results in the poorest performance because of the sensitivity of raw pixel to appearance variations.

5.3.5 Unknown probe pose vs. known probe pose

We calculate performance in the case of unknown probe pose. In this case, we have to use the marginal distributions (Equation 5.2.7). Figure 5.9 shows that our model still results in very high recognition rates even when the probe pose is unknown.

9. When SSD is employed, we use the log-normal probability distribution.



(a) (b)

Figure 5.8 – (a) Performance of fusion strategies combining LBP and Gabor features ; (b) : Performance of fusion strategies combining POEM with other features.

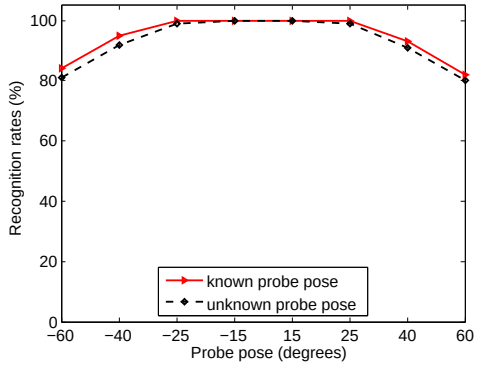


Figure 5.9 – Comparison of the recognition rates when the probe pose is known or not.

Methods	Database	Pose Diff (°)	%Correct	Landmarks
Gross *et al.* [41]	FERET (100)	30	75	39
Gross *et al.* [41]	PIE (34)	6/62	93/39	39
Blanz et al [16]	FRVT (87)	45	86	8
Kanade *et al.* [59]	CMU PIE (34)	45/67.5	100/80	3
Prince *et al.* [104]	FERET (100)	23/65	100/91	21
Prince *et al.* [104]	PIE (100)	16/62	100/91	21
Ours	FERET (100)	25/40/60	100/94/83	Non-perfect

Note that the difficulty of the task depends on the number of individuals in the gallery. This is given in blankets after the database name.

Tableau 5.1 – Comparison of face recognition algorithms across poses.

5.3.6 Comparisons to other studies

When comparing identification rates in case of variable poses, there are three factors that should be carefully considered :

1. the number of individuals in the gallery (100 for our experiments). When there are more individuals, there are more people to confuse the probe with, and the identification task becomes harder.

2. the particular database may influence the difficulty. For example, in the CMU PIE database [118], images at different poses were captured at exactly the same time, which means that expression is always matched. In the FERET database, the images are not taken at the same time, which may yield expression variations in images.

3. the number of hand-labeled keypoints required.

As can be seen from Table 5.1, our algorithm outperforms all other approaches except the algorithm of Prince *et al.* [104]. However, when producing the better results, this algorithm must assume that the pose was known and it required 21 hand-labeled keypoints. The computational time of this algorithm is also believed higher than ours, this topic will be further considered in future work.

Throughout the experiments of this chapter, we use only the *elementary* POEM based representation. Chapter 4 has seen the clear advantages of *learned* POEM based presentation, notably the high discriminative power. We strongly believe that integrating the *learned* POEM features in the proposed model will yield higher-quality results for face recognition across poses.

5.4 Conclusions

In this chapter, we first present a survey of algorithms for face recognition across poses. By this study, we have shown that the statistical approaches are particularly attractive because of both their low computational complexity and the fact that they are suitable for our context. Based on this, we proposed a novel pose invariant face recognition model centered on modeling how face patches change in appearance as the viewpoint varies. We also propose to combine different local feature descriptors, comprising our POEM features, instead of using pixel-based appearance. By carefully designed experiments,

- once again, the strength of the POEM-based representation for face recognition is shown : it performs reliably when pose angle is about 40°.
- we proved that the proposed statistical model performs very reliably over a wide range of pose variations, even when the probe pose is unknown.
- we proved the advantages of our model : we obtained the high recognition rates without requiring perfectly detected landmarks.

By this work, we also accelerated our previous system [127] by using the low complexity POEM features instead of the Gabor filter features. Future work involves integrating the *learned* POEM descriptor into the proposed model for the goal of both accelerating the process and boosting the performance.

Chapitre 6

Patch-based Similarity HMMs : modeling face configural information for an improved classifier

Up to now, we have considered all three main stages in the face recognition pipeline : the retina filter in preprocessing step, the POEM algorithm (*elementary* or *learned*) for feature extraction, and the statistical model as classifier. By these algorithms, not only two greatest challenges, the variations of pose and illumination, but also other difficulties problem for face recognition have been well solved : our algorithms have seen to be robust to expression variations (see results on Fb probe set in Table 4.2), aging (refer to results on Dup1 & Dup2 probe sets in Table 4.2), etc. [1]

The motivation for the work of this chapter is the face geometric property which is important for distinguishing different faces. Our contribution here is to propose an *original methodology* to model the relations between face components. By this, we provide an improved classifier.

This chapter first details the motivation for this work. We then discuss related work in Section 6.2, describe the general proposed framework in Section 6.3. Experimental results are presented in Section 6.4. Finally conclusions are given in Section 6.5.

1. Of course, we consider that these variations are not very important, for example images are taken in sessions within two year apart as in the FERET database. On the contrary, when these variations are really serious, e.g. the aging difference is more than 10 years, a *specific* face recognition must be designed.

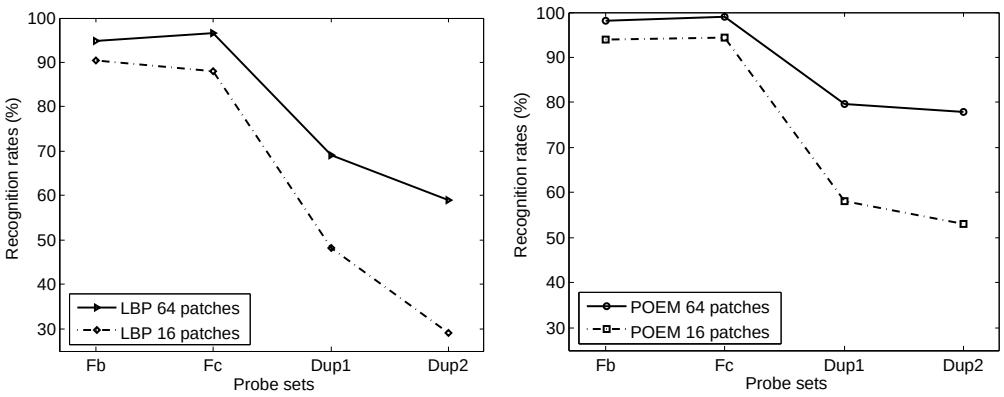
6.1 Motivation

Besides featural information, configural information or relation between features is very important for distinguishing faces. Featural-versus-configural processing for faces has been addressed by many studies, including behavioral [28], neuropsychological [140], electrophysiological [110], and neuroimaging studies [68]. It was shown in [109] that just one feature (such as the eyes or, notably, the eyebrows) can be enough for us to recognize (many) famous faces. However, when features on the top half of one face are combined with the bottom half of another face, the two distinct identities are difficult to recognize [139] or at least reaction times for recognition are increased (see Figure 6.1). The holistic context seems to affect how individual features are processed. When the two halves of the face are misaligned, presumably disrupting normal holistic processing, the two identities are easily recognized. These results suggest that when taken alone, features are sometimes sufficient for facial recognition. Although feature processing is important for facial recognition, these results suggest that configural processing is at least as important.

Figure 6.1 – Try to name the famous faces depicted in the two halves of the left image. Now try the right image. Subjects find it much more difficult to perform this task when the halves are aligned (left) compared to misaligned halves (right), presumably because holistic processing interacts (and in this case, interferes) with feature-based processing. The two individuals shown here are Woody Allen and Oprah Winfrey [119].

Similar to humain, computer vision systems should use configural information for recognizing faces. Most of earlier face recognition algorithms are geometry-based methods which involve precise measurements of geometric attributes such as inter-eye distance, width of mouth, and length of nose. With respect to the classical holistic approaches such as PCA, LDA, researchers try to give them more spatial information of original image by presenting two-dimensional PCA and two-dimensional LDA. The basic idea of EGBM, one of the most common local feature-based method, is to construct a flexible geometrical model over the local features. In the recent face representation methods such as LBP-HS, LGBP-HS and even POEM-HS, dividing images into patches in the construction is for integrating the spatial information into the descriptor. This step plays a very important

role for the recognition performance. When the number of patches decreases from 64 (8 for each of horizontal and vertical directions) to 16, or patch size increases four times, the recognition rates decrease importantly (see Figure 6.2).



(a) (b)

Figure 6.2 – Importance of spatial information when using different histogram-based representations for recognition performance : (a) LBP-HS ; (b) POEM-HS.

Thus, incorporating configural information into face recognition system is an important issue. This is the motivation for the work of this chapter. We propose a novel face recognition framework which exploits the advantage of the well-known technique Hidden Markov Models (HMMs) for modeling spatial relations between face components. Note, our contribution is neither the use of HMMs for face recognition nor HMMs itself, but a completely *novel way* of exploiting the potential advantage of HMMs. This novel strategy results in several desirable advantages including the low complexity, high discriminative power, etc. Although in this chapter we mainly adress and evaluate its performance for face recognition, the proposed algorithm is potential in other applications.

6.2 Related literature

Since our work relates to HMMs, in this section we first briefly discuss about HMM and then detail its applications to face recognition.

6.2.1 Hidden Markov Models

HMM, a set of stochastic models, is used to characterize statistical properties of a signal. It has been successfully used in speech and character recognition applications for

many years and is now being applied to face recognition. HMM consists of two interrelated processes : (1) an underlying, unobservable Markov chain with a finite number of states, a state transition probability matrix and an initial state probability distribution and (2) a set of probability density functions associated with each state. Figure 6.3 illustrates the structure of a simple HMM. The elements of a HMM are :

- N, the number of states in the model.
- M, the number of different observation symbols.
- $S = \{S_1, S_2, ..., S_n\}$ is the finite set of possible hidden states. The state of the model at the time t is given by $q_t \in S : 1 \leq t \leq T$, where T is the length of the observation sequence.
- $\mathbf{A} = \{a_{ij}\}$ is the state transition probability matrix, where

$$a_{ij} = P[q_{t+1} = S_j | q_t = S_i], 1 \leq i, j \leq N, \tag{6.2.1}$$

  with

$$0 \leq a_{ij} \leq 1, \sum_{j=1}^{N} a_{ij} = 1, 1 \leq i \leq N. \tag{6.2.2}$$

- $\mathbf{B} = \{b_{ij}\}$ is the emission probability matrix, indicating the probability of a specified symbol being emitted given that the system is in a particular state, that is :

$$b_{ij} = P[O_t = k | q_t = S_j], \tag{6.2.3}$$

  with $1 \leq i, j \leq N$ and O_t it the observation symbol at time t.
- $\Pi = \{\pi_i\}$ is the initial state probability distribution, that is,

$$\pi_i = P[q_1 = S_i], \tag{6.2.4}$$

  with $\pi_i \geq 0$ and $\sum_{i=1}^{N} \pi_i = 1$.

An HMM is therefore be succinctly defined by the triplet

$$\lambda = \{A, B, \Pi\} \tag{6.2.5}$$

HMMs are typically used to address three problems :

1. *Evaluation.* Given a model λ and a sequence of observations O, what is the probability that O was generated by the model λ, that is, $P(O|\lambda)$.
2. *Decoding.* Given a model λ and a sequence of observations O, what is the hidden state sequence $q*$ most likely to have produced O, that is, $q* = argmax_q[P(q|\lambda, O)]$.
3. *Parameter estimation.* Given an observation sequence O, what model λ is most likely to have produced O.

For further information on HMMs, see the excellent survey [105].

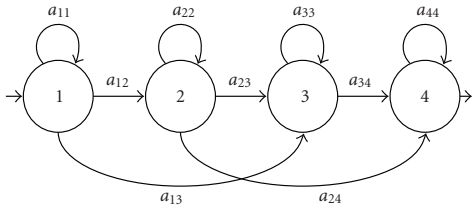


Figure 6.3 – Simple left-to-right HMM.

6.2.2 Using HMMs for face recognition

Given its properties and applications, it is easy to understand that researchers in domain try to use HMM for modeling the spatial relations between face components. Typically, for each individual to be recognized, one distinct HMM is used to model the spatial relation between face components of the considered person, and then an unknown face image is recognized as the subject k if it is the most likely generated by k's HMM. These two steps, building models and recognizing, correspond to the problems *Parameter estimation* and *Evaluation* of HMMs, respectively.

• Concerning the first step, except [66], other existing HMM-based methods [111, 112, 89, 88, 14] need a number of training samples to ensure the reliability of parameter estimation. Furthermore, these systems suffer from the drawback of high complexity :

1. Since these algorithms build a distinct HMM for each individual, they need as much HMMs as individuals to be recognized which can be thousands.
2. The dimension of observation sequence is relatively high (see Section 6.3.4).

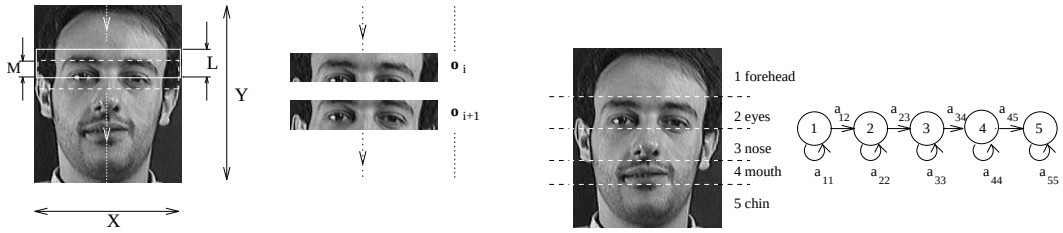
Maybe due to their high complexity, *any HMM-based approaches* so far can report results on large database containing thousands of individuals, such as the FERET database.

• Concerning the second stage, all HMM-based approaches use the Maximum Likelihood criterion to recognize the unknown face image.

6.2.3 Existing HMM-based face recognition approaches

We now discuss in more detail the existing HMM-based face recognition systems. The key question of systems using the architecture described above becomes : "How to encode a face image as an observation sequence and which type of HMM is practical for face recognition ?"

In the pioneer work [111], Samaria proposed the first-order luminance-based 1D HMM for face recognition where the facial image is top-bottom scanned to form an observation sequence. The observation sequence is composed of vectors that represent the consecutive horizontal strips. Figure 6.4(a) shows the way that Samaria generates



(a) (b)

Figure 6.4 – (a) Face strip extraction for generating observation sequence ; (b) Modeling face with a five state left-to-right HMM [111].

the observation sequence from a face image while Figure 6.4(b) shows his five state left-to-right HMM associated with facial regions. Nefian and Hayes [89] modified the approach with some complexity reduction by using Discrete Cosine Transform (DCT) coefficients to generate the observation sequence whereas Bicego *et al.* [14] used discrete wavelet transform (DWT) coefficients of sub windows generated by a raster scan of the image.

As HMMs are one dimensional in nature, a variety of approaches have been adopted to try to represent the two dimensional structure of face images. The pseudo 2D HMM (2D-PHMM) proposed by Samaria [112], extension of the 1D-HMM, is a 1D-HMM composed of super states to model the sequence of columns in the image, in which each super state is a 1D-HMM itself modeling the blocks within the columns. In this work, Samaria added a *marker block* at the end of each line in the image, and introduced an additional *end-of-line state* at the end of each horizontal HMM as shown in Figure 6.5(a). The end-of-line states were allowed two possible transitions : one backs to the beginning of the same row of states, and one transits to the next row of states. Samaria also considered a pseudo 2D-HMM that involved removing end-of-maker blocks as shown in Figure 6.5(b). As results, Samaria reported similar recognition rates for both models, but using a really small database of 10 subjects. Nefian and Hayes [88] proposed a novel structure that they referred to *embedded* HMM. Whilst the super states are also used to model two-dimensional data along one direction, this model differs from a true two-dimensional HMM (2D-HMM) since the transitions between the states in different super states are not allowed, as shown in Figure 6.5(c). The authors also argued that the embedded HMM reduces complexity of the 2D-HMM proposed by Samaria.

Another approach is proposed by Le and Li [66] where a face image is modeled by two 1D discrete HMMs, one for observations in the vertical direction and one for the horizontal direction. This method is also the unique so far, to the best of our knowledge, which is applicable in one sample circumstances. Two factors contribute to the feasibility and effectiveness of their method. First, they generated a large collection of observation vectors from each image, in both vertical and horizontal directions, thus en-

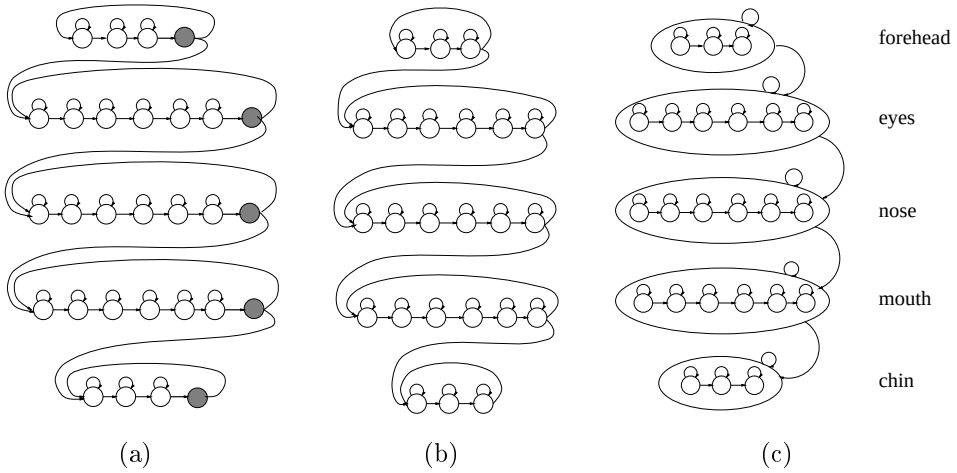


Figure 6.5 – Face modeling with : (a) one-dimensional HMM with end-of-line-states [112] ; (b) one-dimenstional HMM without end-of-line states (also called pseudo 2D HMM) ; (c) embedded HMM [88].

larging the training set. Second, the Haar wavelet transform was applied to the image to lessen the dimension of the observation vectors and improve the robustness performance. Their experiment results tested on the frontal view AR face database show that the proposed method outperforms the PCA, LDA, and local feature analysis approaches. An alternative approach is the low-complexity 2D HMM (LC 2D-HMM) which consists of a rectangular constellation of states, where both vertical and horizontal transitions are supported. The complexity of the LC 2D-HMM is considerably lower than that of the 2D-PHMM, but recognition accuracy is lower as results.

More recent approaches that involve finding the more sophisticated and thus more reliable types of HMMs for face recognition are presented in [26], called Maximum Confidence Hidden Markov Model (MC-HMM), and in [95], called Structural Hidden Markov Model (SHMM).

6.3 Proposed model : PS-HMMs

Similar to other HMM-based face recognition approaches, the three principal steps are *Observation sequence generating*, *HMM modeling & training*, and *Recognition*.

6.3.1 Observation sequence generating

Instead of using the facial features extracted from one image to form an observation sequence as in other systems, the similarities between facial features extracted from

patches of an image pair are exploited in our system. The procedure is carried out as follows (see Figure 6.6) :

• Each face image is first divided into overlapping horizontal strips of height j pixels where the strips overlap by p pixels [2]. Each horizontal strip is then vertically segmented into patches of width k pixels, with overlap of q. For an image of width w and height h, there will be approximately $n = (h/(j-p)+1) * (w/(k-q)+1)$ patches.

• For an image pair of equal size, I^a and I^b, we have two sequences of patches $\{P_1^a, P_2^a, ..., P_n^a\}$ and $\{P_1^b, P_2^b, ..., P_n^b\}$. The observation sequence O^{ab} is generated by concatenating the similarity values between two corresponding patches of images in a determined order : $O^{ab} = \{d(P_i^a, P_i^b)\}, i = 1..n$ where $d(.,.)$ is any distance function between two patches of any facial features (*e.g.*, $L2$ distance of their gray level or χ^2 distance of their POEM histograms). Figure 6.6 illustrates the generation of an observation sequence when faces are scanned from left to right and from top to bottom. This refers to *hor scan strategy*. On the contrary, the top-to-bottom left-to-right scan order refers to *ver scan strategy*. A priori, $O_{ver}^{ab} = (O_{hor}^{ab})'$.

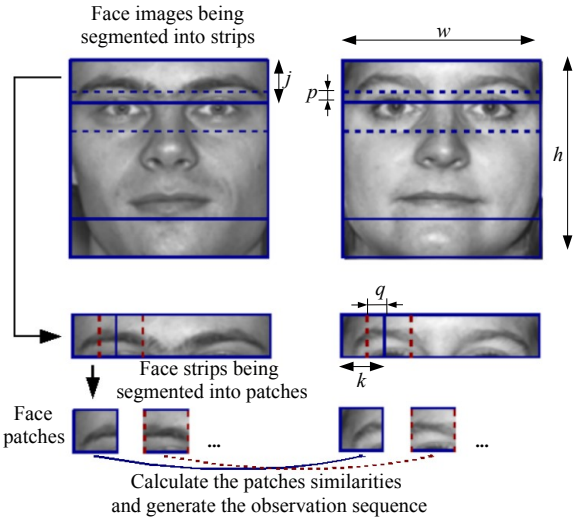


Figure 6.6 – Observation sequence generation.

6.3.2 Modeling and training

Given an offline training database, we build two HMMs, λ^{same} and λ^{dif}. The first HMM λ^{same} models the intra-subject similarities of the dataset, and is used to provide the probability that two face images come from the same subject. The second HMM λ^{dif}

2. $p = 0$ means that there is not overlap.

models the extra-subject similarities and provides the probability that two images are from different identities.

This training step is done just one time in advance using an *independent* database, meaning that *once the models are trained, we can use them in the recognition stage for any database at anytime*. Therefore, even though the training dataset requires several images for each individual and has to be large, we can collect it easily.

For each image pair, I^a and I^b, an observation sequence O^{ab} is generated. All obtained observation sequences are then divided into two groups, intrapersonal $\mathbf{O}^{int}$ and extrapersonal $\mathbf{O}^{ext}$, as follows : $O^{ab} \subset \mathbf{O}^{int}$ if I^a and I^b are images of the same person and $O^{ab} \subset \mathbf{O}^{ext}$ otherwise.

Finally, these two sets of observation sequences $\mathbf{O}^{int}$ & $\mathbf{O}^{ext}$ are used to train the two HMMs, λ^{same} & λ^{dif} respectively.

6.3.3 Recognition

Consider the gallery set containing N images $I^{g_1}, I^{g_2}, ..., I^{g_N}$ of N subjects. Given a probe image I^p, for each image in the gallery set I^{g_j}, we compute the probability that the probe and gallery images come from the same subject. For each image pair, I^p and I^{g_j}, an observation sequence O^{pg_j} is formed. The probabilities of this observation sequence generated by the two trained HMMs are then used to recognize the probe image. Two possible techniques can be applied : Maximum Likelihood (ML) and Maximum *a Posteriori* (MAP) criterion.

• In the case of using ML criterion, only the first HMM λ^{same} is used. The probe image is recognized as the face I^{g_k} if :

$$P(O^{pg_k}|\lambda^{same}) = max_{j=1..N} P(O^{pg_j}|\lambda^{same}), \tag{6.3.1}$$

• In the case of using MAP criterion, both trained HMMs are used. The *a posteriori* probability is calculated using Bayes rule :

$$P(same|O^{pg_j}) = \frac{P(O^{pg_j}|\lambda^{same})P(same)}{P(O^{pg_j}|\lambda^{same})P(same) + P(O^{pg_j}|\lambda^{dif}P(dif)},$$

where $P(same)$ and $P(dif)$ are class priors.

The probe image is recognized as the face I^{g_k} if :

$$P(same|O^{pg_k}) = max_{j=1..N} P(same|O^{pg_j}), \tag{6.3.2}$$

6.3.4 Novelties and Properties

Roughly speaking, all existing HMM-based approaches are built in a similar vein. On the contrary, our models, referred to Patch-based Similarity HMMs (PS-HMM), are designed in a ***completely novel way***. There are novelties at multiple levels :

1. **Patch level :** Whist all other HMM based approaches build models directly upon the extracted facial features, our PS-HMMs use *the similarities between patches* from images and in particular we model the spatial relations between face components using a database containing various individuals.
2. **System level :** Our method requires only *two HMMs for the whole system* for a given *scan strategy* compared to the needing of one distinct HMM for each subject to be recognized as in [89, 95, 111].
3. **Process level :** the training database can be entirely separated from the gallery and test images, thus the problem of only *one reference sample* is automatically solved.

These novelties give our algorithm several desirable ***advantages*** :

1. ***Robustness to personal variations.*** The richer the information for learning available is, the better obtained models are. Using a *whole offline database containing various individuals* for training gives our PS-HMMs the capability of modeling different types of personal variation, such as illumination, expression,.... On the contrary, in order to ensure the capability of modeling these variations, other systems require rich information about all individuals to be recognized, which is impractical for many applications.
2. ***The low complexity.***

   (i) Among HMMs' parameters, the dimension of observation sequence, denoted as d_O, effects critically the complexity in both terms of time and space requirements. In our models, when using one similarity measurement between patches, $d_O = n_p$ where n_p is the number of patches per image. In other systems, $d_O = t \times n_p$ where t is the dimension of patch descriptor, e.g. $t = 24$ in [95].

   (ii) Given a *scan strategy*, we need only *two PS-HMMs* for the *whole system*. Other approaches require as much HMMs as individuals to be recognized which is 1000+ in the FERET database.

Moreover, while other systems can use only Maximum Likelihood (ML) criterion during the recognition stage, our algorithm has the capability of using both Maximum Likelihood and Maximum *a Posteriori* (MAP) criteria. Our experimental results indeed show that MAP gives better performance.

The clear difference between PS-HMMs and the algorithm presented in chapter 5 is that PS-HMMs consider the configural information of face in the addition. The model presented in chapter 5 does not consider the relationship between face patch.

6.4 Experiments and Discussions

In order to show the efficiency of the novel model, we evaluate the performance of PS-HMMs for face recognition on three datasets : FERET, AR and LFW.

6.4.1 Experiment setup

Three parameters need considering in our method :

1. *HMM type and parameter estimation algorithm.* Due to a lack of time, we consider here the very simple HMMs : the 1D 4 state ergodic models where 2 GMMs are used per state [3]. Also, we use the most popular criterion for estimating the HMMs' parameters, the ML criterion [105] (this is different from the ML used for recognition in Section 6.3.3). Whilst simple models, the reported results are interesting since they prove the strength of our framework.
2. *scan strategy.* In our previous work [130], we use the *hor scan strategy* (see Figure 6.6) which yields PS-HMMs *hor*. We also consider here the *ver scan strategy* which yields PS-HMMs *ver*. There are in total 4 PS-HMMs : $\lambda_{hor}^{same}, \lambda_{hor}^{dif}, \lambda_{ver}^{same}$, and λ_{ver}^{dif}. Note that the runtime is not 2 times slower since O_{ver} is very fast to compute from O_{hor} : $O_{ver} = (O_{hor})'$.
3. *similarity function* between two patches. This relates directly to considered facial features. Up to now, POEM features have seen to be more robust than others. However, in order to show the efficiency of PS-HMMs for various face representations, we consider here both the *elementary* LBP and POEM features. Although *learned* features, e.g. WPCA-LBP and WPCA-POEM (Chapter 4) are more discriminative and compact, we only consider here the *elementary* features to see *how far* integrating configural face information into system enhances its performance.

Class priors $P(same) = P(dif) = 0.5$, meaning that any prior information about classes is available. Face areas are divided into 64 non-overlapping [4].

In order to respect the protocole associated with these datasets, we report the results on theFERET and AR databases using the PS-HMMs which are built upon 736 training samples of the FERET database, whereas the results on View 2 of the LFW database are calculated using the PS-HMMs which are trained using its View 1. As mentioned above,

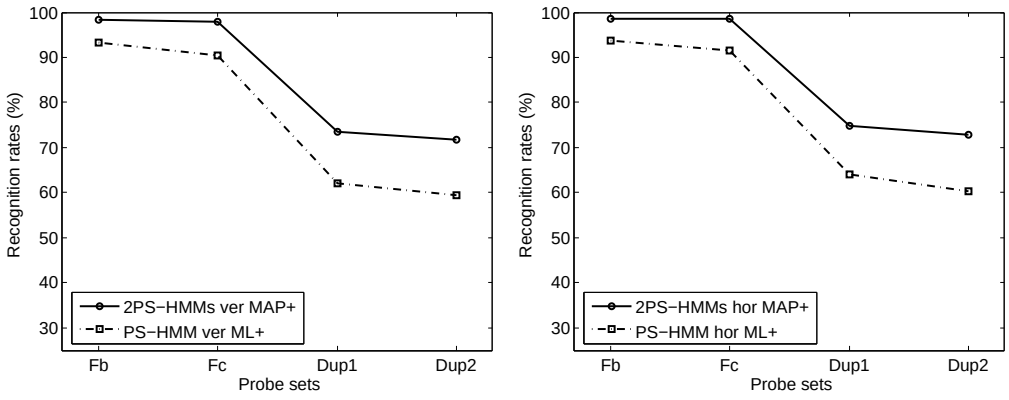
3. In our previous work [130], 3 GMMs per state are used, but we find that 2 GMMs are sufficient and even give better results.

4. All existing HMM-based face recognition algorithms, even our previous work [127], divide face area into overlapping regions since this gives richer information about face or more face spatial relations are modeled. However, our main goal here is to show how far face recognition system is improved when the spatial information is integrated. That is the reason why we divide here face area into 64 non-overlapping patches, as with the LBP-HS and POEM-HS algorithms. Since $d_O = 64$ is quite small, we use only 4 states per HMM, instead of 5 as commonly used.

we wish to *show the performance of PS-HMMs when using various facial features*, we conduct the experiments on the FERET and AR databases using LBP features whereas the experiments carried out on the LFW database using the POEM features.

6.4.2 ML vs. MAP criterion

This section compares the performance of two recognition criteria : ML and MAP. Using either PS-HMMs *hor* or PS-HMMs *ver*, recognition rates on the four FERET probe sets are calculated using either ML or MAP. When using ML, only λ_{hor}^{same} & λ_{ver}^{same} are used. These methods are denoted as *PS-HMM hor ML+* and *PS-HMM ver ML+*, respectively. When using MAP, all $\lambda_{hor}^{same}, \lambda_{hor}^{dif}, \lambda_{hor}^{same}$, and λ_{hor}^{dif} are used. These methods are denoted as *2PS-HMMs hor MAP+* and *2PS-HMMs ver MAP+*.



(a) (b)

Figure 6.7 – Performance of ML and MAP criteria on the FERET database when using different *scan strategies* : (a) : PS-HMMs *ver* ; (b) : PS-HMMs *hor*.

Both Figures 6.7(a) and (b) show that MAP-based PS-HMMs outperform ML-based PS-HMMs. This is easy to understand since using both within-subject and inter-subject similarities (MAP) provides richer information than using only intra-personal similarities. Note, due to their way of using HMMs, other HMM-based face recognition systems can use only ML criterion for recognition task. This again explains why our algorithm is more discriminative compared to others.

6.4.3 PS-HMMs *hor* vs. PS-HMMs *ver*

Figure 6.7 and Table 6.1 show that PS-HMMs *hor* perform more effectively than PS-HMMs *ver*. This suggests us that the spatial relation between face component along the horizontal direction is more important than that along the vertical direction. Related

to the symmetric property of face, the spatial relation in horizontal direction gives richer information.

We try now to combine both PS-HMMs *hor* and PS-HMMs *ver*. Given an image pair, I_i & I_j, two observations along two directions are obtained $O^{ij}_{ver}\&(O^{ij}_{hor})$. Then, the *a posteriori* probabilities determining whether these two observations belong to the *same* class are calculated as in Equation 6.3.3. Resulting probabilities are denoted as $P(same|O^{ij}_{hor})$ and $P(same|O^{ij}_{ver})$. The final probability determining whether two images I_i & I_j come from the same individual is calculated as the sum of two obtained probabilities : $P(same|O^{ij}_{comb}) = P(same|O^{ij}_{hor}) + P(same|O^{ij}_{ver})$. Table 6.1 presenting the high performance of this "**Combined**" shows that combining spatial information in both vertical and horizontal directions enhances the discriminative power of face recognition algorithms.

6.4.4 Results on the FERET database

For the first time, a HMM-based face recognition algorithm reports the performance on the **whole** FERET database. Table 6.1 shows clearly the efficiency of the proposed method.

Methods	Fb	Fc	Dup1	Dup2
PCA	85.0	65.0	44.0	22.0
LBP*	92.5	51	58	47.0
Retina + LBP*	94.9	96.5	67.2	59.0
LBP 2PS-HMMs hor MAP+	**98.5**	**98.5**	**74.8**	**72.9**
LBP 2PS-HMMs ver MAP+	**98.4**	**98**	**73.5**	**71.7**
Combined	**98.7**	**99**	**78.8**	**74.6**

Tableau 6.1 – Performance of PS-HMMs on the FERET database.

6.4.5 Results on the AR database

By the previously trained PS-HMMs, we report results on 12 probe sets, from "AR-02" to "AR-13", using 126 "AR-01" pictures as gallery.

Figure 6.8 clearly shows the efficiency of incorporating spatial relations between face components into a face recognition method, e.g. the LBP algorithm here. The very high recognition rates obtained across different conditions prove that our PS-HMMs are really *independent* of the gallery and test sets.

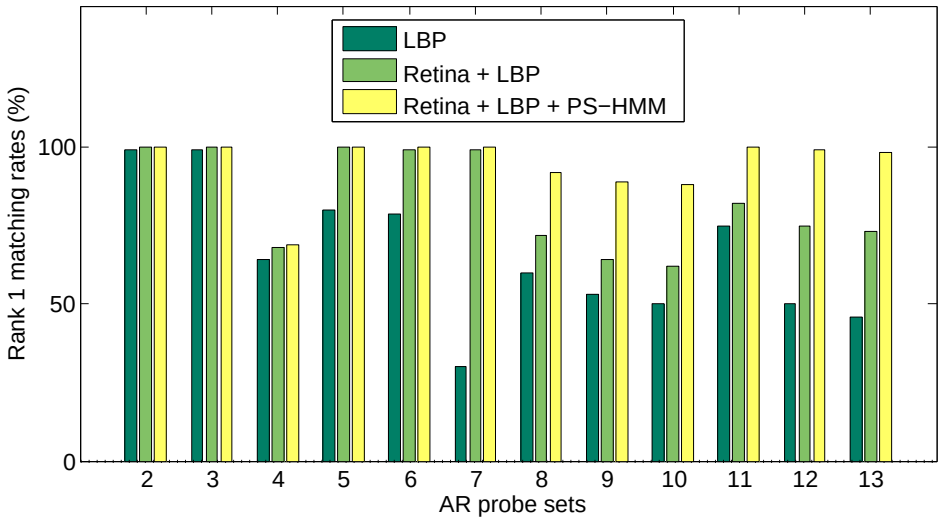


Figure 6.8 – Effect of configural information integrated into the LBP method.

Algorithm in [66] is the unique HMM approach reporting results on the AR database. When compared to this, our algorithm provides better results. Moreover, in [66], 126 HMMs are trained using the AR database itself. In the contrast, our PS-HMMs are independent of the gallery and test sets, meaning that once the models are constructed, we can use it for other database for anytime.

The excellent improvements on probe sets "AR08–AR13" are very interesting to discuss (AR08–AR10 : faces with sunglasses, AR11-AR13 : faces with scarf). When objects are partially occluded, we believe that relations between visible details becomes more important.

6.4.6 Results on the LFW database

We duplicate the same experiments on the LFW database. In order to respect the strict protocole associated with this dataset, we have to retrain our PS-HMMs using images on View 1 and then we report results on View 2. In these tests, we use χ^2 distance between two POEM histograms as patch similarity measure. Other parameters, including HMM type, number of states, etc. are kept.

The task now is to determine whether two faces come from *Same* or *Dif*. Given two images I_i, I_j, we have 2 probabilities. In terms of notation, we should write $P(O^{ij}|\lambda^{same}_{comb})$ where $P(O^{ij}|\lambda^{same}_{comb}) = P(O^{ij_{hor}}|\lambda^{same}_{hor}) + P(O^{ij_{ver}}|\lambda^{same}_{ver})$, but for short, we refer this to P^s_{ij}. Thus we have P^s_{ij} and P^d_{ij}. There are two possible ways to use these values for determining the identity of I_i and I_j :

1. I_i & I_j are of the same person if $P^s_{ij} >= P^d_{ij}$ or of the different individuals if

$P_{ij}^s < P_{ij}^d$.

2. In order to take benefit of the training set of LFW database, the following procedure is carried out : for all image pairs, we calculate $t = P_{ij}^s - P_{ij}^d$. For every test split, an optimal threshold on t which separates the best the 9 training splits is chosen and then used for calculating the classification rate on the considered test split.

The second procedure provides slightly better results than the first one. As results, using PS-HMMs on the LFW dataset, we have only small improvements of around 2.6% points compared to the pure POEM-HS method. Despite this small improvement, it is not very disappointed since this real-world database is very challenging : many personal variations can appear at the same time on one image, including pose, expression, lighting, and even bad image quality, such as compression and blur, etc.

In the similar vein, the performance of the probabilistic model presented in chapter 5 is evaluated for face verification on this dataset. Given two images I_i, I_j, using this probabilistic model, we obtain a probability. Based on probabilities obtained on *View 1*, we find an optimal threshold and then report the final result on *View 2*. Similarly, applying the probabilistic model results in only small improvement (the verification accuracy increases from 75.2 to 76.1).

6.4.7 Discussions

• From results obtained on the three considered databases, it seems that smaller the database is, higher the improvements are (PS-HMMs enhance considerably the recognition rates on AR08-13 sets). When more individuals are considered, the geometric properties of a face become less discriminative : there will be several faces whose geometric properties, such as distance between eyes, are similar. One potential solution is the consideration of "finer scale" relations. By "finer scale" relations, we suggest to divide the face into smaller regions, and then exploit the relations between these small regions. We can also select some important regions and model them using different *scan strategies*, etc.

• The significant improvements of PS-HMMs on "occluded" probe sets (see results on "AR08–AR13" sets – Figure 6.8) suggest that when faces are partially occluded, the configural information of visible part should be carefully considered.

• Although our HMMs are of lower complexity than others, they are still relatively slow. In our experiments, it takes several hours to report the recognition rates on the FERET set. However, due to their improvement of recognition rates, we believe that PS-HMMs are still interesting, at least they give a completely novel way of modeling the configural information. At the time of writing this dissertation, the very classic PS-HMMs are tested with Matlab implementations. We hope that lower complexity PS-HMMs will

be found in future work.

• Due to a lack of time, we use only the ML criterion for estimating the parameters. Researchers in different domains, notably in speech recognition and machine learning, have studied and proposed several algorithms which are more discriminative. In [57], Juang and Katagiri proposed a method which refers to *Minimum Classification Error (MCE)* for discriminative learning the HMM parameters. With respect to the considered applications, speech recognition, they argued that using MCE can reduce about 50% the false classification rate when compared to the ML parameter estimation method. By integrating MCE criterion into optical character recognition algorithm, Huo *et al.* [52] also obtained higher performance. This parameter estimation algorithm will be considered in future work and higher performance will be hopefully obtained.

• In fact, by using the idea of PS-HMMs, we have modeled the face symmetric properties and used them for head tilt estimation, and the obtained results are very encouraging.

- Given a *training* face I_p at pose p, by dividing this image along the vertical direction, we obtained two halves : I_l and I_r. We then flipped the I_r on the vertical axis and obtained the I_{rf}. By applying the procedure described in Figure 6.6 on two *images* I_l and I_{rf}, we obtain an observation sequence O_p. Theoretically, HMMs being trained by such observation sequence O_p can model well face symmetric properties which are useful cues for estimating the head tilt.
- To validate this idea, we used 900 images of 9 different poses of FERET database and followed the above procedure to generate 900 observation sequences which are used to train 9 HMMs. In other words, for a particular pose, we used 100 observation sequences to train one HMM. Therefore, we had 9 HMMs, one per pose.
- Given a *test* face image, its pose is estimated as p if its observation sequence is the most likely generated by p's HMM. In our algorithm, we have to assume that the *head direction* is known, meaning that we did not distinguish a face at poses $-p°$ and $+p°$. With this assume, we estimated the pose of 900 other images of FERET database. As results, we obtained the *accuracy* of 5.6°.

6.5 Conclusions

The work of this chapter is inspired by the importance of the relation between face components for face recognition. By exploiting the similarity between patches from an image pair, we proposed a low complexity yet efficient architecture for face recognition with a single reference image which requires only *two* HMMs for the whole system (*two* models for each *scan strategy*). By employing only the simplest HMMs, we train our

models using the FERET training dataset, and then use them to recognize the faces in the FERET database itself and the *unseen* AR database. Experimental results show that the proposed algorithm provides excellent performance in both cases. Impressively, due to its low complexity, this is *the first time* that a HMM based face recognition approach has been tested on the *entire FERET database*. Future work involves finding and integrating higher performing HMMs in order to enhance the framework performance.

Chapitre 7

POEM for Interest Region Description

Finding correspondences between two images of the same scene or object is part of many computer vision applications, such as object recognition [34, 74], wide baseline matching [124], etc. In such systems, the fundamental idea is to first detect image regions that are covariant to a class of transformations. The most valuable property of an interest region *detector* is its repeatability, i.e. whether it reliably finds the same interest points under different viewing conditions. Next, for each detected region, an invariant *descriptor* is built and used to match interest regions between images. Two important properties of local descriptors are the information content and the invariance. These properties determine the distinctiveness and the robustness of the descriptor. Unfortunately, in general, the more the description is invariant the less information it conveys. As the information content, the descriptors should capture the shape and the texture information of a local structure. A good local descriptor should also be able to tolerate viewpoint & illumination changes, noise, image blur, image compression, and small perspective distortions. As already pointed out in Chapter 4, the POEM feature captures both the shape and texture information, and at the same time is robust to exterior changes thanks to its multi-scale self-similarity based structure. This suggests that POEM could also be a good candidature for interest region description. This will be proved in this chapter. For this purpose, several additional techniques making the POEM descriptor more feasible in the novel task are also presented. Note, within considered applications, there are no difference between the meaning of terms used "interest point detector" and "interest region detector" : once the interest points are selected, local descriptors are built on their neighborhoods which correspond to interest regions.

We briefly discuss related work in Section 7.1 and detail additional techniques in Section 7.2. Experimental setups and results are presented in Section 7.3. Conclusions are given in Section 7.4.

7.1 Related literature

A wide variety of detectors and descriptors have already been proposed in the literature [43, 74, 83, 7]. Also, detailed comparisons and evaluations on benchmarking datasets have been performed [82, 83, 132]. Focusing on region description, we discuss briefly the most representative detector techniques while a more in-depth consideration of local descriptors is given.

Interest Region/Points Detectors. The most widely used points detector is probably the Harris corner detector [43], proposed back in 1988, based on the eigenvalues of the second-moment matrix. However, Harris corners are not scale-invariant. Lindeberg introduced the concept of automatic scale selection [70]. This allows to detect interest points in an image, each with their own characteristic scale. He experimented the determinant of the Hessian matrix as well as the Laplacian (which corresponds to the trace of the Hessian matrix) to detect blob-like structures. Mikolajczyk and Schmid refined this method, creating robust and scale-invariant feature detectors with high repeatability, and they presented Harris-Laplace and Hessian-Laplace detectors [83]. They used a scale-adapted Harris measure or the determinant of the Hessian matrix to select the location, and the Laplacian to select the scale. Focusing on speed, Lowe [74] approximated the Laplacian of Gaussian (LoG) by a Difference of Gaussians (DoG) filter.

Feature Descriptors. An even larger variety of feature descriptors has been proposed. In earlier algorithms, the frequency content of image is exploited, and the Fourier and Gabor transforms are commonly used. Many descriptors are differential, such as steerable filters [35], and complex filters [6]. The idea is to approximate a point neighborhood using a set of image derivatives computed up to a given order.

The most commonly used descriptors are distribution-based, in which they use histograms to represent different characteristics of appearance or shape. The intensity-domain spin image [65] is a 2D histogram where the dimensions are the distance from the center point and the intensity value. Lowe [74] proposed a scale invariant feature transform (SIFT), which combines a scale invariant region detector and a descriptor based on the gradient distribution in the detected regions. The SIFT descriptor is a 3D histogram of gradient locations and orientations. Shape context [10] implements the same idea and is very similar to the SIFT descriptor. In [82], Mikolajczyk and Schmid proposed one extension of the SIFT descriptor, called GLOH (Gradient Location and Orientation Histogram) and one extension of the shape context. The GLOH descriptor replaces the Cartesian location grid used by the SIFT with a log-polar one, and applies PCA to reduce the size of the descriptor. In a similar vein, they used log-polar location grid for the extension of the shape context presented, which is a 3D histogram of edge point locations and orientations (original shape context was computed only for edge point locations and not for orientations [10]). Some other descriptors proposed in the literature

are geodesic-intensity histogram (GIH), PCA-SIFT (using the image gradient patch and applying PCA to reduce the size of descriptor [60]), and moment invariants.

Also, there exist several recent comparative studies on region descriptors [82, 85]. Almost without an exception, the best results are reported for SIFT-based descriptors. After these comparisons, some interesting descriptors have been also developed. In [7], Bay et. al proposed to build the SURF (Speeded Up Robust Features) descriptor on the strengths of the leading existing detectors and descriptors. They used a Hessian matrix-based measure for the detector and Haar wavelet responses for the descriptor. By relying on integral images for image convolutions, computation time is significantly reduced. In [45], Heikkila et. al proposed Center-symmetric local binary patterns (CS-LBP) which is argued to take the advantages of both SIFT and LBP descriptors.

More recently, interesting descriptor learning scheme has been proposed [133, 19]. The authors break up the descriptor extraction process into a number of modules and put these together in different combinations. Furthermore, learning is used to optimize the choice of parameters for each candidate descriptor algorithm, to reduce the dimension of descriptors or to make the descriptors much discriminative. In the presented scheme, the two main modules for *elementary* low-level feature extraction are T-block and S-block, and E-block is used for learning the descriptor. T-block takes the pixels from the image patch and transforms them to produce a vector of non-linear filters responses at each pixel, and S-block spatially accumulates the resulting filter vectors to form the descriptor. Many of these combinations give rise to published descriptors such as SIFT, GLOH, PCA-SIFT, but many are untested. The learning scheme is interesting since learning makes the descriptors more discriminative and compact.

Inspired by the success of the proposed POEM descriptor for face recognition, this chapter aims at evaluating its performance for interest region description, a more extensive field in computer vision. For this purpose, we present a dominant orientation estimation method which is suitable for the POEM descriptor in order to make the descriptor invariant to rotation.

7.2 POEM for interest region description

7.2.1 POEM descriptor construction

Once an interest region is selected and geometrically normalized, we use the same procedure as described in Section 4.2 – Chapter 4 in order to extract POEM feature for each pixel of region. Then, POEM images (each corresponds to one orientation) are divided into non-overlapping patches. Histogram is calculated for each patch. Descriptor of interest region is built by concatenating all histograms estimated over all patches. This

procedure is illustrated in Figure 7.1 where Accumulated EMIs are images obtained by first dividing the gradient image (Figure 7.1(c)) into m images through the gradient direction of pixel (m is the number of discretized orientations) and then replacing the pixel value by the sum of all values within the cell, centered on the current pixel.

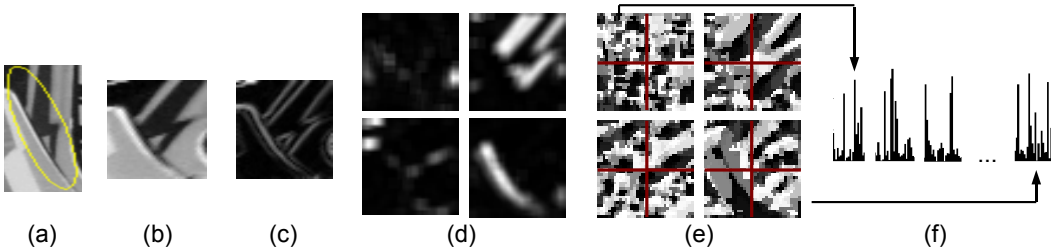


Figure 7.1 – POEM-HS descriptor construction. (a) An elliptical image region detected. (b) Region after affine normalization. (c) Gradient image. (d) Accumulated EMIs. (e) POEM images. (f) POEM-HS descriptor computed for the normalized region.

7.2.2 Invariance to rotation

The invariance to rotation can be obtained either by computing rotation invariant descriptors or by normalizing the region with respect to rotation and then computing the description. Only few rotation invariant photometric measures can be found, such as in [35], while the second approach is quite common in the literature [74, 83, 7]. The basic idea is to estimate one dominant orientation in a local neighborhood and then to normalize the neighborhood to rotation. If the estimation is uncorrect, the computed description is useless, as it is not rotation invariant and can not be correctly matched. The estimation of dominant orientations is often based on the phase of the gradient [74].

Suppose that the angle space (0-π for the unsigned representation and 0-2π for the signed one) is evenly spread with the range of $\Delta\theta$, meaning that the estimated orientation would be one of the following values $\{0, \Delta\theta, 2\Delta\theta, ..., \pi\}$ (or 2π). For example, in [74] $\Delta\theta = 10°$, the estimated orientation would be $10°, 20°, 30°, ...$ Thus we consider $\Delta\theta$ as the *accuracy* of the estimated orientation. Once the gradient is calculated for all pixels within the interest region, the dominant orientation is typically estimated by first calculating the sum of all gradient magnitudes within a sliding orientation window covering an angle of $\Delta\theta$ (weighted window can also be used). Then the longest vector over all windows lends its direction to the interest point. In this algorithm, the sliding window width is equal to the *accuracy* of estimated orientation $\Delta\theta$.

However, this does not give stable performance in the case of POEM descriptor. By varying the size of the sliding window, we find that the best parameter is $\frac{\pi}{m}$ for the

unsigned representation or $\frac{2\pi}{m}$ for the signed one, where m is the bin number (Section 4.2). This is easy to understand since our POEM feature construction is based on the gradient magnitudes accumulated within a sliding window covering this angle. Note that rotation invariance is not necessary for many applications, such as appearance-based face recognition where face images are aligned.

7.3 Experiments

7.3.1 Experiment setup

7.3.1.1 Database and protocole

We use the well-known image matching protocol proposed by Mikolajczyk and Schmid [82]. The protocol is available on the Internet together with the test data [1]. We use the five test image sequences which are designed to test the robustness to :

1. view point changes (Wall, Graffiti),
2. illumination changes (Leuven),
3. Jpeg compression artifacts (Ubc)
4. and image blur (Bike)

Figure 7.2 shows the test image sequences. For each one, we have 2 image pairs by matching the first to other two images. The evaluation criterion is based on the number of correct and false matches between a pair of images. As in [82], we use the threshold-based matching strategy, where two regions are matched if the $L2$ distance between their descriptors is below a threshold. The image regions are considered to be a correspondence if there is at least a 50% overlap between the regions projected onto the same image [82]. A descriptor can have several matches and several of them may be correct. The results are presented with *recall* versus *1-precision* :

$$recall = \frac{\#correct\ matches}{\#correspondences}, \tag{7.3.1}$$

$$1 - precision = \frac{\#false\ matches}{\#all\ matches}, \tag{7.3.2}$$

where the $\#correspondences$ stands for the ground truth number of matching regions between the images. The curves are obtained by varying the distance threshold and a perfect descriptor would give a *recall* equal to 1 for any *precision*.

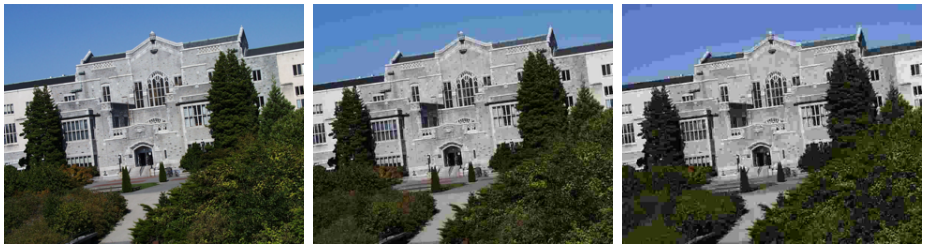
1. http ://www.robots.ox.ac.uk/ vgg/research/affine/detectors.html

(a) Graf 1,4,5

(b) Wall 1,4,5

(c) Leuven 1,5,6

(d) Bike 1,4,5



(e) Ubc 1,4,5

Figure 7.2 – Test images : (a) Graf – viewpoint change ; (b) Wall – viewpoint change ; (c) Leuven – illumination change ; (d) Bike – image blur ; (e) Ubc – JPEG compression artifacts. In the experiments, we use only gray images.

7.3.1.2 Detectors

Two detectors are used : Hessian-Affine (HesAff) and Harris-Affine (HarAff). HesAff detects blob-like structures while HarAff looks for corner-like structures. Both detectors output elliptic regions of varying size determined by the detection scale. Before computing the descriptors, the detected regions are mapped to a circular region of constant radius to obtain scale and affine invariance (c.f. Figures 7.1(a, b)). In the experiments, we fix the normalized region size to 41 $\times$ 41 pixels, as in [82].

7.3.1.3 Parameters

- **POEM parameters**

This section is just to recall the POEM parameters : unsigned representation with 3 bins, built on 10x10 pixel blocks and 7x7 pixel cells, 6 neighbors for each considered cell. Concerning the number of divided patches (refer to Figure 7.1), by varying this number {1,4,9}, we find that the best case is 4 patches per image (2 $\times$ 2).

- **Descriptor normalization**

In this evaluation protocole, the $L2$ (Euclidean) distance is used. We have tested two normalization methods, *Two-step* & *Square-root*, and found that the *Two-step* algorithm performs slightly better (remind that for face recognition, χ^2 distance without normalization performs the best). As details, the histogram (per patch) is first normalized to unit length, the elements larger than a threshold, which is set to 0.2, as suggested in [74, 45] are set to the threshold. The histogram is re-normalized to unit length. All normalized histograms are finally concatenated to form the POEM-HS descriptor.

7.3.2 Results

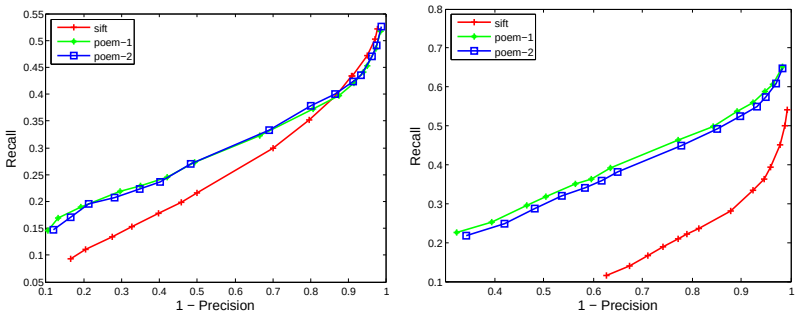
This section compares our POEM descriptor with the SIFT algorithm which is *the reference* in domain. As mentioned above, when estimated orientation is uncorrect, the computed description is useless, as it is not rotation invariant and can not be correctly matched. Thus, for a fair comparison, with respect to performance of POEM descriptor, the dominant orientation is always estimated using the procedure described above and then is used to rotate the considered region.

Figures 7.3, 7.4, 7.5, and 7.6 show the performance of POEM descriptor with two different *accuracy* of estimated orientation. The *accuracies* of *POEM-1* and *POEM-2* are 10° and 15°, respectively. We can see from these figures that POEM outperforms SIFT in almost cases with both detectors. POEM is more robust to lighting changes than SIFT (c.f. Figure 7.4) with considerable improvement of correct matches (10–20%,

depending on test pairs and detectors). When viewpoint angle is 50° (Graf, Wall 5 – Figure 7.2), compared to SIFT, POEM gives a significant improvement (c.f. Figures 7.3 (b,d,f,h)).

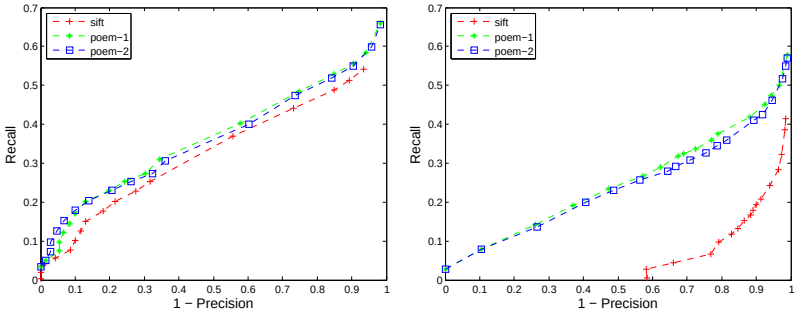
7.4 Conclusion

This chapter aims at evaluating the performance of POEM descriptor for image matching, a problem with a wide variety range of applications. For this purpose, we have proposed a method for estimating the dominant orientation which makes POEM robust to rotation. The very encouraging results obtained with a benchmarking protocole, being better than SIFT, suggest that POEM is also a good candidate for this novel application. As future work, we will apply learning technique on POEM descriptor, as in Chapter 4 for example, in order to produce a more discriminative and more compact descriptor for describing the interest regions.



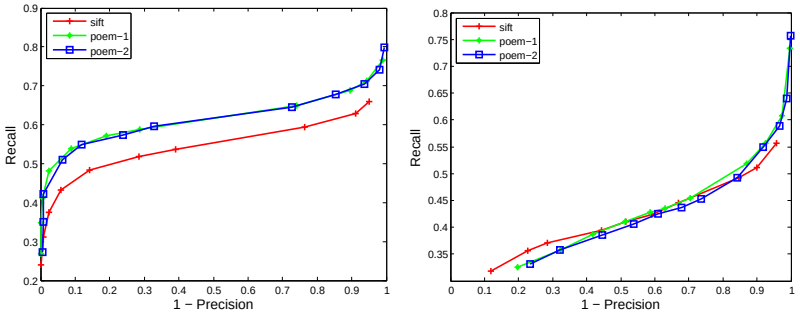
(a) Graf 1|4 + HesAff

(b) Graf 1|5 + HesAff



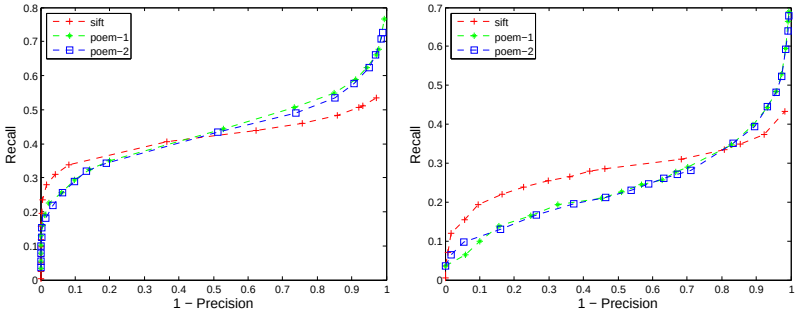
(c) Graf 1|4 + HarAff

(d) Graf 1|5 + HarAff



(e) Wall 1|4 + HesAff

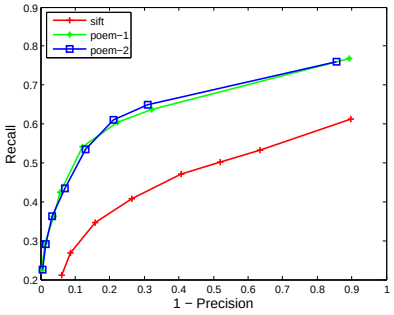
(f) Wall 1|5 + HesAff



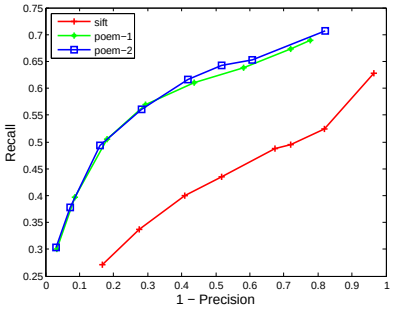
(g) Wall 1|4 + HarAff

(h) Wall 1|5 + HarAff

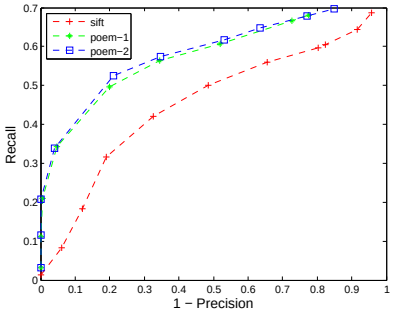
Figure 7.3 – Robustness to viewpoint changes.



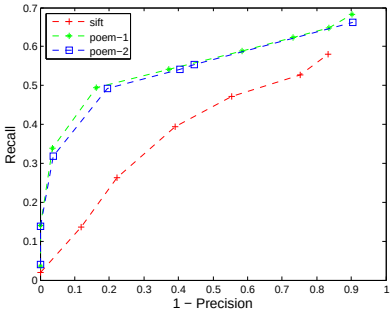
(a) Leuven 1|5 + HesAff



(b) Leuven 1|6 + HesAff

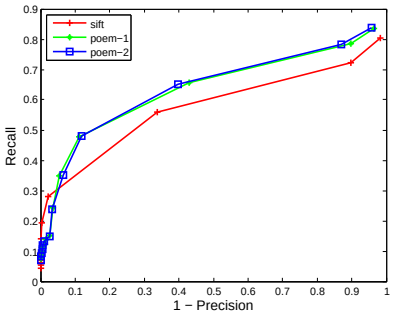


(c) Leuven 1|5 + HarAff

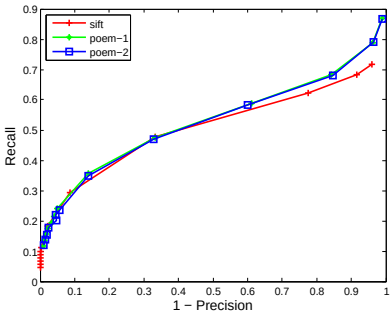


(d) Leuven 1|6 + HarAff

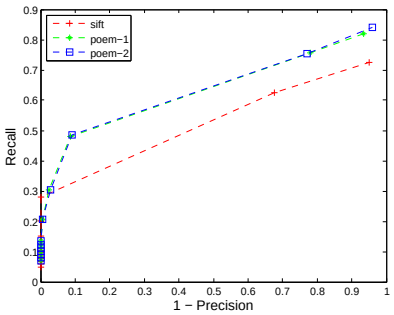
Figure 7.4 – Robustness to lighting changes.



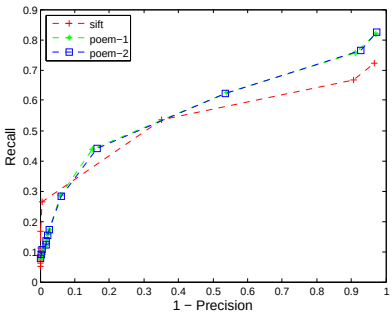
(a) Ubc 1|4 + HesAff



(b) Ubc 1|5 + HesAff

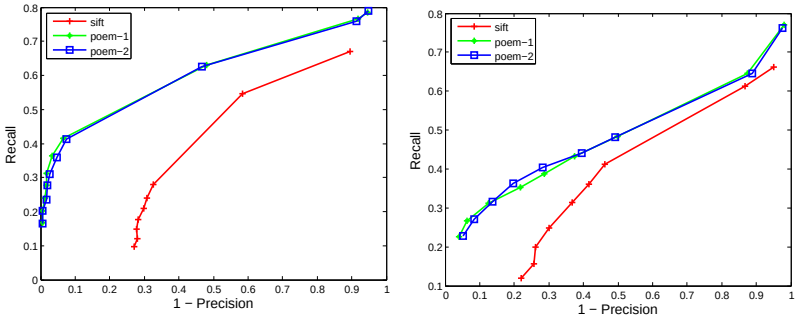


(c) Ubc 1|4 + HarAff



(d) Ubc 1|5 + HarAff

Figure 7.5 – Robustness to compression artifacts.



(a) Bike 1|4 + HesAff (b) Bike 1|5 + HesAff

(c) Bike 1|4 + HarAff (d) Bike 1|5 + HarAff

Figure 7.6 – Robustness to blur.

Chapitre 8

Conclusions and Future work

8.1 Conclusions

This doctoral dissertation proposed several approaches for real-time unconstrained face recognition in one sample circumstance from three aspects. The first aspect is to propose a retina filter which removes directly *illumination* variations. The second aspect is to build very *efficient features* for facial representation. The third aspect is to develop a *pose robust* face recognition algorithm.

It is worth noting that this work is a part of BIORAFALE project funded by Vesalis company. However, due to the complexity of administrative procedure, the "real" data of project has not obtained yet. That is why the efficiency of the proposed algorithms in this work is not evaluated yet on the real data of project. This will be considered in future work.

Here are some of the key contributions made in the thesis :

• Whilst existing illumination normalisation algorithms are based on the retinex theory, which aims at estimating the illumination from the input image, our algorithm, described in Chapter 3, removes directly illumination variations presented in facial images. We first identified which retina cells can act as an illumination removal and then proposed some additional techniques to enhance its performance. Our approach does not only remove the illumination variations but also reduces noise, enhances image contours and contrast. Our algorithm is *very fast* to compute since its complexity is linear (it is *faster than all other* considered algorithms). With respect to preprocessing for face recognition, our approach *outperforms the state-of-the-art* algorithms.

• Presented in Chapter 4 are novel POEM features for object representation. Using these features, we build two descriptors : the *elementary* POEM-HS and the *learned*

PCA-POEM descriptors. Both are of high discriminative power and robust to variations of pose, illumination, expression and aging. POEM-HS outperforms all other *elementary* features whilst the PCA-POEM algorithm *reaches the state-of-the-art* results for both constrained and un-constrained face recognition tasks. Importantly, our descriptors are *very efficient* both to compute and to store in memory : our algorithm is *faster* than all other considered systems.

• In Chapter 5, the second challenge for face recognition, pose variations, is addressed by a local statistical model, centered on modeling how face patches change in appearance as the viewpoint varies. We also proposed to combine different local feature descriptors, comprising our POEM features. Without the requirement of perfectly detected landmarks, our algorithm performs *very reliably* over a wide range of pose variations, even when the probe pose is unknown.

• For the first time, a face recognition algorithm based on HMMs which can perform well on a large database is presented in Chapter 6. Our models, called Patch-based Similarity HMMs (PS-HMMs), presenting a novel way of modeling the spatial relations between face components have several desirable properties. They are discriminative, suitable for face recognition in one sample circumstances. Our algorithm is also much lower complex than all other HMM-based approaches.

• The strength of POEM descriptor was again proved in Chapter 7 by showing its high performance for image matching task (it outperforms significantly and systematically the SIFT algorithm). Also, a dominant orientation estimation algorithm is proposed in order to make POEM descriptor robust to rotation.

• For other face recognition based applications, such as classification of album photos where the task is to determine whether two faces come from the same person or not (as in the LFW dataset), the LDA-WPCA-POEM algorithm presented in Chapter 4 can be a good choice.

• As mentioned in the first part of this dissertation, this work is funded by the BIORAFALE project which aims at recognizing the hooligan in football stade. However, due to the administrative problem, we have not yet the

8.2 Future work

Unconstrained face recognition from one sample reference can be expanded in a multitude of ways. We just list in the follows some potential avenues to explore in the context of the proposed approaches :

• In Chapter 3, by carefully designed experiments, the optimal parameters of our

retina filter have been identified, but only for images acquired in constrained context. Therefore, the parameters of *DoG* filter need to be tuned automatically. To this end, a quality metric should be used.

• The POEM features in Chapter 4 have successfully encoded the relationship between local edge distributions across different orientations. It turns out that the relations between edge distributions of different orientations can be used as useful information of object. Alternatively, the information about gradient phase should be explored.

• Algorithms in Chapters 5, 6 and 7 were evaluated using the *elementary* POEM descriptor. As a natural issue, we will evaluate their performance when using *learned* descriptors.

• In Chapter 6, we used the very simple HMMs (1D ergodic) associated with the ML parameter estimation criterion. Future work involves finding more performing HMMs associated with more discriminative parameter estimation algorithm.

• In future work, we also aim at enhancing the performance of the head pose estimation algorithm presented in Section 6.4.7 – Chapter 6.

• The goal of the Biorafale project is to recognize faces from surveillance camera, given still faces (maybe one per person). Within such context, face recognition is typically solved as a problem of image-image matching : each of the probe images is compared to reference image in some ways, in order to obtain matching scores, upon which the recognition decision will be carried out. Such approaches have ignored completely the temporal information of probe sequence which is really useful for the recognition task. The PS-HMMs can be easily extended for exploiting the temporal information of probe sequence. Figure 6.6-Chapter 6 shows how to generate an observation sequence from two images. We now briefly describe how to generate an observation sequence from a target *image* and *a sequence* of probe images. Suppose I_r the target image, $\{I_{p_1}, I_{p_2}, ..., I_{p_n}\}$ images of probe sequence. For each image in the probe sequence, we couple it with the target image and use the procedure shown in Figure 6.6 to obtain the observation vectors O^{rp_i}. These are then concatenated into a single observation sequence $O^{rp} = \{O^{rp_i}\}$. HMMs being learnt using such observation sequence O^{rp} can model well the spatial and temporal information of probe sequence. Within this approach, we suppose that $\{I_{p_1}, I_{p_2}, ..., I_{p_n}\}$ are images of the same individual, or the tracking results are available. Validating this idea will be considered in future work.

Bibliographie

[1] Y. Adini, Y. Moses, and S. Ullman. Face recognition : The problem of compensating for changes in illumination directions. *IEEE Trans. PAMI*, 19 :721–732, 1997.

[2] T. Ahonen, A. Hadid, and M. Pietikainen. Face recognition with local binary patterns. In *ECCV*, pages 469–481, 2004.

[3] A.B. Ashraf, S. Lucey, and T. Chen. Learning patch correspondences for improved viewpoint invariant face recognition. In *CVPR*, pages 1–8, June 2008.

[4] M.S. Bartlett. *Face Image Analysis by Unsupervised Learning*. Kluwer Academic Publishers, 2001.

[5] R. Basri and D. W. Jacobs. Lambertian reflectance and linear subspaces. *IEEE Trans. PAMI*, 25(2) :218–233, February 2003.

[6] Adam Baumberg. Reliable feature matching across widely separated views. In *CVPR*, pages 1774–1781, 2000.

[7] H. Bay, T. Tuytelaars, and L. Van Gool. Surf : Speeded up robust features. In *ECCV*, pages 404–417, 2006.

[8] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman. Eigenfaces vs. fisherfaces : Recognition using class specific linear projection. *IEEE Trans. PAMI*, 1997.

[9] P.N. Belhumeur and D.J. Kriegman. What is the set of images of an object under all possible illumination conditions ? *Int. J. Comput. Vision*, 28(3) :245–260, 1998.

[10] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Trans. PAMI*, 24(4) :509–522, 2002.

[11] A. Benoit. *The human visual system as a complete solution for image processing*. PhD thesis, INPG, Grenoble, France, 2007.

[12] D. Beymer. Face recognition under varying pose. In *CVPR*, pages 756–761, 1994.

[13] D. Beymer and T. Poggio. Face recognition from one example view. In *ICCV*, pages 500–507, 1995.

[14] M. Bicego, U. Castellani, and V. Murino. Using hidden markov models and wavelets for face recognition. In *ICIAP*. IEEE, 2003.

[15] M. Bicego, A. Lagorio, E. Grosso, and M. Tistarelli. On the use of sift features for face authentication. In *CVPR Workshop*, 2006.

[16] V. Blanz, P. Grother, P. J. Phillips, and T. Vetter. Face recognition based on frontal views generated from non-frontal images. In *CVPR*, pages 454–461, 2005.

[17] V. Blanz, S. Romdhani, and T. Vetter. Face identification across different poses and illuminations with a 3d morphable model. In *FG*. IEEE, 2002.

[18] Volker Blanz and Thomas Vetter. Face recognition based on fitting a 3d morphable model. *IEEE Trans. PAMI*, 25(9) :1063–1074, 2003.

[19] M. Brown, G. Hua, and S. Winder. Discriminative learning of local image descriptors. *IEEE Trans. PAMI*, (1), 2010.

[20] R. Brunelli and T. Poggio. Face recognition : features versus templates. *IEEE Trans. PAMI*, 15(10) :1042–1052, 1993.

[21] Z. Cao, Q. Yin, X. Tang, and Jian S. Face recognition with learning-based descriptor. In *CVPR*, 2010.

[22] R. Chellappa, C. L. Wilson, and S. Sirohey. Human and machine recognition of faces : a survey. In *Proceedings of the IEEE*, volume 83, pages 705–741, 1995.

[23] H.F. Chen, P.N. Belhumeur, and D.W. Jacobs. In search of illumination invariants. In *CVPR*, 2000.

[24] S. Chen. Making flda applicable to face recognition with one sample per person. *Pattern Recognition*, 37(7) :1553–1555, 2004.

[25] T. Chen, W. Yin, X.S. Zhou, D. Comaniciu, and T.S. Huang. Total variation models for variable lighting face recognition. *IEEE Trans. PAMI*, 28(9) :1519–1524, 2006.

[26] J.T. Chien and C.P. Liao. Maximum confidence hidden markov modeling for face recognition. *IEEE Trans. PAMI*, 30 :606–616, 2008.

[27] W. Choi, S. Tse, K. Wong, and K. Lam. Simplified gabor wavelets for human face recognition. *Pattern Recognition*, 41(3) :1186–1199, 2008.

[28] S. M. Collishaw and G. J. Hole. Featural and configurational processes in the recognition of faces of different familiarity. *Perception*, 29(8) :893–909, 2000.

[29] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models-their training and application. *Computer Vision and Image Understanding*, 61(1) :38–59, 1995.

[30] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, pages 886–893, 2005.

[31] J. G. Daugman. High confidence visual recognition of persons by a test of statistical independence. *IEEE Trans. PAMI*, 15(11) :1148–1161, Nov 1993.

[32] Weihong Deng, Jiani Hu, and Jun Guo. Gabor-eigen-whiten-cosine : A robust scheme for face recognition. In *AMFG*, pages 336–349. 2005.

[33] B. Duc, S. Fischer, and J. Bigun. Face authentication with gabor information on deformable graphs. *IEEE Trans. on Image Processing*, 1999.

[34] V. Ferrari, T. Tuytelaars, and Luc Van Gool. Simultaneous object recognition and segmentation by image exploration. In *ECCV*, pages 40–54, 2004.

[35] W.T. Freeman and E.H. Adelson. The design and use of steerable filters. *IEEE Trans. PAMI*, 13(9) :891–906, 1991.

[36] Y. Gao and Y. Qi. Robust visual similarity retrieval in single model face databases. *Pattern Recognition*, 38(7) :1009–1020, 2005.

[37] C. Garcia and M. Delakis. Convolutional face finder : A neural architecture for fast and robust face detection. *IEEE Trans. PAMI*, 26(11) :1408–1423, 2004.

[38] A.S. Georghiades and P.N. Belhumeur. From few to many : illumination cone models for face recognition under variable lighting and pose. *IEEE Trans. PAMI*, 23 :643–660, 2001.

[39] D. Gonzalez-Jimenez and J. L. Alba-Castro. Toward pose-invariant 2-d face recognition through point distribution models and facial symmetry. *IEEE Trans. on Information Forensics and Security*, 2(3) :413–429, 2007.

[40] T. Gritti, C. Shan, V. Jeanne, and R. Braspenning. Local features based facial expression recognition with face registration errors. In *FG*, pages 1–8, 2008.

[41] R. Gross, I. Matthews, and S. Baker. Appearance-based face recognition and light-fields. *IEEE Trans. PAMI*, 26(4) :449–465, 2004.

[42] M. Guillaumin, J. Verbeek, and C. Schmid. Is that you ? metric learning approaches for face identification. In *ICCV*, 2009.

[43] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey Vision Conference*, 1988.

[44] X. He, S. Yan, Y. Hu, P. Niyogi, and H.J. Zhang. Face recognition using laplacianfaces. *IEEE Trans. PAMI*, 27(3) :328–340, 2005.

[45] M. Heikkila, M. Pietikainen, and C. Schmid. Description of interest regions with local binary patterns. *Pattern Recognition*, 42 :452–436, 2009.

[46] B. Heisele, P. Ho, J. Wu, and T. Poggio. Face recognition : component-based versus global approaches. *Computer Vision and Image Understanding*, 91(1-2) :6–21, 2003.

[47] R. Hietmeyer. Biometric identification promises fast and secure processings of airline passengers. *The Int. Civil Aviation Organization J.*, 55(9) :10–11, 2000.

[48] G. B. Huang, V. Jain, and E. Learned-Miller. Unsupervised joint alignment of complex images. In *ICCV*, pages 1–8, 2007.

[49] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild : A database for studying face recognition in unconstrained environments. Technical report, University of Massachusetts, Amherst, 2007.

[50] G.B. Huang, M.J. Jones, and E. Learned-Miller. Lfw results using a combined nowak plus merl recognizer. In *Faces in Real-Life Images Workshop at ECCV*, 2008.

[51] Wiesel T. Hubel, D. Functional architecture of macaque monkey visual cortex. In *Royal Society on Biology 198*, 1978.

[52] Q. Huo, Y. Ge, and Z.D. Feng. High performance chinese ocr based on gabor features, discriminative feature extraction and model training. In *ICASSP*, 2001.

[53] A. Jain, K. Nandakumar, and A. Ross. Score normalization in multimodal biometric systems. *Pattern Recognition*, 38(12) :2270–2285, December 2005.

[54] X. Jiang, E. Rosen, T. Zeffiro, J. Vanmeter, V. Blanz, and M. Riesenhuber. Evaluation of a shape-based model of human face discrimination using fmri and behavioral techniques. *Neuron*, 50(1) :159–172, April 2006.

[55] D.J. Jobson, Z. Rahman, and G.A. Woodell. A multiscale retinex for ridging the gap between color images and the human observation of scenes. *IEEE Trans. On Image Processing*, 6 :965–976, 1997.

[56] M. J. Jones and P. Viola. Face recognition using boosted local features. Technical report, MERL, 2003.

[57] B. H. Juang and S. Katagiri. Discriminative learning for minimum error classification. *IEEE Trans. on Signal Processing*, 40(12) :3043–3054, Dec 1992.

[58] T. Kanade. *Computer recognition of Human faces.* 1977.

[59] T. Kanade and A. Yamada. Multi-subregion based probabilistic approach toward pose-invariant face recognition. In *IEEE ICRA*, volume 2, pages 954–959, 2003.

[60] Y. Ke and R. Sukthankar. Pca-sift : A more distinctive representation for local image descriptors. In *CVPR*, 2004.

[61] N. Kumar, A.C. Berg, P.N. Belhumeur, and S.K. Nayar. Attribute and simile classifiers for face verification. In *ICCV*, 2009.

[62] M. Lades, J. C. Vorbruggen, J. Buhmann, J. Lange, C. von der Malsburg, R. P. Wurtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE Trans. on Computers*, 42(3) :300–311, 1993.

[63] E.H. Land and J.J. McCANN. Lightness and retinex theory. *J. Opt. Soc. Am.*, 61(1) :1–11, January 1971.

[64] A. Lanitis. Automatic face identification system using flexible appearance models. *Image and Vision Computing*, 13(5) :393–401, 1995.

[65] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *IEEE Trans. PAMI*, 27(8) :1265–1278, 2005.

[66] H.S. Le and H. Li. Recognizing frontal face images using hidden markov models with one training image per person. In *ICPR*. IEEE, 2004.

[67] K.C. Lee, J. Ho, and D. J. Kriegman. Acquiring linear subspaces for face recognition under variable lighting. *IEEE Trans. PAMI*, 27(5) :684–698, May 2005.

[68] Y. Lerner, T. Hendler, D. Ben-Bashat, M. Harel, and R. Malach. A hierarchical axis of object processing stages in the human visual cortex. *Cereb. Cortex*, 11(4) :287–297, 2001.

[69] Y. Li, S. Gong, and H. Liddell. Constructing facial identity surfaces for recognition. In *CVPR*, 2003.

[70] T. Lindeberg. Feature detection with automatic scale selection. *Int. J. of Computer Vision*, (2), 1998.

[71] C. Liu. Gabor-based kernel pca with fractional power polynomial models for face recognition. *IEEE Trans. PAMI*, 26(5) :572–581, 2004.

[72] C. Liu and H. Wechsler. Evolutionary pursuit and its application to face recognition. *IEEE Trans. PAMI*, 22(6) :570–582, 2000.

[73] C. Liu and H. Wechsler. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. *IEEE Trans. Image Processing*, 11 :467–476, 2002.

[74] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *Int Journal of Computer Vision*, 60(2) :91–110, 2004.

[75] M. Behrmann M. Moscovitch, G. Winocur. Facing the issue : New research shows that the brain processes faces and objects in separate brain systems. *Journal of Cognitive Neuroscience*, 1997.

[76] B. S. Manjunath, R. Chellappa, and C. von der Malsburg. A feature based approach to face recognition. In *CVPR*, 1992.

[77] A. M. Martinez. Recognizing imprecisely localized, partially occluded, and expression variant faces from a single sample per class. *IEEE Trans. PAMI*, 24(6) :748–763, 2002.

[78] A. M. Martinez and A. C. Kak. Pca versus lda. *IEEE Trans. PAMI*, 23(2) :228–233, 2001.

[79] A. Mellakh. *Reconnaissance des visages en conditions dégradées.* PhD thesis, l'Institut National des Télécommunications, 2009.

[80] E. Meyers and L. Wolf. Using biologically inspired features for face processing. *Int Journal of Computer Vision*, 76 :93–104, 2008.

[81] L. Meylan, D. Alleysson, and S. Susstrunk. Model of retinal local adaptation for the tone mapping of color filter array images. *Journal of the Optical Society of America A*, 24 :2807–2816, 2007.

[82] K. Mikolajczyk and C. Schmid. A performance evaluation of local descriptors. *IEEE Tran. PAMI*, 27(10) :1615–1630, 2005.

[83] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Van Gool. A comparison of affine region detectors. *Int J. of Computer Vision*, 65(1-2) :43–72, 2005.

[84] B. Moghaddam and A. Pentland. Probabilistic visual learning for object representation. *IEEE Trans. PAMI*, 19(7) :696–710, 1997.

[85] P. Moreels and P. Perona. Evaluation of features detectors and descriptors based on 3d objects. *Int. J. of Computer Vision*, 73(3) :263–284, 2007.

[86] Y. Moses, Y. Adini, and S. Ullman. Face recognition : The problem of compensating for changes in illumination direction. In *ECCV*. 1994.

[87] Y. Mu, S. Yan, Y. Liu, T. Huang, and B. Zhou. Discriminative local binary patterns for human detection in personal album. In *CVPR*, 2008.

[88] A. V. Nefian. An embedded hmm-based approach for face detection and recognition. In *ICASSP*. IEEE, 1999.

[89] A.V. Nefian and M.H. Hayes III. Hidden markov models for face recognition. In *ICASSP*, pages 2721–2724. IEEE, 1998.

[90] H. Nguyen, L. Bai, and L. Shen. Local gabor binary pattern whitened pca : A novel approach for face recognition from single image per person. In *ICB*. 2009.

[91] E. Nowak and F. Jurie. Learning visual similarity measures for comparing never seen objects. In *CVPR*, pages 1–8, 2007.

[92] E. Nowak, F. Jurie, and B. Triggs. Sampling strategies for bag-of-features image classification. In *ECCV*, pages 490–503, 2006.

[93] T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Trans. PAMI*, 24(7) :971–987, 2002.

[94] Field D. Olshausen, B. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381, 1996.

[95] D. Bouchaffra P. Nicholl, A. Amira and R.H. Perrott. A statistical multiresolution approach for face recognition using structural hidden markov models. *EURASIP Journal on Advances in Signal Processing*, 2008.

[96] Y.K. Park, S.L. Park, and J.K. Kim. Retinex method based on adaptive smoothing for illumination invariant face recognition. *Signal Processing*, 88(8) :1929–1945, 2008.

[97] A. Pentland, B. Moghaddam, and T. Starner. View-based and modular eigenfaces for face recognition. In *CVPR*, 1994.

[98] V. Perlibakas. Distance measures for pca-based face recognition. *Pattern Recognition Letters*, 25(6) :711–724, 2004.

[99] J. Phillips, H.Moon, and S.A. Rizvi et al. The feret evaluation methodology for face-recognition algorithms. *IEEE Trans. PAMI*, 22 :1090–1104, 2000.

[100] J.P. Phillips. Support vector machines applied to face recognition. In *NIPS*, volume 11, pages 803–809, 1999.

[101] N. Pinto, J. J. di Carlo, and D. D. Cox. Establishing good benchmarks and baselines for face recognition. In *Faces in Real Life Images workshop at ECCV08*, 2008.

[102] N. Pinto, J.J. DiCarlo, and D.D. Cox. Establishing good benchmarks and baselines for face recognition. In *Faces in Real-Life Images Workshop in ECCV*, 2008.

[103] N. Pinto, J.J. DiCarlo, and D.D. Cox. How far can you get a modern face recognition test set using only simple features ? In *CVPR*, 2009.

[104] S.J. D. Prince, J.H. Elder, J. Warrell, and F.M. Felisberti. Tied factor analysis for face recognition across large pose differences. *IEEE Trans. PAMI*, 30(6) :970–984, 2008.

[105] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2) :257–286, 2002.

[106] S. Romdhani, V. Blanz, and T. Vetter. Face identification by fitting a 3d morphable model using linear shape and texture error functions. In *ECCV*. 2002.

[107] C. Rosenberger and L. Brun. Similarity-based matching for face authentication. In *ICPR*, pages 1–4, 2008.

[108] H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Trans. PAMI*, 20(1) :23–38, 1998.

[109] J. Sadr, I. Jarudi, and P. Sinha. The role of eyebrows in face recognition. *Perception*, 32(3) :285–293, 2003.

[110] N. Sagiv and S. Bentin. Structural encoding of human and schematic faces : Holistic and part-based processes. *Journal of Cognitive Neuroscience*, 13(7) :937–951, 2001.

[111] F. Samaria. Face segmentation for identification using hidden markov model. In *BMVC*, 1993.

[112] F. Samaria. *Face recognition using hidden Markov Models*. PhD thesis, Cambridge University, 1994.

[113] C. Sanderson, S. Bengio, and Y. Gao. On transforming statistical models for non-frontal face verification. *Pattern Recognition*, 39 :288–302, 2006.

[114] C. Sanderson and B.C. Lovell. Multi-region probabilistic histograms for robust and scalable identity inference. In *ICB*, 2009.

[115] B. Schiele and J.L. Crowley. Recognition without correspondence using multidimensionalreceptive field histograms. *Int. J. Comput. Vision*, 36 :31–50, 2000.

[116] S. Shan, W. Gao, Y. Chang, B. Cao, and P. Yang. Review the strength of gabor features for face recognition from the angle of its robustness to mis-alignment. In *ICPR*, pages 338–341. IEEE, 2004.

[117] S. Shan, P. Yang, X. Chen, and W. Gao. Adaboost gabor fisher classifier for face recognition. In *AMFG*, pages 279–292. 2005.

[118] T. Sim, S. Baker, and M. Bsat. The cmu pose, illumination, and expression database. *IEEE Trans. PAMI*, 25(12) :1615–1618, 2003.

[119] P. Sinha, B. Balas, Y. Ostrovsky, and R. Russell. Face recognition by humans : Nineteen results all computer vision researchers should know about. *Proceedings of the IEEE*, 94(11) :1948–1962, 2006.

[120] Y. Taigman, L. Wolf, and T. Hassner. Multiple one-shots for utilizing class label information. In *British Machine Vision Conference*, 2009.

[121] X. Tan, S. Chen, Z. H. Zhou, and F. Zhang. Face recognition from a single image per person : A survey. *Pattern Recognition*, 39(9) :1725–1745, 2006.

[122] X. Tan and B. Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. In *AMFG*, pages 168–182, 2007.

[123] M. Turk and A. Pentland. Eigenfaces for recognition. *J. Cognitive Neuroscience 3*, pages 71–86, 1991.

[124] T. Tuytelaars and Luc J. Van Gool. Matching widely separated views based on affine invariant regions. *Int J. of Computer Vision*, 59(1) :61–85, 2004.

[125] M. Vasilescu and D. Terzopoulos. Multilinear image analysis for facial recognition. In *ICPR*, 2002.

[126] P. Viola and M.J. Jones. Robust real-time face detection. *Int Journal of Computer Vision*, 57 :137–154, 2004.

[127] Ngoc S. Vu and Alice Caplier. Efficient statistical face recognition across pose using local binary patterns and gabor wavelets. In *BTAS*, pages 44–48. IEEE, 2009.

[128] Ngoc-Son Vu and Alice Caplier. Illumination-robust face recognition using the retina modelling. In *ICIP*. IEEE, 2009.

[129] Ngoc-Son Vu and Alice Caplier. Face recognition with patterns of oriented edge magnitudes. In *ECCV*, 2010.

[130] Ngoc-Son Vu and Alice Caplier. Patch-based similarity hmms for face recognition with a single reference image. In *ICPR*. IEEE, 2010.

[131] H. Wang, S.Z. Li, and Y. Wang. Generalized quotient image. In *CVPR (2)*, pages 498–505, 2004.

[132] S. Winder and M. Brown. Learning local image descriptors. In *CVPR*, 2007.

[133] S. Winder, G. Hua, and M. Brown. Picking the best daisy. In *CVPR*, pages 178–185, 2009.

[134] L. Wiskott, J. M. Fellous, N. Kuiger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Trans. PAMI*, 19(7) :775–779, July 1997.

[135] L Wolf, T Hassner, and Y Taigman. Descriptor based methods in the wild. *Faces in Real-Life Images Workshop in ECCV*, October 2008.

[136] L. Wolf, T. Hassner, and Y. Taigman. Similarity scores based on background samples. In *ACCV*, 2009.

[137] Tan X. and Triggs B. Fusing gabor and lbp feature sets for kernel-based face recognition. In *AMFG*, 2007.

[138] J. Yang, D. Zhang, A.F. Frangi, and J.Y. Yang. Two-dimensional pca : A new approach to appearance-based face representation and recognition. *IEEE Trans. PAMI*, 26(1) :131–137, 2004.

[139] A. W. Young, D. Hellawell, and D. C. Hay. Configurational information in face perception. *Perception*, 16(6) :747–759, 1987.

[140] G. Yovel and B. Duchaine. Specialized face perception mechanisms extract both part and spacing information : Evidence from developmental prosopagnosia. *Journal of Cognitive Neuroscience*, 18(4) :580–593, April 2006.

[141] B. Zhang, S. Shan, X. Chen, and W. Gao. Histogram of gabor phase patterns (hgpp) : A novel object representation approach for face recognition. *IEEE Trans. Image Processing*, 16 :57–68, 2007.

[142] G. Zhang, X. Huang, S. Li, Y. Wang, and X. Wu. Boosting local binary pattern (lbp)-based face recognition. In *Advances in Biometric Person Authentication*, pages 179–186. 2005.

[143] T. Zhang, B. Fang, Y. Yuan, Y.Y. Tang, Z. Shang, D. Li, and F. Lang. Multiscale facial structure representation for face recognition under varying illumination. *Pattern Recognition*, 42(2) :251–258, 2009.

[144] W. Zhang, S. Shan, W. Gao, X. Chen, and H. Zhang. Local gabor binary pattern histogram sequence (lgbphs) : a novel non-statistical model for face representation and recognition. In *ICCV*, volume 1, 2005.

[145] W. Zhao, R. Chellappa, S. Corporation, A. Rosenfeld, and P. J. Phillips. Face recognition : A literature survey. *ACM Surveys*, 2000.

[146] J Zou, Q Ji, and G Nagy. A comparative study of local matching approach for face recognition. *IEEE Trans. Image Processing*, 16(10) :2617–2628, 2007.

Publications associées

Les contributions présentées dans ce manuscrit ont été publiées dans les articles suivants :

1. Conférences internationales

[1] N-.S. Vu, A. Caplier. "Face recognition with Patterns of Oriented Edge Magnitudes", Euro Conf on Computer Vision ECCV 2010, Septembre, Heraklion, Crete.

[2] N-.S.Vu, A. Caplier. "Patch-based similarity HMMs for face recognition with a single reference image", Int Conf on Pattern Recognition ICPR 2010, Istambul.

[3] N-.S. Vu, A. Caplier. "Illumination-robust face recognition using retina modeling", Int Conf on Image Processing ICIP 2009, Cairo.

[4] N-.S. Vu, A. Caplier. "Efficient statistical face recognition across pose with LBP and Gabor wavelet", Int Conf on Biometrics : Theory, Applications and System BTAS 2009, Washington DC.

[5] C. Fiche, P. Ladret, N-.S. Vu. "Blurred face recognition algorithm guided by a no-reference blur metric", SPIE Machine Vision Applications 2010, San Jose.

2. Conférences nationales

[1] N-.S. Vu, A. Caplier. "Reconnaissance de visages en conditions de pose variables", WISG 2010, Troyes.

[2] N-.S. Vu, A. Caplier. "Normalisation d'illumination base sur un modèle de rétine : application à la reconnaissance de visages", Visage, RFIA 2010, Caen.

3. Brevets déposés

[1] N-.S. Vu, A. Caplier. "Method for performing face recognition on a digital image using illumination normalization".

[2] N-.S. Vu, A. Caplier. "Procédé et dispositif de reconnaissance de visages en conditions de pose variables".

4. Articles en cours de relecture

[1] N-.S. Vu, H. Dee, A. Caplier. "Face recognition using the POEM descriptor", soumis à Pattern Recognition.

[2] N-.S. Vu, A. Caplier. "Learned POEM for robust face recognition", soumis à IEEE Trans. On Image Processing.

Contributions à la reconnaissance de visages à partir d'une seule image et dans un contexte non-contrôlé

Résumé : Bien qu'ayant suscité des recherches depuis 30 ans, le problème de la reconnaissance de visages en contexte de vidéosurveillance, sachant qu'une seule image par individu est disponible pour l'enrôlement, n'est pas encore résolu. Dans ce contexte, les deux défis les plus difficiles à relever consistent à développer des algorithmes robustes aux variations d'illumination et aux variations de pose. De plus, il y a aussi une contrainte forte sur la complexité en temps et en occupation mémoire des algorithmes à mettre en oeuvre dans de tels systèmes.

Le travail développé dans cette thèse apporte plusieurs avancées innovantes dans ce contexte de reconnaissance faciale en vidéosurveillance. Premièrement, une méthode de normalisation des variations d'illumination visant à simuler les performances de la rétine est proposée en tant que pré-traitement des images faciales. Deuxièmement, nous proposons un nouveau descripteur appelé POEM (Patterns of Oriented Edge Magnitudes) destiné à représenter les structures locales d'une image. Ce descripteur est discriminant, robuste aux variations extérieures (variations de pose, d'illumination, d'expression, d'âge que l'on rencontre souvent avec les visages). Troisièmement, un modèle statistique de reconnaissance de visages en conditions de pose variables, centré sur une modélisation de la manière dont l'apparence du visage évolue lorsque le point de vue varie, est proposé. Enfin, une nouvelle approche visant à modéliser les relations spatiales entre les composantes du visage est présentée. A l'exception de la dernière approche, tous les algorithmes proposés sont très rapides à calculer et sont donc adaptés à la contrainte de traitement temps réel des systèmes de vidéosurveillance.

Towards unconstrained face recognition from one sample

Abstract : Although having been an active research topic for 30 years, recognizing a person from surveillance having seen only one image is unsolved. Within this context, the two greatest challenges are the variations of pose and illumination. Moreover, there are strict constraints upon the complexity in both terms of computational time and stockage requirements.

The work developed throughout this dissertation gives several advantages in the context of real-time and unconstrained face recognition. Firstly, an illumination normalization method simulating the performance of human retina is proposed as preprocessing algorithm. Secondly, we propose novel features called POEM (Patterns of Oriented Edge Magnitudes) for representing a local image structure. This descriptor is discriminative and robust to exterior variations (variations of pose, illumination, expression and pose that we always see when dealing with face images). Thirdly, a statistical model for robust face recognition across poses, centered on modeling how facial patch appearance changes as the viewpoint varies, is proposed. Finally, a novel approach modeling the spatial relationships between face components is developed. Except the last algorithm, all proposed methods are very fast and are therefore suitable for the constraints upon real-time of surveillance applications.

Key words : Unconstrained face recognition, real-time, illumination normalization, pose variations, retina, local descriptors, statistical model, HMMs.

Laboratoire GIPSA-lab - ENSE^3 - BP 46, 38402 Saint-Martin d'Hères, FRANCE.