Repousser les limites de l'identification faciale en contexte de vidéo-surveillance

Cécile Fiche

UNIVERSITÉ DE GRENOBLE

THÈSE

Pour obtenir le grade de

DOCTEUR DE L'UNIVERSITÉ DE GRENOBLE

Spécialité : **Signal - Images - Parole - Télécoms (SIPT)**

Arrêté ministériel : 31/01/2012

Présentée par

Mlle. Cécile Fiche

Thèse dirigée par **Mme. Alice Caplier**
et codirigée par **Mme. Patricia Ladret**

préparée au sein **du laboratoire Images Parole Signal et Automatique de Grenoble (GIPSA-lab)**
et de **l'école doctorale d'Électronique, Électrotechnique, Automatique et Traitement du Signal (EEATS)**

Repousser les limites de l'identification faciale en contexte de vidéo-surveillance

Thèse soutenue publiquement le **31/01/2012**,
devant le jury composé de :

M. Pierre-Yves COULON
Institut polytechnique de Grenoble, Président
M. Mohamed-Chaker LARABI
Université de Poitiers, Rapporteur
M. Mohamed DAOUDI
Institut Télécoms de Lille, Rapporteur
M. Frédéric LERASLE
Université Paul Sabatier de Toulouse, Examinateur
Mme. Alice CAPLIER
Institut polytechnique de Grenoble, Examinatrice
Mme. Patricia LADRET
Université Joseph Fourier de Grenoble, Examinatrice

Résumé

Les systèmes d'identification de personnes basés sur le visage deviennent de plus en plus répandus et trouvent des applications très variées, en particulier dans le domaine de la vidéosurveillance. Or, dans ce contexte, les performances des algorithmes de reconnaissance faciale dépendent largement des conditions d'acquisition des images, en particulier lorsque la pose varie mais également parce que les méthodes d'acquisition elles mêmes peuvent introduire des artéfacts. On parle principalement ici de maladresse de mise au point pouvant entraîner du flou sur l'image ou bien d'erreurs liées à la compression et faisant apparaître des effets de blocs. Le travail réalisé au cours de la thèse porte donc sur la reconnaissance de visages à partir d'images acquises à l'aide de caméras de vidéosurveillance, présentant des artéfacts de flou ou de bloc ou bien des visages avec des poses variables. Nous proposons dans un premier temps une nouvelle approche permettant d'améliorer de façon significative la reconnaissance des visages avec un niveau de flou élevé ou présentant de forts effets de bloc. La méthode, à l'aide de métriques spécifiques, permet d'évaluer la qualité de l'image d'entrée et d'adapter en conséquence la base d'apprentissage des algorithmes de reconnaissance. Dans un second temps, nous nous sommes focalisés sur l'estimation de la pose du visage. En effet, il est généralement très difficile de reconnaître un visage lorsque celui-ci n'est pas de face et la plupart des algorithmes d'identification de visages considérés comme peu sensibles à ce paramètre nécessitent de connaître la pose pour atteindre un taux de reconnaissance intéressant en un temps relativement court. Nous avons donc développé une méthode d'estimation de la pose en nous basant sur des méthodes de reconnaissance récentes afin d'obtenir une estimation rapide et suffisante de ce paramètre.

Mots Clefs - Reconnaissance de visages - Conditions non contrôlées - Estimateur de pose - Métriques de qualité sans référence - Flou - Effets de bloc

Abstract

The person identification systems based on face recognition are becoming increasingly widespread and are being used in very diverse applications, particularly in the field of video surveillance. In this context, the performance of the facial recognition algorithms largely depends on the image acquisition context, especially because the pose can vary, but also because the acquisition methods themselves can introduce artifacts. The main issues are focus imprecision, which can lead to blurred images, or the errors related to compression, which can introduce the block artifact. The work done during the thesis focuses on facial recognition in images taken by video surveillance cameras, in cases where the images contain blur or block artifacts or show various poses. First, we are proposing a new approach that allows to significantly improve facial recognition in images with high blur levels or with strong block artifacts. The method, which makes use of specific no-reference metrics, starts with the evaluation of the quality level of the input image and then adapts the training database of the recognition algorithms accordingly. Second, we have focused on the facial pose estimation. Normally, it is very difficult to recognize a face in an image taken from another viewpoint than the frontal one and the majority of facial identification algorithms which are robust to pose variation need to know the pose in order to achieve a satisfying recognition rate in a relatively short time. We have therefore developed a fast and satisfying pose estimation method based on recent recognition techniques.

Keywords - Face recognition - Uncontrolled condition of acquisition - Pose estimation - No-reference quality metrics - Blur - Blocks artefacts

Qui recherche la lune, ne voit pas les étoiles.

Proverbe français

Remerciements

Enfin ! Me voilà plongée dans l'écriture de la dernière page de ce manuscrit, les remerciements. Bizarrement, cela n'a pas été trop dure de m'y atteler... je tairai en revanche le nombre de reformulations dont ils ont fait l'objet. Comment remercier ceux qui m'ont suivie, écoutée, supportée et j'en passe... ? Ma thèse a été un doux mélange de remises en question, de doutes, de certitudes, de craquages et d'amusements dont j'ai fait subir les conséquences, plus ou moins heureuses, à presque toutes les personnes qui me connaissent. Combien de fois me suis-je demandée ce que je faisais là et si je n'étais pas un brin anormale de m'être engouffrée dans une voie pareille. Maintenant que j'en suis arrivée à bout, je n'ai plus aucun doute sur le sujet. La vie d'une thésarde est en effet tout sauf un parcours de santé. L'avantage, c'est que l'on ne s'ennuie jamais. L'inconvénient, c'est qu'il faut bien trois ans pour s'y habituer ! Trouver son équilibre relève d'un savant dosage qu'il appartient à chacun d'établir et c'est la raison pour laquelle je souhaite remercier ici toutes les personnes qui m'ont permis de l'obtenir.

En premier lieu ma famille, et en particulier mes parents. Malgré mon merveilleux caractère de ces derniers mois et toutes mes demandes express et de dernière minute (de préférence...), vous avez toujours été attentifs et là pour moi, merci ! J-B, mon frangin, parce que tu as toujours su trouver les mots justes quand il le fallait et réussi à égayer mes petits moments de blues quand j'en avais besoin. Mado, tu m'as accueillie les bras ouverts dès que cela a été nécessaire et tu as su trouver à me distraire malgré tous tes tracas du quotidien. Marie, pour ta disponibilité sans faille, ton optimisme et ta joie de vivre communicative. Et enfin Titi, pour tes délires et ton enthousiasme général !

Je remercie également l'ensemble des membres du jury qui ont accepté de lire et d'évaluer la qualité de mon manuscrit. L'enthousiasme avec lequel vous avez accueilli mon travail est un moteur formidable pour la suite. Merci à Monsieur Pierre-Yves Coulon d'avoir accepté de présider ce jury. Merci également à Messieurs Mohamed Daoudi et Mohamed-Chaker Larabi, tous deux rapporteurs, pour leurs remarques constructives, leurs encouragements et leurs retours positifs vis-à-vis de ma thèse. Et enfin, merci à Monsieur Frédéric Lerasle, examinateur, pour la qualité de ses observations et de ses suggestions sur mon travail et pour tous ses encouragements.

Je tiens également à remercier mes deux encadrantes de thèse qui m'ont permis d'arriver à bout de ce manuscrit. Merci à Patricia Ladret pour la liberté que tu m'as laissée et la confiance que tu m'as accordée au cours de ces trois ans. J'ai particulièrement apprécié ta disponibilité et ton soutien dans les moments difficiles, je n'oublie pas le coup de fil salvateur de fin de thèse et je garde un très bon souvenir de notre conférence à San

Francisco en compagnie d'Eric. Merci à Alice Caplier pour ton implication, tes remarques et la qualité de tes suggestions pour l'écriture de mon manuscrit et la préparation de mon oral. Merci à vous deux pour vos encouragements et votre confiance tout au long de cette thèse, j'en avais bien besoin ! Merci également à Anne Guérin pour m'avoir éclairée sur le fonctionnement des SVMs et enfin à Lucia, Jean-Marc et Laurent pour leur bonne humeur et pour m'avoir souvent débloquée des situations en un temps record.

Ce travail n'aurait sans doute pas pu aboutir sans la présence de mes amis et collègues du Gipsa, autant d'un point de vue scientifique que d'un point de vue humain. Son, le pilier du bureau D1146, ta gentillesse, ta disponibilité et tes conseils avisés sur mon travail m'ont été d'un grand secours et une source de motivation constante. Claire dont le regard sur la thèse m'a souvent permis de relativiser. Benjamin qui est allé chercher, plus souvent qu'à son tour, mon panier à l'Amap lorsque j'étais dans le rush. Ladan pour nos discussions, bien souvent au milieu d'un couloir, ta générosité et toute l'aide que tu m'as apportée pour concrétiser mon après thèse. Barth, ton humour décalé, les débats scientifico-philosophiques au RU ou les délires au café des admins (auxquels s'est souvent joint Laurent !) m'ont beaucoup manqué. Guanghan, Hao, Weiyuan et Zhongyang, une rencontre formidable avec vous quatre ! Un joli mélange d'enthousiasme, de bienveillance, de subtilité, d'un grain de folie, le tout saupoudré d'une bonne humeur à toute épreuve rendent votre groupe incontournable ! Jérémie, outre le fait que tu sois le seul thésard qui ait réussi à me faire croire pendant un an que tu étais en post-doc... j'ai tout de suite adhéré à ton humour déconcertant, tes proverbes à contre-pied et je pense que tu dois être le seul à pouvoir m'acheter avec une tarte au citron ! Vincent, pour les nombreux délires que nous avons eu avec Raluca à l'*Entracte* qui doit encore se souvenir de la course-poursuite au café et pour les nombreuses pauses que nous avons prises ensemble, à 10h comme à 22h, qui m'ont souvent permis de décompresser. Aktham pour ton écoute, ta disponibilité et ta patience infinie. Et bien évidemment, je ne peux oublier ma co-bureau et avant tout amie Raluca qui a fait de ces derniers mois au Gipsa, les meilleurs de ces trois années ! Multumesc pentru tot ;)

Ces remerciements ne seraient être complets si je n'avais une pensée pour tous mes amis d'ici et d'ailleurs qui ont fait preuve d'une patience et d'un soutien sans faille à mon égard. Athénaée, pour tous les bons moments sur Aix, sur Fréjus et Grenoble dont je garde de super souvenirs ! Lise, ma londonienne préférée, merci pour tous les fous rire et les discussions que j'ai pu avoir avec toi malgré les kilomètres qui nous séparent. Vincent, pour être venu à ma soutenance malgré toutes les difficultés du quotidien. Katia&Olivier, pour tous les délires que j'ai pu partager avec vous, à Grenoble et ailleurs, en voiture, à moto ou à pied (et bientôt dans les airs...). Vinmille, pour ton aide précieuse sur la forme de ce manuscrit, la touche artistique que tu as apportée à notre bureau et ton auto-dérision que j'apprécie particulièrement. Camilo, mon super coloc' ! Tu as atterri sur Grenoble à point nommé ! Aurélien, un Lindy-hopeur hors pair... quand est-ce que tu donnes des cours sur Paris ? Jessica, courage, le plus difficile dans la rédaction, c'est de s'y mettre ! Et bien évidemment, je n'oublie pas toute la team du *sirop d'la rue* (Edith, Julien, Lucas, Nico-chapeau et Thomas) pour toutes les soirées passées dans ce bar mythique et pour ne pas avoir respecté à la lettre mon fameux paris-TuTu ! Lucie, pour tes remarques pertinentes

sur les diapos de ma soutenance et pour toutes nos discussions téléphoniques remonte-moral souvent pleines d'humour... à quand notre prochaine descente au casino ? Merci à Thomas pour ta générosité, les jeudis du Scrabble et pour avoir supporté mon humour parfois caustique ! Lucas pour m'avoir supportée tout court... mais aussi pour tous les Fréjus passés et à venir à jouer au poker-frosties, à chercher des œufs de Pâques en haut des cyprès ou à rapiécer ton tapis. Et enfin Edith pour nos parties de Mamie-Scrabble, nos nuits blanches à rédiger (c'était quand même vachement plus supportable à deux !) et pour toutes nos discussions interminables sur nos thèses... et autre ! Merci.

The end... OUF ! !

Sommaire

Introduction générale

Depuis plusieurs années, les systèmes d'identification de personnes deviennent de plus en plus répandus et trouvent des applications très variées dans la vie de tous les jours. Ce type de système permet notamment de s'assurer de l'identité d'une personne et d'en éviter les usurpations éventuelles. De manière générale, nous pouvons classer ces systèmes en trois grandes catégories :

1. ceux qui sont basés sur ce que l'on mémorise, par exemple un code de carte bancaire ou un mot de passe

2. ceux qui sont basés sur ce que l'on possède comme une carte d'identité, un permis de conduire, un badge...

3. ceux qui sont basés sur ce qui caractérise chaque individu. Cette dernière catégorie regroupe l'ensemble des méthodes dites « biométriques » car elles permettent de reconnaître une personne grâce à des caractéristiques physiques ou des comportements qui lui sont propres. Parmi l'ensemble de ces méthodes, on peut citer celles qui se servent de l'iris, des empreintes digitales, de la voix et, bien évidemment, du visage, lequel fait l'objet de notre étude.

La première catégorie de systèmes d'identification est très largement répandue dans la vie de tous les jours et fait maintenant partie intégrante de notre mode de vie actuel. La seconde et la troisième sont en revanche beaucoup plus souvent rencontrées dans les grandes entreprises ou dans certains lieux publics tels que les gares ou aéroports dans lesquels une sécurité renforcée est nécessaire. Elles y sont principalement employées pour faciliter le contrôle d'accès. Une pièce d'identité est requise que la biométrie permet d'authentifier. En effet, s'il est encore possible de connaître le mot de passe de quelqu'un à son insu ou de s'approprier une pièce d'identité qui n'est pas la sienne, il est en revanche très difficile d'usurper les caractéristiques d'une personne avec de tels systèmes. En outre, l'iris et les empreintes digitales sont reconnues comme des caractéristiques uniques à chacun d'entre nous. Néanmoins, l'utilisation de ces systèmes implique l'accord préalable des personnes qui y seront soumises. Les personnes qui souhaitent entrer dans une entreprise acceptent ces contrôles et ont intérêt à faciliter leur propre identification quelque soit la méthode utilisée. Ces vérifications sont d'ailleurs généralement menées dans un environnement contrôlé où tous les paramètres ont été optimisés pour rendre l'identification la plus fiable possible. Or, à la suite d'événements récents, nous avons vu une volonté crois-

sante de la part des politiques d'augmenter le niveau de sécurité dans les lieux publics (gares, aéroports, écoles) mais aussi privés (banques, entreprises), notamment grâce au développement de la vidéosurveillance. Ce système permet en effet d'acquérir des centaines d'images d'un endroit précis à n'importe quel moment sans nécessiter d'infrastructures particulières. Elle ne permet donc pas de contrôler mais plutôt de surveiller un lieu et de prévenir, en théorie, certains agissements non désirables. En effet, les images acquises par une caméra de vidéosurveillance sont généralement enregistrées puis envoyées à un système de contrôle avant d'être affichées sur un écran dans un centre de surveillance. Ces images sont ensuite analysées par une personne habilitée. Malheureusement de nombreuses études ont montré qu'un être humain pouvait difficilement se concentrer et analyser les images, issues en même temps de plusieurs caméras, au-delà d'une durée de vingt minutes [GF09]. Par ailleurs, une étude a récemment montré que l'identification d'une personne par des individus avec lesquels elle n'est pas familière est source de nombreuses erreurs [DV09]. Dès lors, on peut se demander quel est l'intérêt d'avoir un parc de vidéosurveillance en continuel développement s'il n'est pas possible de traiter l'ensemble de l'information acquise par ces systèmes. Une réponse donnée depuis plusieurs années est bien entendu l'essor des systèmes de reconnaissance automatique basés sur le visage. En effet, contrairement aux méthodes biométriques intrusives comme celles basées sur l'iris, les empreintes digitales ou la main, celles basées sur le visage permettent d'identifier une personne sans nécessiter la mise en place d'infrastructure particulière ni même l'accord préalable de la personne à reconnaître. C'est là tout leur intérêt, permettre la reconnaissance d'un visage dans un environnement non contrôlé comme nous sommes capables de le faire, par nous même, dans la vie de tous les jours. Malheureusement, si le taux d'identification est très élevé dans un cadre où tous les paramètres d'illumination et d'acquisition sont maîtrisés, ce n'est pas encore le cas lorsque ces paramètres sont inconnus a priori et sont susceptibles de varier. En particulier, il n'existe pas, à l'heure actuelle, de système automatique de reconnaissance performant permettant la reconnaissance d'une personne à partir d'images de vidéosurveillance. Le projet Biorafale dans lequel s'inscrit cette thèse, vise à développer un système de reconnaissance temps réel pour détecter les individus interdits de stade. Elle s'inscrit donc dans le cadre de la reconnaissance de visages à partir d'images issues de caméras vidéosurveillance.

Dans cette introduction, nous explicitons le principe général de la reconnaissance puis nous décrivons en quoi consiste le projet Biorafale avant de présenter notre problématique et enfin le plan de la thèse.

1. Principes de la reconnaissance de visages

Avant d'être capable de reconnaître une personne sur une image ou sur une vidéo, plusieurs étapes de traitement sont nécessaires. Dans les grandes lignes, on peut découper un processus de reconnaissance en quatre étapes principales qui sont présentées sur la **Figure 0.1**.

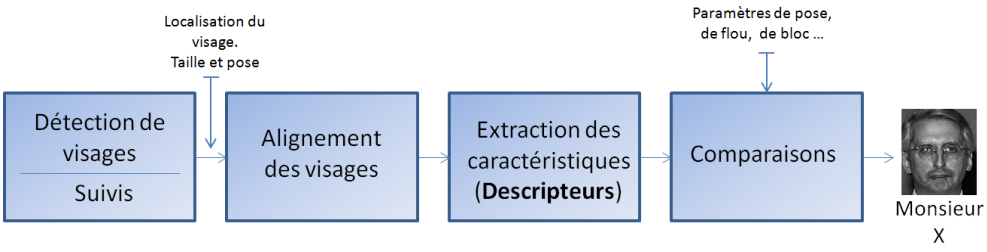

Figure 0.1 – *Ensemble du processus de reconnaissance d'un visage.*

La détection (ou le suivi) du visage d'une personne dans une image ou une vidéo Cette étape consiste à dire si oui ou non il y a un visage sur une image (ou une personne), combien il y en a et où ils sont situés dans l'image.

L'alignement du visage Cette étape consiste à permettre une normalisation de l'image de visage détecté afin de pouvoir la comparer avec les autres images de visages de la base qui n'ont pas forcément la même taille ni subi le même éclairage. Cela nécessite donc la détection de certains traits du visage (généralement, les yeux, le nez et la bouche) et la normalisation des distances existantes entre ces traits. Cela suppose également de connaître la pose du visage. L'étape de normalisation d'illumination nécessite également un traitement particulier.

L'extraction des caractéristiques du visage Une fois l'alignement réalisé, il s'agit d'extraire les caractéristiques les plus pertinentes d'un visage pour permettre son identification et ne pas le confondre avec celui d'un autre individu. Le vecteur de caractéristiques ainsi associé à un visage doit être robuste à toutes les variations possibles, telles que l'expression, la pose, l'illumination.

L'étape de comparaison Cette étape est une étape de décision qui permet de déterminer l'identité d'une personne par la comparaison des vecteurs de caractéristiques. Certaines informations sur la qualité des images par exemple, obtenues lors d'une étape de prétraitement indépendante peuvent être introduites à ce niveau pour permettre une identification plus performante.

L'identification d'un visage nécessite donc plusieurs étapes indispensables à son fonctionnement et elles représentent chacune une problématique à part entière.

2. Projet Biorafale

Le projet Biorafale, débuté fin 2008 et financé par OSEO, vise à mettre au point un système de reconnaissance performant pour permettre l'identification de personnes interdites de stade. Cela nécessite le développement d'un logiciel capable de reconnaître le visage d'une personne au moment de son entrée dans le stade au niveau des portiques ou lors de la phase de fouille. Les contraintes liées à ce projet sont multiples. D'une part, le projet se place dans un contexte de vidéosurveillance. Cela signifie que les conditions d'acquisition des images sont non contrôlées et que l'identification doit pouvoir se faire en temps réel. D'autre part, il vise à s'adapter aux systèmes de vidéosurveillance déjà existants et non à mettre en œuvre un système spécifique. Autrement dit, le but n'est pas de reconstruire tout le parc de vidéosurveillance mais bien de pouvoir déployer un logiciel de reconnaissance de visage pour plusieurs types de caméras plus ou moins récentes. Celles-ci étant généralement mises en place au moment de la construction du stade, leurs caractéristiques peuvent changer d'un stade à l'autre. Par ailleurs, les systèmes de vidéosurveillance envisagés ne disposent pas de stéréo-vision permettant de reconstruire le visage d'une personne en trois dimensions (3D). Il s'agit donc de développer un système de reconnaissance de visages sur des images à deux dimensions (2D) uniquement. De plus, nous ne disposons dans la base de données des interdits de stade que d'une vue de face par personne à reconnaître.

La mise au point de ce prototype s'appuie sur une collaboration entre plusieurs partenaires. Ainsi, les laboratoires LASMEA de Clermont-Ferrand, EURECOM de Nice et le GIPSA-Lab de Grenoble participent à ce projet au niveau de la phase de conception des algorithmes. Les partenaires industriels Vesalis, IBM et Effidence facilitent et coordonnent l'intégration de ces algorithmes dans un système global. Le Ministère de l'intérieur et la préfecture de police de Paris permettent quant à eux la mise en œuvre de ce projet d'un point de vue juridique comme pratique. Les rôles de chacun de ces partenaires sont donc bien définis. En particulier, le LASMEA intervient au niveau de la phase de détection des visages et de suivi qui correspond à la première étape du processus de reconnaissance. Le GIPSA-Lab intervient principalement au niveau des deux étapes suivantes à savoir l'alignement des visages et l'extraction de caractéristiques qui visent à améliorer les performances de l'étape d'identification. EURECOM intervient quant à lui directement au niveau de la phase de reconnaissance en apportant de l'information supplémentaire basée principalement sur l'apparence de la personne.

Dans le cadre de cette thèse, nous supposons que les étapes de détection et d'alignement sont déjà réalisées. Nous nous situons donc au niveau des étapes d'extraction

de caractéristiques et de comparaison du processus de reconnaissance. D'autre part, nous nous plaçons dans un contexte de vidéosurveillance pour lequel les images acquises peuvent avoir été dégradées et nous supposons qu'une seule vue de face par personne est disponible dans la base de données des interdits de stade.

3. Problématique

Dans le cadre de l'identification d'un visage, une des difficultés principales réside dans l'extraction de caractéristiques qui soient suffisamment représentatives d'un visage donné (sans risque de confusion avec d'autres) tout en étant robustes aux variations que pourrait subir ce même visage. L'utilisation de tels vecteurs permet de décrire l'image d'un visage de façon beaucoup plus précise que l'image brute de départ. Cette étape consiste à mettre en évidence les variations des traits d'un visage qui sont propres à un individu donné. Or dans le cadre d'un environnement non contrôlé dans lequel s'inscrit la vidéosurveillance, il n'est pas possible de limiter ces variations. De plus, elles sont rarement causées par un artéfact isolé mais sont au contraire la résultante d'une combinaison de plusieurs artéfacts. Les variations peuvent être intrinsèques à la personne et donc liées par exemple à son âge, son expression (sourire ou non par exemple), ce qu'elle porte (des lunettes, pas de lunettes) ainsi qu'à l'orientation de son visage (de profil, de face). Cette dernière constitue un artéfact majeur pour la reconnaissance dont les conséquences sur les performances des algorithmes d'identification n'ont toujours pas été résolues. D'autres artéfacts peuvent également être causés par un événement extérieur tel qu'une variation d'illumination (l'éclairement d'un visage peut être plus important sur certains traits que sur d'autres, causant une inhomogénéité en terme de luminance), une occultation (une partie importante du visage peut être cachée par un objet extérieur) ou bien une mauvaise acquisition de l'image qui affecte sa qualité. L'apparition de flou ou de blocs est en effet très courant sur des vidéos. Pour autant peu de solutions ont été proposées pour palier ces effets dans le cadre de l'identification d'un visage. Ainsi, la présence d'un ou de plusieurs de ces artéfacts sur une image fait diminuer considérablement les performances des algorithmes de reconnaissance de visages. Ils sont parfois tels que le système de reconnaissance trouve plus de similarités entre les visages de deux individus différents qu'entre les images présentant des variations différentes d'un même visage. Que faire dans ce cas pour traiter la reconnaissance d'un individu ?

Idéalement, l'idée est de pouvoir représenter l'image d'un visage avec un vecteur de caractéristiques qui varierait peu lorsque l'image d'une même personne subit une modification extérieure, quelle qu'elle soit, mais qui varie au contraire beaucoup lorsque l'image est celle du visage d'une personne différente. A l'heure actuelle, plusieurs solutions ont été proposées pour traiter certaines modifications du visage de façon individuelle. Il existe en effet des descripteurs de visages robustes à certaines variations. C'est le cas du descripteur LBP, proposé par Ojala et al. dans [OPM02], qui permet de s'affranchir des problèmes

causés par des variations d'illumination ou bien du descripteur LPQ proposé par Ojansivu et al. dans [OH08] qui se veut robuste aux variations de flou. Le problème de ce type de méthode est qu'elles sont généralement spécifiques à un artéfact donné. En présence d'autres artéfacts, comme c'est le cas dans un environnement non contrôlé, leurs performances diminuent. De plus, elles ne permettent généralement pas d'obtenir une identification de 100% même en ne considérant qu'un seul artéfact. Vouloir proposer un vecteur de caractéristiques robuste à toutes les variations rencontrées sur des images de vidéosurveillance est donc une tâche très difficile à résoudre. En revanche, utiliser les avantages de ces descripteurs et proposer des solutions pour améliorer leurs performances face à d'autres artéfacts est beaucoup plus adapté. Dans cette thèse, plutôt que de proposer un nouvel extracteur de caractéristique robuste à un artéfact donné, nous proposons d'apporter de l'information complémentaire sur l'image, au moment de la classification, afin d'améliorer les performances des algorithmes de reconnaissance dans des conditions d'utilisation inhabituelles (images floues par exemple). Cette approche a l'avantage de pouvoir être adaptée aux divers extracteurs de caractéristiques disponibles sans en modifier les spécificités.

Plus spécifiquement, nous proposons d'apporter des améliorations en termes d'identification sur des images de visages présentant des artéfacts de flou, de blocs ou des variations de pose tout en respectant les contraintes liées à la vidéosurveillance. Peu d'études se sont en effet intéressées à l'impact de la qualité des images acquises avec des caméras de vidéosurveillance, en particulier en cas de flou et de blocs. Les performances des méthodes de reconnaissance sur des images de visages avec de larges variations de pose sont quant à elles encore insuffisantes. C'est pourquoi nous avons également élaboré une nouvelle approche permettant d'apporter de l'information sur la variation de l'image de visage à reconnaître par rapport à l'image de référence que nous avons à notre disposition. Pour les trois perturbations Flou, Bloc et Pose, nous proposons une seule et même démarche. Celle-ci consiste à adapter la base de données contenant les visages connus en fonction de la dégradation de l'image test du visage que nous souhaitons identifier. En effet, une image test qui présente de multiples artéfacts contient des informations qui dépendent de ces artéfacts et cela fausse la reconnaissance. L'idée est donc d'introduire au niveau des images de visages connus (généralement de bonne qualité) le même degré d'artéfact. La démarche comprend trois étapes distinctes :

Premièrement *Modélisation* de la dégradation.

Deuxièmement *Estimation* du niveau de l'artéfact présent dans l'image.

Troisièmement *Sélection* de la galerie selon le niveau de l'artéfact estimé.

Notons que dans ce travail seront considérées comme des perturbations toutes les modifications sur les images qui engendrent une dégradation des performances des algorithmes de reconnaissance de visages. Le schéma récapitulant cette démarche est présenté sur la **Figure 0.2**.

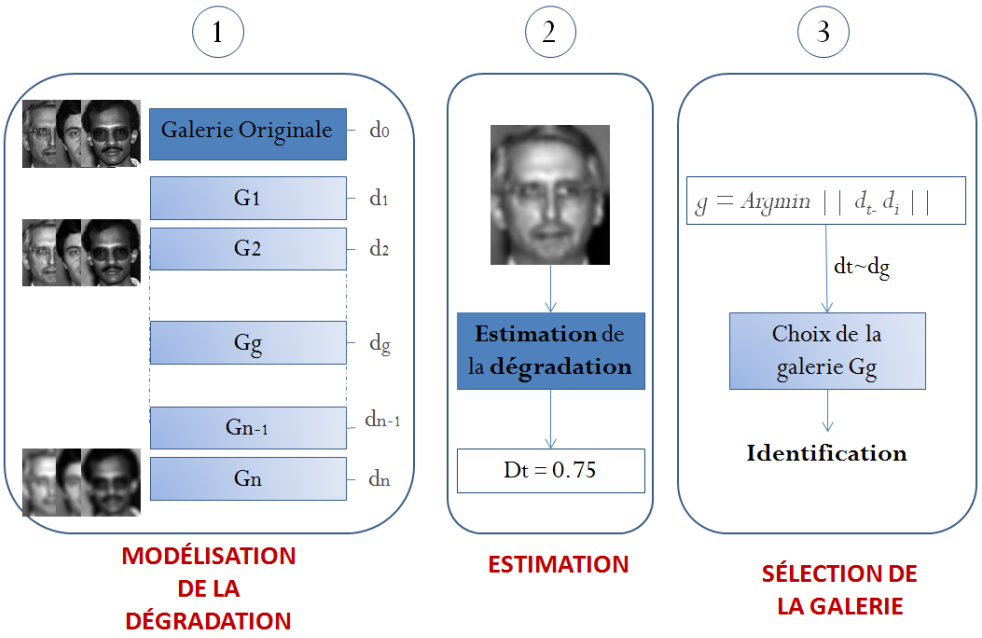

Figure 0.2 – *Principe de l'approche proposée. 1) Modélisation de la dégradation de l'image d'entrée. 2) Estimation du niveau de l'artéfact présent dans l'image. 3) Sélection de la galerie afin de permettre une comparaison adaptée.*

Dans cette thèse, nous avons adapté ce schéma pour le traitement des trois perturbations considérées : flou, effet de bloc et pose. Dans le cas des artéfacts de flou et d'effet de bloc, notre contribution principale a porté sur la première et la troisième étape de la méthode globale, à savoir la modélisation de la dégradation (flou ou bloc) et la sélection de la galerie conformément aux dégradations estimées sur l'image test (**Figure.0.2.1& 3)**). Dans le cas de la perturbation liée à la pose, nous nous sommes cette fois-ci focalisés sur la seconde étape en développant un nouvel estimateur de pose (**Figure.0.2.2)**). L'intérêt de l'étape de sélection a déjà été prouvé puisqu'il a déjà été démontré que si nous comparons des images de même pose, le taux de reconnaissance est meilleur que si nous comparons des images de poses différentes.

Chacune de ces perturbations est traitée séparément pour évaluer l'amélioration apportée par notre approche à chacune d'elle. Dans tous les cas, nous nous sommes soumis aux contraintes du projet Biorafale. Nous avons donc supposé ne disposer que d'une image de visage par personne dans la base de données, cette image représentant un visage pris de face dans des conditions optimales avec une bonne qualité d'acquisition. L'étape de reconnaissance a quant à elle été destinée à l'identification de visages 2D uniquement. Par ailleurs, nous nous sommes essentiellement concentrés dans cette thèse à l'étude du taux d'identification. Chacune des personnes test présentées à l'algorithme était présente dans la galerie. Les cas de faux positifs ou de faux négatifs ne sont donc pas traités dans ce manuscrit. En ce qui concerne la pose, seule l'orientation de la tête autour de l'axe du

visage est considérée (Yaw). Ni les variations de la pose dans le plan de l'image (Roll), ni les variations de la pose de bas en haut (Pitch) ne sont traitées.

4. Principales contributions

Nous proposons dans cette thèse trois contributions principales. Les deux premières portent sur l'amélioration des performances des algorithmes de reconnaissance en présence d'images fortement dégradées par des effets de flou d'une part et des effets de blocs d'autre part. La troisième porte sur l'amélioration des performances des méthodes d'identification de visages lorsque la pose varie.

4.1 Amélioration des performances des algorithmes de reconnaissance en présence d'images fortement dégradées par du flou et des effets de blocs

Dans une vidéo, la qualité des images est variable. L'approche présentée, que ce soit pour le flou ou l'effet de bloc, permet d'obtenir des performances de reconnaissance convenables même lorsque les images à disposition sont toutes de mauvaise qualité. Évaluer la qualité de l'image d'entrée permet d'apporter une information qui peut être utilisée de deux façons. En effet, d'une part, elle peut permettre de définir un seuil de qualité à partir duquel les images de la vidéo sont rejetées par le système de reconnaissance. Mais que faire alors lorsque seules des images de qualité médiocre sont disponibles ? Dans ce cas, l'information de qualité permet, selon l'approche que nous avons proposée, d'adapter les images de la galerie en conséquence.

L'application de cette approche pour la gestion des images floues a permis d'améliorer très nettement les performances des deux algorithmes de reconnaissance testés que sont les méthodes proposées par Ahonen et al. basées sur le descripteur LBP [OPM02] pour l'un et sur le descripteur LPQ [OH08] pour l'autre. Pour autant, l'approche proposée est indépendante du système de reconnaissance utilisé et peut s'appliquer à plusieurs autres méthodes d'identification. Par ailleurs, nous avons testé notre approche pour deux types de flou : le flou de mise au point et le flou de bougé, et pour les deux catégories de flou, les résultats obtenus sont très convaincants. De plus, pour estimer le niveau de flou des images d'entrée, nous avons utilisé la métrique de flou sans référence BluM proposée par Crête et al. dans [CRDLN07]. Aucun a priori sur la qualité des images étudiées n'a donc été nécessaire.

Les deux méthodes de reconnaissance que nous venons de citer sont relativement robustes aux artéfacts de bloc excepté pour de très forts taux de compression. L'application de cette même démarche pour la gestion des effets de blocs a permis d'améliorer signifi-

cativement les taux d'identification obtenus pour de très forts taux de compression. De même que précédemment, l'approche proposée pour l'effet de bloc est indépendante du système de reconnaissance utilisé et aucun a priori sur la qualité de l'image n'a été nécessaire. Nous avons pour cela fait appel à la métrique sans référence d'estimation de l'effet de bloc proposée par Crête et al. dans [FR07].

4.2 Amélioration des performances des méthodes d'identification de visages lorsque la pose varie

Dans la deuxième partie de la thèse, nous avons concentré notre travail sur l'élaboration d'un nouvel estimateur de pose. Nous avons modélisé le problème comme un problème de classification étant donné que, pour une application à la reconnaissance de visages, il n'est pas nécessaire de connaître l'angle de la pose au degré près. Les méthodes de reconnaissance tolèrent en effet une variation d'angles de plusieurs degrés. L'angle de la pose est donc estimé de façon discrète. Nous avons pour cela utilisé dans un premier temps une combinaison du descripteur de visage POEM [VC10] avec une méthode de classification basée sur les SVMs. Les résultats obtenus sont comparés aux performances de ce même estimateur lorsque le descripteur proposé est LGBP qui a été proposé par Ma et al. dans [MZS+06]. L'estimateur que nous avons développé n'engendre pas un temps de calcul important contrairement au descripteur LGBP. Notre estimateur peut donc très bien s'appliquer à notre contexte de vidéosurveillance. Dans un second temps, nous avons amélioré les performances de notre estimateur en combinant le descripteur POEM utilisé avec un ensemble de 40 ondelettes de Gabor. Les performances ont été notablement améliorées et surpassent celles de l'état de l'art. Le temps de calcul de cet estimateur a été en contrepartie fortement augmenté empêchant sa mise en œuvre dans un contexte de vidéosurveillance.

Structure du mémoire

Dans ce manuscrit, nous présentons le travail qui a été réalisé en reconnaissance de visages sur des images acquises dans un contexte de vidéosurveillance. Nous nous sommes intéressés dans un premier temps à l'amélioration des performances des algorithmes de reconnaissance de visages sur des images présentant des artéfacts de flou et des effets de blocs. Puis dans un second temps nous avons traité l'amélioration de la reconnaissance de visages dans le cas où la pose varie en proposant un nouvel estimateur de pose.

Dans le **chapitre 1**, nous présentons un état de l'art des algorithmes de reconnaissance de visages 2D. En particulier, nous détaillons les méthodes d'identification qui ont permis une avancée notable dans le domaine et celles qui sont le plus adaptées à notre contexte. Puis, nous décrivons les bases de données qui ont été utilisées pour valider les résultats obtenus au cours de cette thèse. Enfin, nous présentons une évaluation des performances de plusieurs algorithmes de reconnaissance en cas d'images présentant des artéfacts de flou ou de blocs. Cette dernière partie permet de mettre en évidence le fort impact de ces artéfacts sur les capacités de reconnaissance et l'intérêt d'évaluer en amont la qualité des images d'entrée.

Dans le **chapitre 2**, nous faisons une revue des artéfacts qui peuvent être introduits dans des images de vidéosurveillance puis nous présentons dans l'état de l'art des métriques de qualité sans référence consacrées à l'évaluation du niveau de flou dans une image d'une part et du niveau de blocs d'autre part.

Dans le **chapitre 3**, nous présentons l'approche que nous avons proposée pour effectuer une reconnaissance de visage sur des images floues puis pour des images dégradées par effet de bloc.

Dans le **chapitre 4**, nous traitons le problème de l'estimation de la pose appliquée à la reconnaissance d'un visage. Nous décrivons l'intérêt d'un estimateur de pose et les caractéristiques qu'il doit avoir pour une application à un contexte de videosurveillance. Nous présentons ensuite l'état de l'art des méthodes existantes avant de présenter l'estimateur de pose que nous avons développé et les résultats que nous avons obtenus.

Chapitre 1

Reconnaissance faciales 2D : état des lieux et limites des méthodes en contexte de vidéosurveillance

Introduction

Les systèmes d'identification de personnes basés sur la biométrie deviennent de plus en plus répandus et trouvent des applications très variées dans la vie de tous les jours. Beaucoup d'algorithmes ont vu le jour depuis une trentaine d'années et sont essentiellement utilisés pour le contrôle d'individus dans des lieux publics tels que les aéroports ou privés comme certaines grosses entreprises. Mais la majorité de ces systèmes fonctionne dans un environnement dit contrôlé. C'est à dire que toutes les conditions pouvant détériorer la reconnaissance sont maîtrisées. Les individus sont coopératifs si bien que les visages sont acquis de face, sans expression ni variation d'illumination. Or, dans le cas de la vidéo surveillance, il est impossible de contrôler les conditions d'acquisition que ce soit au niveau du comportement des individus (pose, expression...) ou au niveau des conditions d'éclairement.

Ainsi, ce que nous faisons chaque jour sans y penser est en fait un exercice très complexe qu'il est extrêmement difficile de modéliser. Notre habileté à traiter l'information perçue par notre système visuel, y compris lorsque les conditions sont mauvaises (visages peu éclairés, photos de mauvaise résolution etc...), est en effet remarquable. Il n'est donc pas étonnant que de nombreuses études aient été menées pour comprendre les mécanismes

mis en jeu lors du traitement de l'information. La compréhension de notre système visuel permet en effet le développement de nouvelles stratégies permettant l'élaboration de nouveaux systèmes de reconnaissance de plus en plus performants. C'est pourquoi, nous débuterons ce chapitre par une brève introduction à la reconnaissance de visages d'un point de vue neurocognitif avant de présenter l'état de l'art des techniques de reconnaissance de visages.

1. Introduction : la reconnaissance de visages d'un point de vue neurocognitif

Le visage peut fournir un nombre incroyable d'informations sur une personne et constitue un élément clé lors de l'étape d'identification. En outre, il permet de déterminer en une fraction de seconde, le sexe, la race, l'identité de son interlocuteur et d'y associer, dans le cas d'une personne connue, une multitude d'informations la concernant, comme son nom, son âge, sa profession... Il permet également de transmettre des émotions : par exemple, un changement d'expression comme un sourire, un hochement de tête ou une grimace permettront d'exprimer la joie, l'approbation ou bien le dégoût tandis que percevoir la peur chez une personne pourra renvoyer un signal de danger chez son interlocuteur. Tout ceci constitue autant d'éléments facilitant les interactions sociales entre individus. En raison du rôle fondamental que joue la perception des visages dans la communication non verbale (notamment celle des émotions), il est peu surprenant que de nombreuses études aient été menées sur ce sujet depuis de nombreuses années [BCT09], [DC86], [RB08], [Tho80]. Notre habileté à reconnaître des visages est en effet remarquable quand on sait qu'ils sont tous constitués des mêmes éléments fondamentaux (deux yeux, un nez, une bouche...), d'une même configuration (les yeux au dessus du nez, le nez au dessus de la bouche) et que la variabilité de chacun de ces éléments est très faible. En effet, il est possible de distinguer plusieurs formes de bouche mais il est difficile d'en définir une infinité. Pourtant, nous sommes tous capables d'identifier très précisément et rapidement plusieurs centaines d'individus en un temps très court ce qui nous rend expert dans ce domaine [DC86]. La difficulté est donc de trouver les mécanismes de perception qui permettent de rendre un visage unique. Les études menées jusqu'à maintenant ont permis de montrer que la reconnaissance d'un individu suppose à la fois l'utilisation des caractéristiques (éléments fondamentaux) du visage mais aussi de la configuration des traits qui le composent ainsi que des relations qui peuvent exister entre ces traits [BCT09], [DC86], [RB08]. Ainsi, si l'on perturbe l'une ou l'autre de ces informations, le taux d'identification chute significativement mais cela n'empêche pas pour autant la reconnaissance. Plusieurs expériences ont été menées dans ce sens. La plupart visent à modifier la configuration du visage en changeant par exemple la distance entre les yeux et d'autres traits qui le composent, mais aussi à perturber notre habileté à traiter l'information configurale en modifiant certains contours ou en inversant le visage comme le montre l'illusion de Thatcher présentée sur la **Figure 1.1** initialement proposée par Thompson et al. [Tho80].

Figure 1.1 – *Illusion de Thatcher. L'image en bas à gauche est la photo originale de Margaret Thatcher. L'image en bas à droite correspond à la même photo mais dans laquelle certains traits du visage ont été retournés ce qui donne son aspect grotesque à la photo. Les deux images du haut correspondent aux deux du bas renversées. Bien qu'elles soient respectivement identiques, il est plus difficile de distinguer une différence entre les deux photos du haut qu'entre celles du bas. Cette illusion permet ainsi de mettre en évidence l'existence de mécanismes spécifiques au traitement d'un visage à l'endroit et de montrer que les mécanismes de reconnaissance dépendent de l'orientation du visage. [Tho80]*

Dans [BCT09], les auteurs présentent les techniques qui ont principalement été utilisées dans l'état de l'art pour montrer l'importance du traitement configural lors de l'étape de reconnaissance et qui ont permis d'en préciser la définition. Ainsi, l'information configurale comprend à la fois l'information relationnelle correspondant aux distances entre les traits du visage mais aussi l'information holistique (ou globale) qui permet de considérer l'ensemble des traits du visage comme un tout.

L'étude de certains patients atteints de pathologies rares a permis de mettre en évidence cette utilisation combinée d'informations : à la fois configurale et basée sur les traits du visage eux mêmes. La prosopagnosie en fait partie. Elle résulte de lésions cérébrales localisées apparaissant généralement après un accident. Ce trouble neurologique empêche les patients atteints de reconnaître des personnes à partir de leur visage, y compris des

personnes de leur entourage voire leur propre visage. Généralement les personnes atteintes doivent développer d'autres stratégies pour inférer l'identité d'une personne, notamment en se basant sur la voix ou en utilisant des caractéristiques propres à chacun d'entre nous (la façon de marcher, de rire...). Dans [RB08], les auteurs expliquent que les patients prosopagnosiques font une analyse du visage trait par trait en portant une attention particulière sur chacun d'entre eux tandis qu'il suffit à un patient sain de poser son regard au centre d'un visage pour l'identifier. Autrement dit, les patients prosopagnosiques ont perdu la capacité à traiter l'information globale d'un visage et ne peuvent se baser que sur les caractéristiques locales. Le test de « l'illusion des visages composites » originellement proposé par Young et al. [YHH87] permet d'illustrer l'utilisation de ces deux types d'information lors de l'étape de reconnaissance. Cette expérience consiste à présenter au patient des visages dits composites car la partie supérieure appartient à un visage A tandis que la partie inférieure appartient à un autre visage B. Le but étant de reconnaître à quelle personne la partie supérieure du visage appartient. Ce test utilisé à de nombreuses reprises dans la littérature possède plusieurs variantes. Notamment la **Figure 1.2** tirée de [RB08] illustre ce test dans le cas où les moitiés supérieures sont toutes identiques, seules les moitiés inférieures varient.

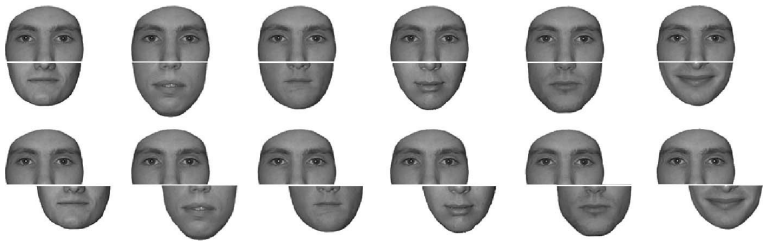

Figure 1.2 – *Illusion des visages composites : les visages présentés sont divisés en deux parties. La partie supérieure ne varie pas, la partie inférieure correspond à un visage différent à chaque fois. Pour un sujet sain, il est généralement difficile de voir que la moitié supérieure du visage appartient toujours à la même personne alors que ce n'est pas le cas pour un patient prosopagnosique. Ne pouvant traiter l'information globale d'un visage, ce dernier n'a en effet aucun mal à faire la distinction entre les deux parties du visage.[RB08]*

Lorsque les deux parties sont parfaitement alignées, on constate que la partie inférieure perturbe fortement la perception de la moitié supérieure qui apparaît différente pour chaque cas alors qu'elle est toujours identique. Cette illusion disparaît lorsque les deux parties ne sont plus alignées. Cette expérience a été menée avec des patients ne présentant aucune pathologie neurologique et des patients prosopagnosiques. Seuls les sujets sains étaient perturbés par l'illusion. Les autres arrivaient parfaitement à percevoir les visages identiques ce qui s'explique par leur incapacité à traiter l'information de façon globale.

Il est donc admis dans la littérature que la reconnaissance d'une personne nécessite une combinaison des informations locales et globales mais aucune étude n'a encore permis de comprendre totalement comment ces informations sont traitées par le système visuel

humain. En terme de reconnaissance automatique de visages, nous pouvons faire le même constat. En effet, nous pouvons classer dans un premier temps des méthodes selon deux catégories : celles se servant des caractéristiques globales du visage et celles se servant des caractéristiques locales du visage. Comme on le verra dans l'état de l'art, de nombreuses méthodes ont été développées dans chacune de ces deux catégories. Les deux approches présentent en effet des avantages complémentaires ce qui amène à penser qu'une combinaison des deux approches pourrait permettre de s'affranchir des problèmes rencontrées dans les deux catégories. Cela nous amène dans un deuxième temps à la troisième catégorie de méthode de reconnaissance : les méthodes hybrides qui se servent d'une combinaison des informations locales et globales. Malheureusement, ce type de méthode n'en est encore qu'à ses débuts et peu de méthodes ont jusqu'à maintenant été proposées [TChZFZ06]. La façon de combiner les deux méthodes n'est pas maîtrisée non plus.

2. État de l'art des techniques 2D de reconnaissance de visage

Il existe d'excellentes synthèses sur le sujet [ZCPR03] et [TChZFZ06], et le but n'est pas ici de faire un résumé exhaustif de l'ensemble des méthodes déjà existantes dans le domaine. Nous nous intéresserons plus particulièrement à l'identification de visages à partir d'images en niveaux de gris, les images issues de caméras vidéo surveillance étant souvent en noir et blanc, en particulier celles acquises de nuit, et nous présenterons plus spécifiquement dans ce chapitre les méthodes de reconnaissance qui ont conduits à une avancée notable dans le domaine.

La première catégorie regroupe les méthodes d'identification utilisant l'ensemble du visage comme l'information à traiter par le système. Ces méthodes sont souvent appelées *méthodes globales* ou *holistiques* et visent généralement à réduire l'espace de représentation du visage. On peut citer, parmi les plus connues, l'algorithme EigenFaces [TP91], FisherFaces [BHK97], [MK01] ou encore l'analyse en composantes indépendantes (ACI) [BMS02] et l'évolution poursuite [LW00].

La deuxième catégorie regroupe les méthodes d'identification basées sur des caractéristiques particulières du visage (les traits). On les appelle les *méthodes locales*. Parmi les méthodes les plus répandues et appartenant à cette catégorie, on peut citer les méthodes basées sur le descripteur LBP [AHP04], les méthodes basées sur les ondelettes de Gabor [SB06], [LW02], la méthode EBGM [WFKvdM99] mais également la méthode présentée dans [VC10] où un nouveau descripteur de visage est proposé.

2.1 Méthodes Globales

2.1.1 Analyse en Composantes Principales (ACP)

L'ACP est une méthode permettant d'extraire efficacement de l'information au sein d'un jeu de données souvent complexe en réduisant la dimension de l'espace dans lequel ces données sont observées et en les arrangeant dans un nouvel espace de façon à mettre en évidence l'information utile et à éliminer celle qui est secondaire. Il existe plusieurs bases possibles pour ré-exprimer ces données dans un tel espace. Pour simplifier la résolution de ce problème, l'ACP impose à la fois une hypothèse de linéarité : les données d'entrée $\mathbf{X}$ et de sortie $\mathbf{Y}$ sont reliées via une transformation linéaire $\mathbf{P}$ telle que : $PX = Y$ mais aussi d'orthogonalité des vecteurs de la nouvelle base : $\mathbf{P}$ est alors une matrice orthonormale. Ainsi, cette méthode consiste à trouver une nouvelle base, combinaison linéaire des vecteurs de la base originale, dont les vecteurs de la nouvelle base sont tous orthogonaux entre eux. La première composante principale correspond à la direction de variance maximum des données d'apprentissage. La seconde composante est déterminée grâce à la contrainte d'orthogonalité et est associée à la deuxième plus grande variance des données d'apprentissage et ainsi de suite. Autrement dit, ce sont les composantes principales associées aux variances les plus grandes qui portent l'information tandis que celles associées aux variances les plus faibles représentent le bruit. En ne gardant que les "k" premières composantes, l'information utile est préservée et les données sont exprimées dans un espace de plus petite dimension.

L'algorithme ACP adapté à l'analyse et l'identification de visage est connu sous le nom de EigenFaces (visages propres) et a été développé par M.A. Turk et A.P Pentland en 1991 [TP91]. Il se divise en une phase d'apprentissage et une phase de classification. Au cours de la phase d'apprentissage, un espace propre est construit à partir d'une base d'apprentissage en utilisant la méthode ACP puis ces mêmes images sont projetées sur l'espace ainsi obtenu. Durant la phase de classification, un visage test est projeté à son tour sur ce même espace pour être alors identifié en le comparant aux projections de chacun des visages de la base d'apprentissage. Cela revient à projeter les images sur une base orthogonale de vecteurs particuliers qui présentent les caractéristiques les plus indépendantes possible des visages (la redondance a donc été éliminée) pour mettre en évidence leurs différences. En termes mathématiques, cela revient à considérer une image de taille $N = n \times m$ comme un vecteur de dimension N. Autrement dit on représente ce visage comme un point dans un espace de dimension N. Soit Γ_i le vecteur de dimension N correspondant à une image "i" de la base d'apprentissage composée de M images. Soit Ψ l'image moyenne de cette base d'apprentissage définie par :

$$\Psi = \frac{1}{M} \sum_{i=1}^{M} \Gamma_i \ . \tag{1.1}$$

Soit ϕ_i un vecteur de dimension N correspondant à l'image i dont on a soustrait l'image moyenne telle que :

$$\phi_i = \Gamma_i - \Psi. \tag{1.2}$$

Étant donnée la dimension de chacun de ces vecteurs, il serait difficile d'implanter un quelconque algorithme sans les projeter sur une base adaptée. C'est là tout l'intérêt de la méthode ACP qui permet de trouver un ensemble de K vecteurs orthogonaux P_j décrivant de façon adaptée la distribution de ces visages en réalisant une combinaison linéaire des vecteurs de la base originale. Pour cela, on souhaite éliminer la redondance d'information traduite en termes mathématiques par la matrice de covariance "C_x" définie comme une matrice de dimension $N \times N$:

$$C_x = \sum_{i=1}^{M} \phi_i \phi_i^t = XX^t. \tag{1.3}$$

Où X est la matrice de dimension $N \times M$ telle que : $X = [\phi_1, \phi_2, ..., \phi_M]$.

Diminuer la redondance d'information est équivalent à dire que la covariance entre deux images doit être la plus petite possible ce qui revient à diagonaliser C_x. (Pour plus de détails sur les calculs, se référer à **l'annexe A**). L'ACP suppose que plus la valeur d'une variance est élevée plus elle traduit la différence existant entre deux images. Autrement dit, les directions associées aux variances élevées sont gardées et correspondent aux directions principales. Les directions principales ne sont rien d'autre que les M vecteurs propres "v_i" de la matrice "A" définie par :

$$A = X^t X. \tag{1.4}$$

Ces vecteurs forment ainsi une combinaison linéaire des M images de la base d'apprentissage pour former les EigenFaces $X.v_i$ également vecteurs propres de C_x. On montre sur la **Figure 1.3** un exemple d'EigenFaces obtenus sur la base Feret [PMRR97].

Figure 1.3 – *Illustration de plusieurs Eigenfaces obtenus sur la base Feret [PMRR97].*

En pratique on réorganise les valeurs propres de "C_x" dans l'ordre croissant pour n'en garder que les M' ($M' < M$) valeurs les plus élevées. On en déduit les M' directions principales associées aux variances des données d'apprentissage les plus élevées et qui correspondent aux vecteurs propres associés. La valeur M' varie en fonction de la base d'apprentissage utilisée. Il convient donc, dans un premier temps, de calculer l'ensemble des valeurs propres pour déterminer celles dont les valeurs sont suffisamment importantes pour être considérées par l'algorithme. La décision finale sur la valeur de M' revient à l'utilisateur. En effet, il n'existe pas vraiment de règle concernant ce choix. Généralement, celui-ci est déterminé en calculant la concentration d'information (en pourcentage) contenue par chaque valeur propre et en ne gardant que les plus significatives. Le tracé d'une courbe de pourcentage cumulé peut ainsi faciliter la prise de décision (un seuil peut être fixé au point de flexion de la courbe). Également souvent utilisé, le critère empirique de

Kaiser permet quant à lui de ne garder que les composantes principales dont la valeur propre est supérieure à 1.

Une fois le nombre de composantes principales fixé, il reste enfin à projeter les images Φ_i sur l'espace des visages E_v ainsi formé. E_v est donné par la matrice $E_v = [X.v_1, X.v_2, ..., X.v_{M'}]$, la matrice de projection Ω_k qui décrit la $k^{ième}$ classe de visage est définie par : $\Omega_k^t = [\omega_{k_1}, \omega_{k_2}, ..., \omega_{k_{M'}}]$. Enfin, le projeté Φ_p sur E_v d'une image Φ (ramenée à sa moyenne) est donnée par :

$$\phi_p = \sum_{i=1}^{M'} \omega_i X.v_i. \tag{1.5}$$

Pour l'identification, l'idée est de trouver le $k^{ème}$ visage de la base d'apprentissage qui minimise la distance euclidienne ϵ entre le vecteur de projection Ω_k et celui de l'image test Ω_t : $\epsilon = ||\Omega_t - \Omega_k||$. Cette distance peut être seuillée pour minimiser les erreurs de classification. Pour augmenter la pertinence de la reconnaissance, M.A. Turk et A.P Pentland [TP91] proposent également de considérer la distance à l'espace des visages (DFFS ou Distance From Face Space) comme un critère de décision. Plus cette distance est courte, plus le visage que l'on cherche à reconnaître a de chance d'être correctement reconnu. Quatre cas différents peuvent être rencontrés comme cela est illustré sur la **Figure 1.4**. Le premier cas représente un visage correctement reconnu. La distance ϵ est suffisamment petite pour considérer la projection appartenant à la classe 1. Dans le second cas le visage n'est pas reconnu car la projection n'appartient à aucune des classes. La distance ϵ calculée pour chacune des trois classes représentées est trop importante pour pouvoir considérer ce visage comme appartenant à une de ces classes. Pour le troisième et le quatrième cas, la distance à l'espace des visages est importante contrairement aux deux premiers cas. Le troisième cas illustre cependant un cas de *faux positif* : la DFFS est importante mais la distance ϵ est suffisamment courte pour considérer ce visage comme appartenant à la classe 3.

De nombreuses techniques d'identification de personne ont par la suite été développées pour améliorer les performances de l'algorithme Eigenfaces. Les méthodes basées sur l'analyse discriminante linéaire en font notamment partie. L'idée est de trouver les directions de projection les plus discriminantes dans l'espace des visages.

2.1.2 Analyse Discriminante Linéaire (Linear Discriminative Analysis : LDA)

L'algorithme LDA adapté à l'analyse et l'identification de visage est connu sous le nom de FisherFaces et a été développé par Belhumeur et al. à l'université de Yale aux USA en 1997 [BHK97]. Contrairement à l'algorithme précédent qui permet d'extraire des caractéristiques particulières à chaque image, celui-ci permet de réaliser une véritable séparation de classes [MK01]. Il utilise en effet une étiquette de classe associée à chacune des variables lors de l'apprentissage. Cette technique est donc basée sur un apprentissage supervisée, c'est à dire que l'on dispose cette fois-ci d'informations supplémentaires concernant les

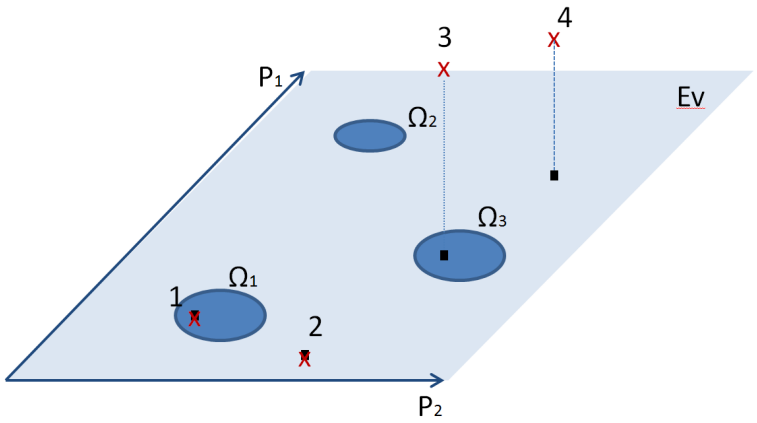

Figure 1.4 – *Illustration des quatre cas pouvant être rencontrés après projection d'une image sur l'espace des visages E_v. Schéma adapté de [TP91].*

données d'apprentissage qui doit nous permettre de réaliser une classification de ces données et de trouver la classe à laquelle appartient toute nouvelle observation. Cela demande donc de diviser au préalable la base d'apprentissage en c classes différentes. Autrement dit, chaque personne de cette base est équivalent à une classe et à chacune d'entre-elles est associée au moins deux images. L'objectif de cet algorithme est cette fois de maximiser le rapport entre les variations inter-classe (les variations entre les images de personnes différentes) et les variations intra-classe (les variations entre les images d'une même personne). Ainsi, comme précédemment, la méthode FisherFace consiste à trouver un espace adéquat sur lequel vont être projetées les images de la base d'apprentissage tout comme celles de la base test. L'identification est réalisée en comparant la projection de l'image test avec chacune des projections des images de la base d'apprentissage. Comme précédemment Γ_i est le vecteur de dimension N correspondant à une image i de la base d'apprentissage, laquelle est composée de M images.

Ψ_{c_i} représente l'image moyenne de la classe c_i et est définie par :

$$\Psi_{c_i} = \frac{1}{N_c} \sum_{i=1}^{N_c} \Gamma_i. \tag{1.6}$$

où N_c correspond au nombre d'images de la classe c_i. Ψ représente l'image moyenne de toutes les classes c_i et est définie par :

$$\Psi = \frac{1}{c} \sum_{i=1}^{c} N_c \Psi_{c_i}. \tag{1.7}$$

ϕ_i est un vecteur de dimension N correspondant à l'image i dont on a soustrait l'image moyenne de la classe c_i : $\phi_i = \Gamma_i - \Psi_{c_i}$

Les variations inter-classe S_b et intra-classe S_w sont quant à elles définies comme suit :

$$S_b = \sum_{i=1}^{N_c} N_c(\Psi_{c_i} - \Psi)(\Psi_{c_i} - \Psi)^t. \tag{1.8}$$

$$S_w = \sum_{i=1}^{c} \sum_{\Gamma_k \in c}^{N_c} N_c(\Gamma_k - \Psi_{c_i})(\Gamma_k - \Psi_{c_i})^t. \tag{1.9}$$

On souhaite donc maximiser le rapport R défini par :

$$R = \frac{\zeta^t S_b \zeta}{\zeta^t S_w \zeta}. \tag{1.10}$$

ce qui revient à chercher un vecteur $\zeta^m = [\zeta_1^m, \zeta_2^m, ..., \zeta_{M'}^m]$ maximisant le critère d'optimisation de Fisher suivant :

$$\zeta^m = \arg\max_{\zeta} \frac{\zeta^t S_b \zeta}{\zeta^t S_w \zeta}. \tag{1.11}$$

ce qui est équivalent à chercher les vecteurs propres de la matrice supposée symétrique $S_w^{-1} S_b$ satisfaisant l'équation aux valeurs propres suivante :

$$S_w^{-1} S_b . \lambda = w . \lambda \tag{1.12}$$

Une fois le vecteur ζ^m identifié, on procède comme pour la méthode ACP en projetant chacune des images de la base d'apprentissage sur le nouvel espace ainsi formé. On fait de même pour les images de la base test puis on réalise l'identification en minimisant la distance euclidienne ϵ entre la projection de l'image test sur « ζ^m » et la projection de l'image de la base d'apprentissage sur ce même espace.

Cette méthode est utilisée efficacement dans de nombreux problèmes de classification et de réduction de dimension. Cependant, dans le cas où les données sont de trop grandes dimensions, il n'est pas possible d'appliquer directement cette méthode sur les images sans diminuer au préalable la dimension des données [ZCPR03]. Dans ce cas, au lieu d'utiliser directement la valeur des pixels des images, une ACP est premièrement appliquée sur les données et c'est la représentation des images dans l'espace des visages qui est utilisée [Zha02].

2.1.3 Méthode Bayésienne et espace probabiliste des visages

Dans [Mog00], les auteurs proposent une généralisation de la méthode LDA en utilisant un classificateur Bayésien pour classer les données obtenues après projection dans l'espace des visages. Pour cela, les auteurs considèrent non plus une image I donnée mais la différence $\Delta = I_1 - I_2$ entre deux images I_1 et I_2 supposées mieux représenter les variations d'apparence d'un individu. En définissant uniquement les classes ϱ_i correspondant aux variations intra-personnelles et ϱ_e correspondant aux variations extra-personnelles, le problème d'identification devient un simple problème de classification binaire. La mesure

de similarité, où $P(\varrho_i|\Delta)$ est la probabilité intra-personnelle a posteriori, est alors donnée par la loi de Bayes :

$$S(I_1 - I_2) = P(\varrho_i|\Delta) = \frac{P(\Delta|\varrho_i)P(\varrho_i)}{P(\Delta|\varrho_i)P(\varrho_i) + P(\Delta|\varrho_e)P(\varrho_e)} \tag{1.13}$$

La difficulté étant alors d'exprimer les deux densités de probabilité $P(\varrho_i|\Delta)$ et $P(\varrho_e|\Delta)$. Pour cela, les auteurs utilisent une méthode de représentation des données développée par Moghaddam et Pentland dans [MP97]. Celle-ci permet de décomposer l'espace de représentation en deux parties orthogonales : l'espace des visages F, formé de M' vecteurs propres et l'espace orthogonal à F, F', formé de tous les vecteurs propres restant qui ne sont pas utilisés d'ordinaire dans la méthode ACP et qui représentent généralement le bruit. A partir de ce type de représentation, les auteurs ont pu déterminer un estimateur pour chacune des deux densités de probabilité et les utiliser pour classer leurs images et procéder à l'identification.

2.1.4 Analyse en Composantes Indépendantes (ACI)

L'analyse en composantes indépendantes peut être assimilée à un problème de séparation de sources comme initialement formulé dans [JH91] dans le sens où elle permet d'extraire les structures fondamentales d'une image. Cette méthode, appliquée au problème d'identification de visages, peut également être vue comme une généralisation de la méthode ACP car elle permet non seulement de minimiser les dépendances statistiques de second ordre (covariance) mais également celles d'ordre supérieur. Tout comme l'ACP, l'ACI permet une projection linéaire des données dans un espace de plus petite dimension, mais cet espace, contrairement à l'espace des visages, n'est pas nécessairement orthogonal et permet une meilleure représentation des données [BMS02]. L'approche consiste donc à considérer une matrice X (les images de visage) comme étant une combinaison linéaire et indépendante des sources « s » telle que :

$$x^t = As^t \tag{1.14}$$

où A est la matrice de mélange. On peut également définir une matrice de séparation W qui permet à partir des images X (observations) d'estimer les sources s telles que :

$$u^t = Wx^t = WAs^t \tag{1.15}$$

où u correspond à l'estimation de la source s. Le but de l'ACI est donc de trouver une estimation de la matrice de mélange A ou de séparation W ainsi qu'une estimation de la matrice des sources $\mathcal{S}$ en réduisant au minimum la dépendance de ses composantes. La difficulté étant de trouver une fonction permettant de mesurer cette dépendance et de la minimiser. Dans [BMS02], les auteurs ont utilisé un algorithme *Infomax* qui s'inspire entre autre du traitement de l'information visuelle chez les vertébrés [BS95]. Dans ce même article, on retrouve l'image d'un visage en faisant une combinaison linéaire des images de base estimées lors d'une phase d'apprentissage. La représentation d'une image par l'ACI correspond donc aux coefficients de cette combinaison linéaire. En proposant deux

architectures différentes pour exprimer la matrice des observations X, les auteurs arrivent à obtenir deux types d'images de base ce qui leur permet d'obtenir deux représentations différentes d'une image à partir de l'ACI. Ces deux architectures sont présentées sur la **Figure 1.5**. Il est intéressant de noter que la première architecture permet d'extraire des structures de base telles que les yeux, le nez ou la bouche, caractéristiques principales d'un visage comme illustrée sur la **Figure 1.6**.

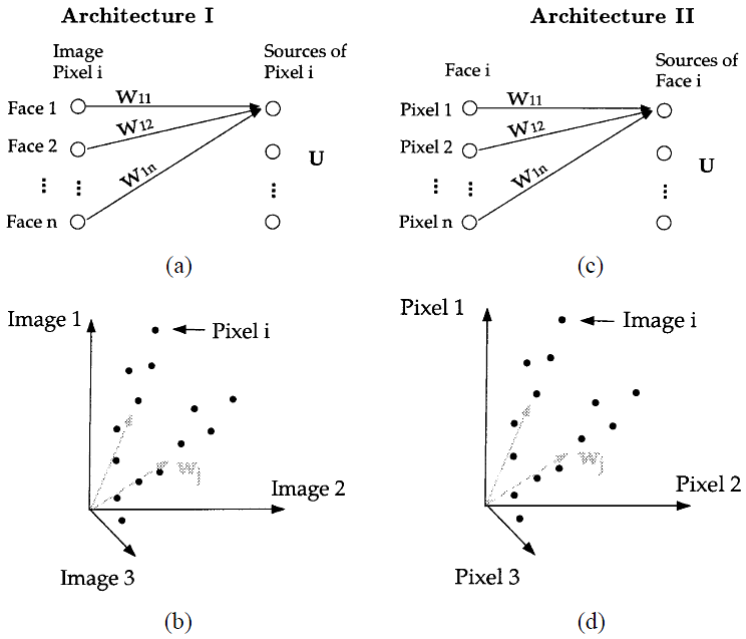

Figure 1.5 – *Illustration des deux architectures proposées pour l'application de l'ACI à la reconnaissance de visage dans [BMS02]. La première architecture considère que les images sont des variables aléatoires alors que la seconde architecture considère les pixels comme étant des variables aléatoires.*

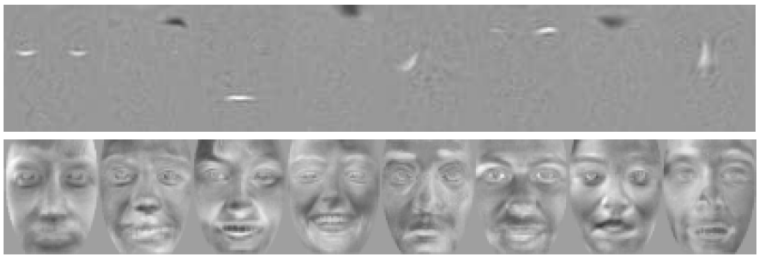

Figure 1.6 – *Images de base obtenues avec la méthodes ACI pour l'architecture 1 en haut et pour l'architecture 2 en bas. [DBBB03].*

2.1.5 Autres méthodes

Toutes les méthodes que nous venons de citer se basent sur l'hypothèse que les variables (les visages) sont linéairement séparables dans l'espace de représentation. Or ce n'est généralement pas le cas. Par conséquent de nombreux auteurs ont étendu les différentes approches, notamment l'ACP et l'ALD, à une version non linéaire en introduisant divers fonctions noyaux (kernel). L'idée est d'exprimer dans un premier temps les variables dans un nouvel espace de représentation par transformation non linéaire et d'appliquer ensuite la méthode dans l'espace trouvé. De meilleurs résultats ont ainsi été obtenus avec KACP [SSM98] et l'algorithme Fisherface combiné à une fonction noyau (Kernel Fisherfaces algorithm) [Yan02]. Un autre problème rencontré avec ce type de méthodes est qu'elles nécessitent plusieurs échantillons d'images par personne pour pouvoir palier les variations éventuelles des visages étudiés (variation d'illumination, de pose etc.). Une des approches utilisées pour remédier partiellement à ce problème est d'introduire plus d'information sur l'image en construisant une nouvelle représentation de celle-ci. Dans [WZ02], les auteurs utilisent les projections horizontales et verticales de l'image pour faire ressortir les propriétés de l'image utiles à la reconnaissance. Celles-ci permettent ensuite de synthétiser une autre image qui sera utilisée lors de l'étape d'identification. Pour la méthode Fisherfaces, un des problèmes principaux réside dans le fait qu'il est impossible de calculer les variations intra-classe lorsqu'il n'y a qu'une image par personne à disposition. Dans [WPV05], les auteurs utilisent donc une base d'apprentissage dans laquelle ils disposent de plusieurs images par personne qui leur permettent de trouver une valeur générique de la variation intra-classe.

2.1.6 Conclusion

Les méthodes holistiques permettent de représenter un visage dans sa globalité en prenant en compte sa texture et son apparence. Que ce soit pour la ACP, LDA, ACI ou bien l'évolution poursuite dont le principe est intimement lié à ces trois méthodes, le but est de décrire l'image dans un espace de plus petite dimension. Néanmoins, l'identification d'un visage ne peut se réduire à un problème de classification linéaire et bien qu'il existe des algorithmes intégrant des fonctions noyaux qui permettent la mise en œuvre de classification non linéaire, ils ne permettent pas pour autant de s'affranchir de tous les problèmes. Ces méthodes restent en particulier très sensibles à de petites variations d'apparence du visage, notamment liées aux changements d'illumination, de pose ou d'expression et en particulier lorsque la base d'apprentissage n'est composée que d'une seule image par visage. L'algorithme est alors d'autant plus sensible aux variations qu'il n'est pas capable de distinguer le cas où deux visages sont issus de deux individus différents et le cas où les deux visages appartiennent à un même individu mais pris dans des conditions différentes. Pour palier ce type de problème, il est généralement plus adapté de faire appel à des méthodes dites « locales » qui permettent de caractériser un visage de façon multiple sans avoir la contrainte liée à la dimension des vecteurs de caractéristiques.

2.2 Méthodes Locales

Contrairement aux méthodes globales, les méthodes locales permettent de caractériser localement un visage par des vecteurs de caractéristiques de petite dimension ce qui permet de s'affranchir de tous les problèmes de dimensionnalité rencontrés avec les méthodes holistiques. D'autre part, elles permettent de représenter un visage grâce à des caractéristiques multiples beaucoup plus robustes à certaines variations comme les variations d'illumination par exemple [AHP04]. Il existe deux stratégies pour cela : les méthodes basées sur des caractéristiques locales (Local feature-based method) et les méthodes basées sur l'apparence locale (local appearance-based method) du visage. Les premières consistent à détecter dans un premier temps les traits du visage tels que les yeux, le nez, la bouche... puis à en extraire des caractéristiques. Généralement, celles-ci correspondent aux propriétés des traits eux-mêmes ou bien aux relations pouvant exister entre eux (angles, distances...). Les secondes consistent à extraire directement les caractéristiques locales au niveau de régions prédéfinies du visage. Ces régions sont généralement obtenues en découpant simplement l'image en patchs de dimension égale et de forme équivalente.

2.2.1 Local feature-based methods

Pour extraire les caractéristiques géométriques, ce type de méthode doit dans un premier temps isoler certains traits remarquables du visage. Ce sont principalement les yeux, le nez et la bouche mais on peut également trouver le contour du visage, les sourcils, le menton.... Plusieurs techniques ont été développées depuis une vingtaine d'années à cet effet. Dans [BP93], les auteurs se basent sur une méthode de projection intégrale pour extraire automatiquement 35 caractéristiques du visage (positions particulières du visage, angles entre deux traits distincts etc...). L'utilisation de projections d'intégrales sur plusieurs gradients de l'image (typiquement gradient horizontal et gradient vertical) permet en effet de localiser des caractéristiques particulières du visage. Par exemple, les pics d'une projection horizontale du gradient vertical de l'image permettent de détecter la position (plus ou moins approximative) du nez. Néanmoins, bien que robuste aux variations d'illumination, cette méthode de détection est généralement peu stable, en particulier lorsque les visages sont pris dans des conditions très variables. D'autres méthodes de détection plus robustes ont donc été proposées. Celles-ci sont principalement les détecteurs de Viola et Jones [VJ04] et de Rowley [RBK98] basés sur l'utilisation de filtres de Haar pour l'un et sur un système de réseaux de neurones pour l'autre. Ces détecteurs sont communément utilisés pour la détection de visage dans une image. Malheureusement, bien que les performances de ces méthodes de détection soient largement démontrées, les méthodes d'identification de visages basées uniquement sur les caractéristiques géométriques ne sont pas suffisamment fiables et ne permettent pas le développement d'algorithmes de reconnaissance suffisamment robustes.

Les méthodes d'identification basées sur une représentation des caractéristiques du visage par déformation de graphes présentent en effet de meilleurs résultats. Ces méthodes [Man92], [LVB+93], [WFKvdM99], [GQ05] visent à représenter une image à l'aide d'un graphe formé par un ensemble de nœuds reliés entre eux par des arêtes. Dans [Man92], les

auteurs proposent d'identifier un visage en représentant dans un premier temps l'image par des points caractéristiques détectés localement à l'aide d'ondelettes de Gabor et correspondant entre-autre à des changements de courbure dans l'image. Les points remarquables ainsi détectés correspondent aux nœuds du graphe caractérisés à la fois par leur position dans l'image et par un vecteur de caractéristiques calculé au voisinage de ce point. L'information sur la topologie globale du graphe est obtenue en prenant en compte la distance euclidienne $d_{i,j}$ séparant deux nœuds voisins i et j. Une fois la topologie du graphe formulée, la reconnaissance se fait en comparant le graphe de l'image d'entrée I avec l'ensemble des graphes (ou modèles) contenus dans la galerie qui est la base contenant les images de visages à reconnaître. Le visage associé au modèle O^* qui minimise la fonction de coût $C(I, O)$ correspond au visage à identifier. Bien que cette fonction de coût prenne en compte à la fois les similarités entre les caractéristiques de chaque nœud et les similarités sur la topologie globale de chaque graphe (par comparaison des distances $d_{i,j}$), ce type de modèle n'est pas suffisamment robuste aux variations qui peuvent modifier l'apparence du visage. En effet, une fois la topologie du graphe réalisée, il n'est plus possible de le modifier. Pourtant, un visage pris de face et un visage avec une pose différente ne présentent pas la même topologie. La correspondance des deux images s'en trouve faussée. Afin de palier ce problème, une méthode basée sur une Architecture de Liens Dynamiques (Dynamic Link Architecture ou DLA) a été proposée dans [LVB+93]. Une image est cette fois-ci représentée par un graphe avec une topologie de type grille déformable comme on peut le voir sur la **Figure 1.7** .

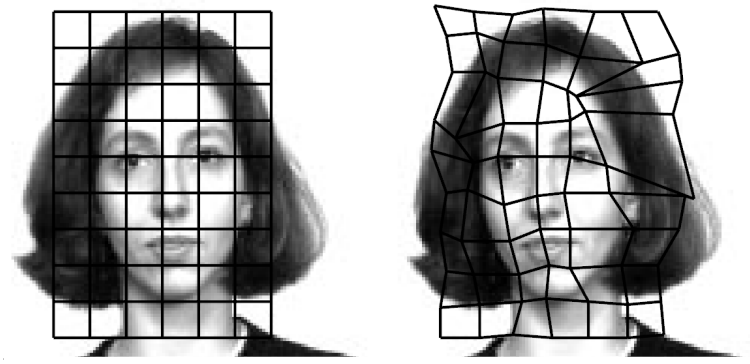

Figure 1.7 – *Dans un premier temps le graphe est initialisé à la même position que l'image contenue dans la galerie (a) puis, les paramètres du graphe sont ensuite ajustés pour minimiser la fonction de coût totale C_{tot}. L'image de gauche, (b) représente le graphe obtenu après minimisation. [LVB+93]*

A chaque nœud de la grille est associé un jeu de coefficients d'ondelettes de Gabor prises à différentes échelles et orientations que l'on appelle *Jet*. On explicitera leur fonctionnement dans le paragraphee suivant. Pour plus de robustesse face aux variations d'illumination, seule l'amplitude des ondelettes est prise en compte. De la même façon que dans [Man92], les auteurs prennent en compte aussi bien l'information portée par chacun des

nœuds du graphe mais aussi l'information globale en considérant les liens (ou distances) entre chacun de ces nœuds. Durant l'étape de reconnaissance, on souhaite faire correspondre le graphe de l'image I d'entrée avec un graphe de la galerie donné. Pour cela, dans un premier temps, le graphe de l'image I est positionné différemment dans l'image tout en gardant une topologie fixe. La position sélectionnée est celle qui minimise une fonction de côut mesurant les similarités entre les jets J de l'image I et ceux du modèle M. Dans un second temps, les liens l entre chaque nœud, ainsi que les nœuds n eux mêmes peuvent être déplacés également jusqu'à minimisation d'une fonction de coût totale C_{tot}. Cette dernière est définie pour un jeu de nœuds $\{x_i\}$ donné et prend en compte à la fois les similarités S_n entre jets, mais également la variation des distances entre chaque nœud S_l sous forme de deux fonctions de coût C_n et C_l respectivement telles que :

$$C_{tot}(\{x_i\}) = \lambda.C_l + C_n. \tag{1.16}$$

$$C_{tot}(\{x_i\}) = \lambda. \sum_{i,j \in L} S_l(\overrightarrow{\Delta}^I_{ij}, \overrightarrow{\Delta}^M_{ij}) - \sum_{i \in N} S_n(J^I(x^I_i), J^M_i). \tag{1.17}$$

où λ est un coefficient permettant de contrôler la rigidité du graphe : le choix d'une grande valeur sera donc pénalisant.

Cette procédure est répétée pour tous les modèles de la galerie, celui pour lequel la fonction de coût totale est la plus faible est identifiée comme le visage reconnu. Cette méthode, basée sur la DLA, est maintenant beaucoup plus connue sous le terme d'Elastic Bunch Graph Matching (EBGM) [WFKvdM99] où plusieurs améliorations ont été faites. Tout comme dans [Man92], les auteurs ont notamment fait correspondre les nœuds du réseau à des caractéristiques particulières du visage, telles que les yeux, le nez ou la bouche. D'autre part, pour pouvoir prendre en compte une plus grande variabilité des visages tout en évitant d'associer à chaque image un graphe, les auteurs ont introduit la notion de *Face Bunch graphe* ou FBG comme représentée sur la **Figure 1.8** . Un FBG est une représentation générale du visage pour laquelle on associe à chaque nœud, un ensemble de jets (Bunch) représentant toutes les variabilités possibles de ce nœud. Au niveau d'un œil par exemple, on peut noter ou non la présence de lunettes, d'un œil fermé etc., il y aura donc autant de jets que de combinaisons possibles pour cet œil. Lors de la phase de représentation d'une image, un seul jet est retenu pour chacun des bunchs : celui qui maximise la fonction de similarité avec le FBG. Bien que robuste, cette méthode nécessite un temps de calcul relativement long et ne peut s'adapter à tous les cas de reconnaissance, notamment à ceux présentant des occultations partielles du visage, le nombre de nœuds étant défini à l'avance. Par ailleurs, elle est fortement tributaire de l'efficacité des algorithmes de détection qui ne sont, jusqu'à maintenant, pas encore suffisamment robustes aux différentes variations de luminosité ou de pose usuellement rencontrées.

Il existe des méthodes plus récentes dans l'état de l'art qui ne nécessitent pas un nombre fixe de points (nœuds) à détecter. Dans [GQ05] les auteurs ont développé une nouvelle méthode de représentation des visages basée sur la détection des points remarquables de

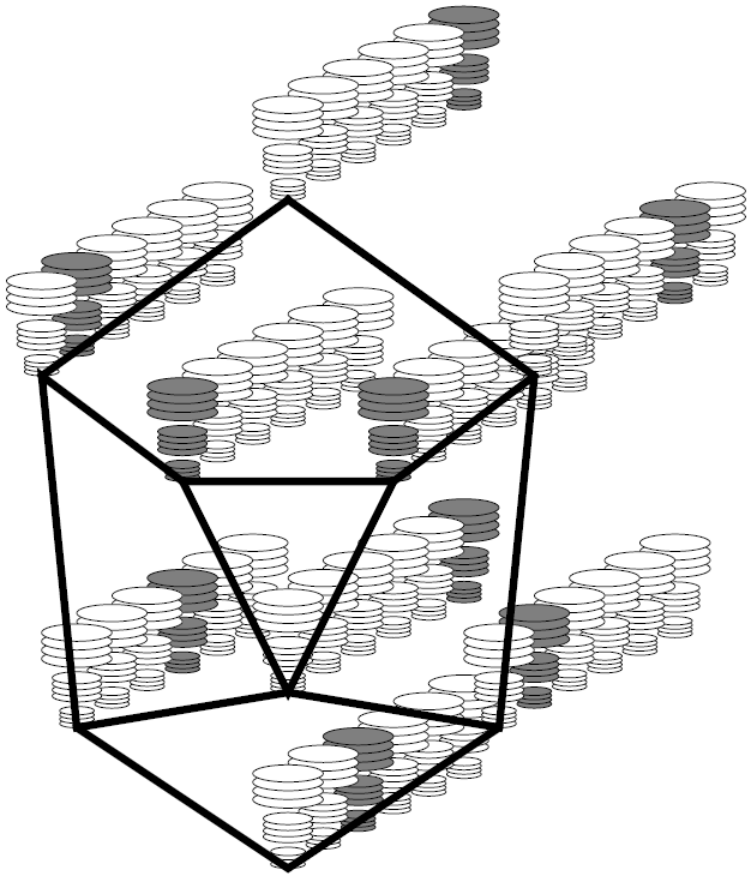

Figure 1.8 – *Représentation du Face Bunch Graph comme utilisé dans la méthode de reconnaissance faciale EBGM [WFKvdM99] .*

chaque courbe de l'image. En apprenant les différentes connections (angles) qui peuvent exister entre un point et ses plus proches voisins, les auteurs prennent également en compte la structure de l'image et forment ainsi un vecteur de caractéristiques qui leur sert à représenter l'image. Cependant, bien qu'il ne soit plus nécessaire de détecter des points particuliers du visage, cet algorithme reste fortement dépendant des variations du visage, notamment des changements d'expression. Ainsi, bien que relativement efficace dans le cas idéal, les performances de cette méthode chutent radicalement lorsqu'elle est confrontée à des images prises dans des conditions réelles car elles dépendent fortement de l'algorithme de détection des points caractéristiques. C'est pourquoi, d'autres approches plus récentes ne nécessitant pas une localisation précise de ces points ont récemment été proposées. L'ensemble de ces méthodes est regroupé sous le terme *d'appareance-based method* et certaines d'entre-elles sont présentées dans le paragraphe suivant.

2.2.2 Local appearance-based method

L'ensemble de ces méthodes peut se résumer en **4 étapes**. Toutes ne sont pas indispensables et certaines peuvent être négligées mais chacune présente un intérêt non négligeable pour l'obtention d'un algorithme de reconnaissance performant.

- la *première étape* consiste à **découper l'image en régions ou en patchs**. La taille de ces patchs ainsi que leur forme varient en fonction des publications mais en général elles sont rectangulaires avec ou sans chevauchement des unes avec les autres.
- Une fois l'image découpée, la *deuxième étape* consiste à **extraire les caractéristiques locales de l'image** et donc de chacune des régions ainsi définies. C'est une étape très importante qui va déterminer les performances du système de reconnaissance. Elle nécessite l'utilisation d'extracteurs de caractéristiques spécifiques en fonction de la tâche à réaliser. En effet, certains extracteurs sont plus sensibles que d'autres aux variations de l'image comme celles concernant l'illumination ou la pose par exemple. Dans le paragraphe suivant, on présente trois méthodes d'extraction de caractéristiques basées sur les ondelettes de Gabor [SB06], la méthode LBP [AHP04] connue pour être relativement robuste aux variations d'illumination et le descripteur Poem [VC10] robuste aux variations de pose.
- La *troisième étape* consiste à **sélectionner les caractéristiques les plus discriminantes**. Cela nécessite l'utilisation des algorithmes ACP et LDA. Cette étape peut être sautée en fonction de la tâche que l'on souhaite réaliser.
- Enfin, la *quatrième étape* correspond à l'étape de **classification** nécessaire à l'identification finale du visage. Elle peut correspondre à un simple calcul de distance entre les vecteurs de caractéristiques comme c'est le cas dans [AHP04] ou bien elle peut faire appel à des algorithmes de classification plus évolués comme les machines à vecteurs de supports (SVM) utilisées entre-autre dans [GLC00] et [HHP$^+$01]. Nous expliciterons cette technique plus en détails dans le chapitre 4 relatif à l'estimation de la pose.

Extraction des caractéristiques d'un visage par ondelettes de Gabor

La représentation à la fois en temps et en fréquence d'un signal permet de mieux visualiser son contenu par rapport à sa transformée de Fourier classique qui ne donne que sa représentation fréquentielle. Bien entendu, cela entraîne une contrepartie non négligeable puisque cela limite la représentation du signal aussi bien dans le domaine temporel que dans le domaine fréquentiel comme l'illustre le principe d'incertitude de Heisenberg :

$$\Delta t.\Delta f \geq \frac{1}{2} \tag{1.18}$$

Plus la fenêtre en temps Δt au niveau de laquelle on regarde le signal sera grande, plus la bande de fréquences Δf sera petite et donc précise. Cette inégalité souligne le fait qu'il n'est pas possible d'obtenir simultanément la localisation du signal en temps et en fréquence et qu'il faut faire un compromis pour obtenir une précision acceptable de ces deux grandeurs. Cette représentation permet de détecter des caractéristiques particulières dans une image tout en minimisant les effets liés aux variations d'illumination et de

pose [ZCPR03]. Par conséquent, la transformation en ondelettes s'avère très efficace pour l'extraction de caractéristiques faciales. En particulier, les filtres de Gabor possèdent des propriétés intéressantes aux sens biologique et mathématique du terme. Daugman [Dau85] a en effet montré que de tels filtres modélisent parfaitement la réponse spatiale à deux dimensions de cellules simples appartenant au cortex visuel. Par ailleurs, les ondelettes de Gabor correspondent aux fonctions qui minimisent le principe d'incertitude d'Heisenberg en donnant le meilleur rapport de résolution spatio-temporelle dans le domaine 2-D.

La fonction de Gabor est définie comme une fonction gaussienne modulée par un signal sinusoïdal. Autrement dit, ce n'est rien d'autre qu'un filtre passe-bande de forme gaussienne dont la fréquence centrale est la fréquence de la sinusoïde. De façon générale, la fonction de Gabor $G(x, y)$ dans le domaine spatial à deux dimensions est donnée par le produit d'une sinusoïde complexe $P(x, y)$, la porteuse, avec une fonction gaussienne $E(x, y)$, l'enveloppe, telle que :

$$G(x, y) = P(x, y).E(x, y) \tag{1.19}$$

Cependant, dans la majorité des papiers traitant de la reconnaissance de visage et utilisant les ondelettes de Gabor, la définition utilisée pour la représentation de telles fonctions est la suivante :

$$\psi_{\mu,\nu}(z) = \frac{\|k_{\mu,\nu}\|^2}{\sigma^2} e^{\frac{-\|k_{\mu,\nu}\|^2\|z\|^2}{2\sigma^2}} \left(e^{ik_{\mu,\nu}z} - e^{-\frac{\sigma^2}{2}} \right) . \tag{1.20}$$

Où μ et ν sont respectivement l'orientation et l'échelle du filtre de Gabor tandis que $k_{\mu,\nu}$ est le vecteur d'onde défini comme suit :

$$k_{\mu,\nu} = k_\nu e^{i\theta_\mu} \text{ où } k_\nu = \frac{f_{max}}{\sqrt{2}^\nu} \text{ et } \theta_\mu = \frac{\pi\mu}{8}. \tag{1.21}$$

f_{max} est la fréquence maximum du filtre et $\sqrt{2}$ est le facteur d'espacement entre les fréquences centrales. Pour obtenir cette forme d'ondelettes de Gabor, plusieurs hypothèses ont été faites. Celles-ci sont présentées en détails dans l'annexe B de cette thèse.

Pour finir, la représentation en ondelettes de Gabor $G_{\mu,\nu}$ d'un visage pour une orientation μ et une échelle ν est obtenue par convolution de l'image de visage $I(x, y) = I(z)$ avec le filtre de Gabor correspondant tel que :

$$G_{\mu,\nu} = I(z) \star \psi_{\mu,\nu}(z) \tag{1.22}$$

Cette image est généralement représentée par un jeu de 40 ondelettes [LW02], [SB06]. Dans ce travail, nous avons utilisé un jeu de 40 fonctions de Gabor correspondant à 8 orientations $\mu = \{0, 1, ..., 7\}$ et 5 échelles $\nu = \{0, 1, ..., 4\}$ différentes. Une illustration de cette représentation est donnée sur la **Figure 1.9**.

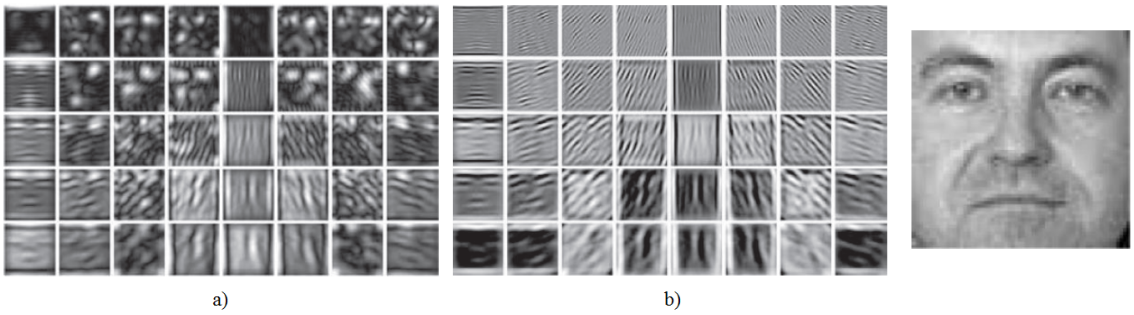

Figure 1.9 – *Représentation d'une image de visage avec 40 ondelettes de Gabor. a) Réponse en amplitude. b) Réponse en phase.*

Extraction des caractéristiques d'un visage avec la méthode LBP (Local Binary Patterns)

L'opérateur d'analyse de texture LBP a été introduit pour la première fois par Ojala et al. dans [OPH96]. Pour chacun des pixels d'une image, l'opérateur considère un voisinage de 3×3 pixels puis étiquette chacun des 8 voisins par un nombre binaire (0 ou 1), la valeur centrale de ces pixels servant de seuil. Autrement dit, tous les pixels dont la valeur de niveau de gris est supérieure ou égale à celle du pixel central se verront attribuer le chiffre un et le chiffre zéro pour tous les autres. Un schéma résumant les étapes principales de cet opérateur est donné sur la **Figure 1.10**.

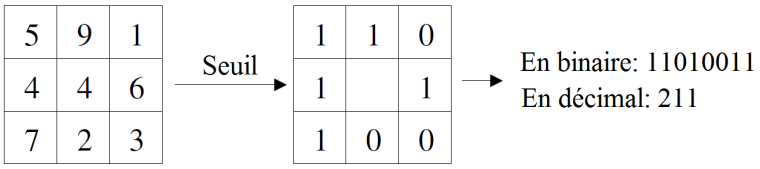

Figure 1.10 – *Etapes de construction du descripteur LBP dans sa plus simple version. On considère ici un voisinage carré, la valeur de niveau de gris du pixel central sert de seuil aux 8 pixels voisins qui l'entourent. [AHP04] .*

D'un point de vue mathématique, ceci se résume comme suit : pour un pixel donné p dont le niveau de gris est n_p, l'indice LBP qui lui est associé est :

$$LBP_{P,R}(n_p) = \sum_{v=1}^{P} S\{n_p - n_v\} 2^{v-1} \tag{1.23}$$

où n_v représente le niveaux de gris du pixels voisins v pris parmi les P voisins considérés dans un voisinage circulaire de rayon R autour du pixel p et où :

$$S\{n_p - n_v\} = \left\{ \begin{array}{ll} 1 & \text{Si } n_p \geq n_v \\ 0 & \text{Si } n_p < n_v \end{array} \right\} . \tag{1.24}$$

Après calcul de l'indice LBP pour chacun des pixels p de l'image $I(x, y)$, nous obtenons une image dite labellisée $I_l(z)$.

Par la suite, plusieurs extensions de cette méthode ont été proposées [OPM02]. Dans un premier temps, la méthode a en effet été étendue à des voisinages circulaires (et non plus carrés) de rayon différent pour pouvoir détecter des motifs plus gros. La **Figure 1.11** montre plusieurs de ces voisinages circulaires pour lesquels le nombre de voisins et la taille du cercle varient.

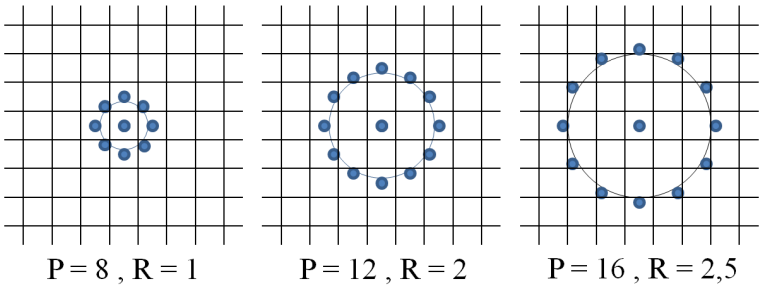

Figure 1.11 – *Représentation de 3 voisinages possibles utilisés par l'algorithme LBP. P fait référence au nombre de pixels voisins du pixel central considéré et R fait référence au rayon du cercle pris comme motif de représentation.*

Dans un second temps le descripteur a été réduit à l'utilisation des motifs dits uniformes seulement. Dans [OPM02], les auteurs ont en effet montré qu'en considérant uniquement ce type de motifs formés par 8 pixels voisins, ils prenaient en compte 90% de tous les motifs possibles en considérant un voisinage circulaire de rayon un et 70% de l'information en considérant un voisinage circulaire de rayon trois. Ces motifs uniformes (*uniforms patterns*) sont définis comme suit : si on considère un code binaire comme étant une suite circulaire de bits, toutes les suites contenant au plus deux transitions bit à bit (0-1 ou 1-0) sont appelées uniformes. Ils contiennent la majorité de l'information concernant la texture de l'image alors qu'ils ne représentent que 58 des 256 motifs possibles codés sur 8 bits. On appelle l'image obtenue après application du descripteur l'image labellisée $I_l(x, y)$. Un histogramme H de l'image $I_l(x, y)$ peut par la suite être construit comme suit :

$$H_i = \sum_{x,y} Q\{I_l(x, y) = i\} \ , \ i \in \{0, ..., n-1\} \tag{1.25}$$

où on définit n comme étant le nombre de motifs possible et $Q\{A\}$ telle que :

$$Q\{A\} = \left\{ \begin{array}{ll} 1 & \text{Si A est vraie} \\ 0 & \text{Si A est fausse} \end{array} \right\} . \tag{1.26}$$

Dans le cas où l'on considère uniquement les motifs uniformes, l'histogramme sera donc composé de 59 bins : 58 d'entre-eux correspondent aux 58 motifs uniformes et le dernier correspond aux occurrences de tous les autres.

Pour garder l'information spatiale de l'image, celle-ci est au préalable découpée en m régions R_j et un histogramme $H_{i,j}$ est calculé pour chacune de ces régions comme suit :

$$H_{i,j} = \sum_{x,y} Q\{I_l(x,y) = i\}.Q\{(x,y) \in R_j\} \ , \ i \in \{0,...,n-1\}, j \in \{0,...,m-1\} \quad (1.27)$$

C'est à partir de cet opérateur que T. Ahonen [AHP04] a introduit l'algorithme de reconnaissance de visage que nous utilisons dans ce travail. Il permet d'obtenir de très bons résultats, peu sensibles aux variations d'illumination, il est souvent plus robuste et plus efficace que ceux présentés dans la littérature [AHP04]. L'image de visage est dans un premier temps divisée en m régions $\{R_0,R_1,...,R_{m-1}\}$. L'opérateur LBP est appliqué sur l'ensemble de l'image afin d'obtenir l'image labellisée puis un histogramme des étiquettes est construit pour chacune des régions afin d'obtenir des descripteurs locaux de visage. La représentation globale du visage est obtenue par combinaison de tous les descripteurs. Au cours de ce travail, nous avons utilisé l'opérateur $LBP^{u2}_{P,R}$ et notre image a été divisée en 8*8 régions. P fait référence au nombre de points d'échantillonnage également répartis sur un cercle de rayon R et u_2 indique que seuls les motifs uniformes ont été pris en compte. On résume l'ensemble de la méthode sur la **Figure 1.12**. Cette méthode de reconnaissance de visages sera désignée dans la suite du manuscrit par l'appelation abrégée : « méthode LBP ».

Caractérisation d'un visage via le descripteur POEM (Patterns of Oriented Edge Magnitudes)

Plus récemment, Vu et al. [VC10] ont développé une nouvelle méthode de reconnaissance en proposant un nouvel extracteur de caractéristique basé sur des informations d'orientation multi-échelle au niveau des contours locaux du visage. Le descripteur POEM (Patterns of Oriented Edge Magnitudes) a permis d'obtenir de très bons résultats lorsqu'il est appliqué à la reconnaissance de visages sur la base FERET faisant concurrence aux autres descripteurs présentant les meilleurs résultats dans l'état de l'art jusque là tout en étant d'une complexité faible et permettant un calcul rapide. Nous présentons dans un premier temps le principe du descripteur puis son application à la reconnaissance de visages.

Définition du descripteur

Dans un premier temps, il s'agit de déterminer l'orientation de contours localement dans l'image. Pour caractériser l'orientation au niveau du pixel courant, les auteurs définissent une cellule dans laquelle l'orientation est représentée aussi bien au niveau du pixel

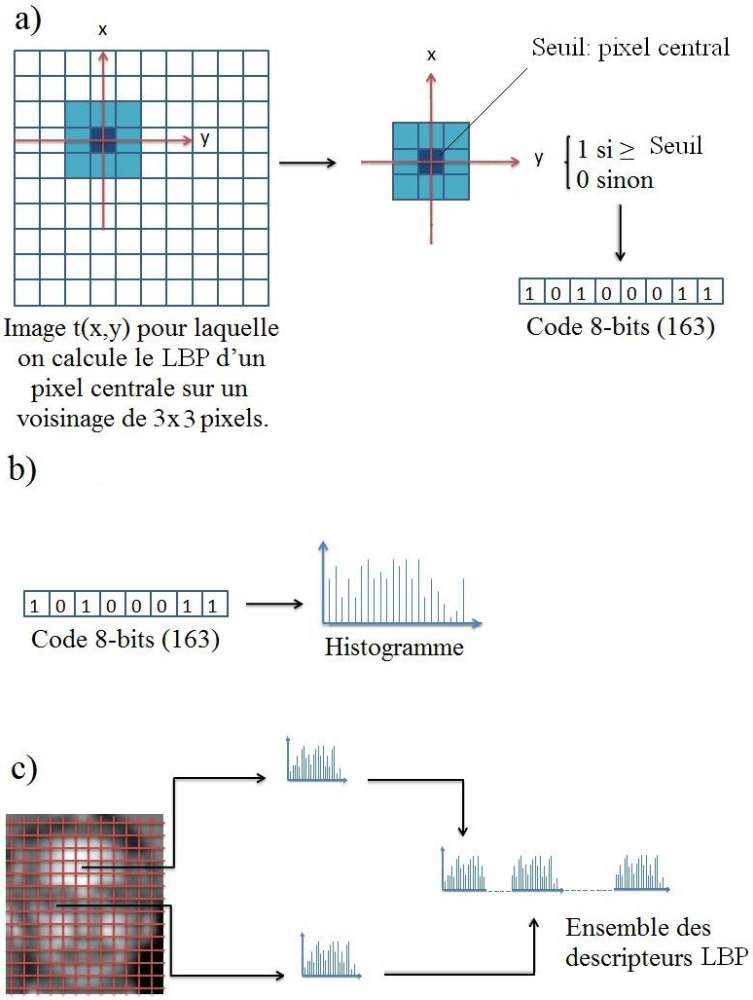

Figure 1.12 – *Etapes de construction du descripteur LBP dans sa plus simple version. On considère ici un voisinage carré, la valeur de niveau de gris du pixel central sert de seuil aux 8 pixels voisins qui l'entourent. (a) : fonctionnement du descripteur LBP sur un voisinage carré. (b) : Transformation du code 8 bits obtenu en histogramme. (c) : Obtention des histogrammes sur des régions locales de l'image labellisée. Schéma adapté de [PH11]*

que de ses voisins proches. La valeur de l'orientation du gradient est ensuite discrétisée entre $0-\pi$ ou $0-2\pi$. Généralement, le nombre d'orientations est compris entre 2 et 7 ou 4 et 14 selon le cas. Ainsi, à chaque pixel est associée la valeur discrétisée de son orientation et la valeur absolue de son gradient.

Pour caractériser l'orientation dans une petite localité de l'image, une région est tout d'abord définie autour de chaque pixel sous le terme de cellule comme on peut le voir sur la **Figure 1.13** . Puis les orientation des voisins du pixel p considéré appartenant à la cellule sont prises en compte grâce à un histogramme des orientations. Le pixel central de la cellule se voit alors attribuer le nombre d'orientations représentées dans celle-ci, chacune pondérée en fonction de la valeur absolue des gradients correspondant.

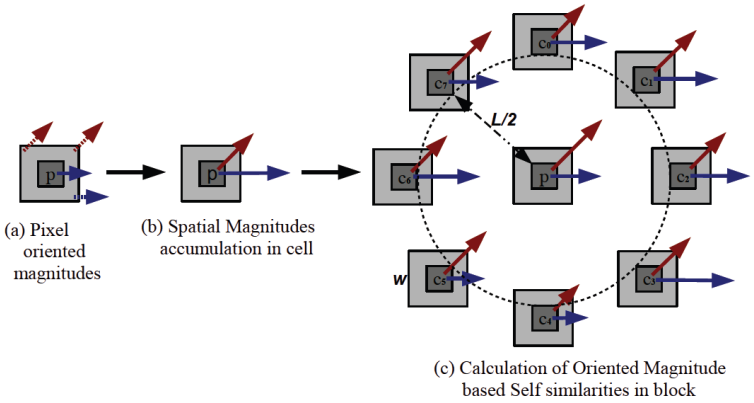

Figure 1.13 – *Etapes de construction du descripteur Poem. (a) : Calcul de l'orientation et de l'amplitude du gradient au niveau de chaque pixel d'une cellule. (b) : Accumulation de l'information au niveau du pixel central pour lequel on attribue les orientations qui sont représentées dans la cellule pondérée en fonction de l'amplitude des gradients. (c) : Mesure de l'indice LBP au niveau d'une région plus large (bloc). [VC10]*

Une fois l'étape d'accumulation terminée, il s'agit dans un second temps de prendre en compte l'information comprise dans l'image à une échelle plus grande. Pour cela, l'indice LBP est calculé pour chaque pixel à la différence près qu'au lieu de considérer le niveau de gris de chacun, ce sont les valeurs absolues des gradients, pour une orientation donnée, qui sont pris en compte. Une région plus large est donc définie pour la mesure de l'indice pour chaque pixel de l'image. Ces régions sont représentées sur la **Figure 1.13** sous le terme de bloc. Pour accélérer le calcul de la méthode, seuls les motifs uniformes sont considérés. L'indice Poem est alors obtenu comme suit :

$$Poem^{\theta_i}_{L,w,n}(p) = \sum_{j=1}^{n} Q\{S(\mathcal{G}^{\theta_i}_{p}, \mathcal{G}^{\theta_i}_{c_j})\}2^j \tag{1.28}$$

où $\mathcal{G}_p$ et $\mathcal{G}_{c_j}$ représentent respectivement l'amplitude des gradients du pixel central p et de ses voisins cj dans le bloc. $S(.,.)$ représente la fonction de similarité qui est par exemple, à

l'instar de l'extracteur LBP, la différence entre deux amplitudes de gradient. L et w sont les tailles des blocs et des cellules respectivement. n est le nombre de voisins du pixel p et enfin, Q est définie par :

$$Q\{x\} = \left\{ \begin{array}{ll} 1 & \text{Si } x \geq \tau \\ 0 & \text{Si } x < \tau \end{array} \right\} . \tag{1.29}$$

Finalement, le descripteur Poem final est obtenu après concaténation de tous les descripteurs Poem calculés pour une orientation donnée :

$$Poem_{L,w,n}(p) = \{Poem^{\theta_1}, Poem^{\theta_2}, ..., Poem^{\theta_m}\} \tag{1.30}$$

où m représente le nombre d'orientations discrètes choisi.

Application du descripteur Poem à la reconnaissance de visages

Une fois le descripteur Poem calculé pour une image donnée et pour chacune des m orientations θ_i, il s'agit de rendre l'information apportée par le descripteur plus compacte afin de pouvoir comparer un ensemble important d'images. Pour cela, les auteurs utilisent une représentation par histogramme de la même façon que pour le descripteur LBP dans [AHP04]. Le schéma présenté sur la **Figure 1.14** résume les différentes étapes de l'extraction pour une image de visages. Le gradient de l'image est calculé dans un premier temps. Puis, m orientations du gradient sont sélectionnées (4 sur la **Figure 1.14**) formant m images de gradient orienté (uni-oriented Edge Magnitude Image (EMI)). Ensuite, pour une orientation du gradient donnée, on attribue à chaque pixel de l'image la somme des gradients de tous les pixels contenus dans la cellule qui lui est associée. On obtient m images de gradient orientés accumulés (Accumulated Edge Magnitude Image (AEMI)). Enfin, on applique le descripteur LBP sur chacune de ces images, on les découpe en régions et on récupère l'histogramme de chacune de ces régions, pour chacune des images avant de les concaténer pour former la séquence d'histogramme Poem (Poem-HS). Au final, il y a donc 4 paramètres principaux à ajuster pour optimiser ce descripteur : le nombre d'orientations m, la taille de la cellule $w \times w$, la taille des blocs $L \times L$ et enfin le nombre de régions à prendre en compte pour le calcul des histogrammes.

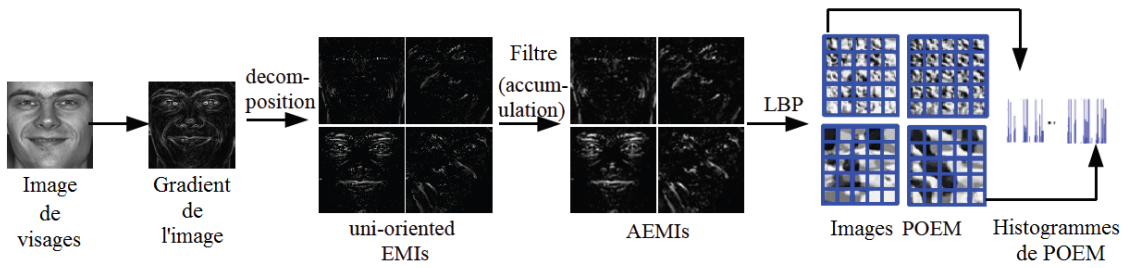

Figure 1.14 – *Etapes de construction de la méthode de reconnaissance de visages basée sur le descripteur Poem. [VC10]*

Conclusion

Nous venons de résumer les méthodes locales selon deux catégories, « Local feature-based method » et « Local appearance-based method ». Pour ces deux catégories, les méthodes de reconnaissance proposées permettent d'augmenter la robustesse de l'identification en présence de plusieurs perturbations comme l'illumination, la pose, l'expression ... Néanmoins, à l'heure actuelle, de nombreuses méthodes de reconnaissance se basent principalement sur une extraction des caractéristiques locales de l'image grâce à l'utilisation d'un ou plusieurs descripteurs de caractéristiques tels que LBP, $POEM$ ou les ondelettes de Gabor. C'est la raison pour laquelle nous avons souhaité présenter ces méthodes en détail dans ce chapitre. Il est cependant important de noter que bien qu'elles permettent d'extraire de l'image des caractéristiques locales indépendantes de certaines perturbations prises individuellement, elles ne permettent pas de traiter de manière satisfaisante l'ensemble des perturbations présentes en même temps sur une image. Une solution à ce problème pourrait venir de l'utilisation des méthodes hybrides.

2.3 Méthodes Hybrides

Comme on l'a vu en introduction, le système visuel humain traite l'information sous forme locale et globale et la reconnaissance d'un visage nécessite la combinaison de ces deux types d'informations. Malheureusement, jusqu'à présent, nous ne savons pas comment l'ensemble de ces informations est traité au niveau de notre cerveau. Néanmoins, plusieurs méthodes de reconnaissance automatique de visages cherchent à modéliser un système d'identification regroupant l'information à partir de ces deux types de caractéristiques, ce sont les méthodes dites hybrides. Le but de ces approches est donc de trouver quelles sont les caractéristiques d'un visage utiles à la reconnaissance et comment les combiner de façon à faciliter l'identification d'une personne. Ces caractéristiques présentent en effet des propriétés différentes et complémentaires : alors que les caractéristiques locales sont très sensibles au bruit, les caractéristiques globales quant à elle le sont peu. En revanche, ces dernières sont très sensibles à la pose contrairement aux caractéristiques locales. On voit ainsi clairement pourquoi ce type de méthode est si prisé. Malheureusement, il n'existe pas à ce jour d'algorithme permettant de combiner de façon optimale ces informations. Néanmoins, certaines méthodes locales peuvent être considérées comme des méthodes hybrides dans la mesure où certaines informations globales sont prises en compte. C'est par exemple le cas de la méthode présentée dans [Mar02]. Par ailleurs, les méthodes publiées dans cette catégorie [CTL94], font souvent appel à plusieurs images par personne ce qui ne rentre pas dans le cadre de notre problématique.

3. Bases de données

Dans le cadre de ce travail de thèse, les méthodes développées seront testées sur deux bases de données :

– **La base Feret** qui est une base de données de visages utilisée classiquement en reconnaissance. Bien que les conditions d'acquisition des images de cette base aient été parfaitement maitrisées, nous avons décidé de présenter des résultats sur cette base car cela nous permet de confronter nos performances à celles d'autres méthodes de l'état de l'art et d'introduire de manière contrôlée des artéfacts.

– **La base de données Biorafale** pour laquelle les images ont été acquises en contexte de vidéosurveillance (condition d'acquisition, qualité et éclairement non contrôlés). Cette base fournit des données plus proches des conditions finales d'utilisation de nos algorithmes dans la mesure où les images présentent des artéfacts multiples qui n'ont pas été introduit artificiellement.

3.1 La base de donnée FERET

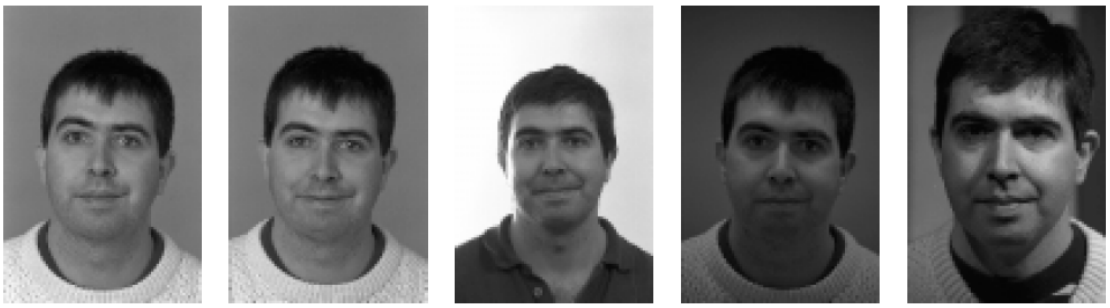

Figure 1.15 – *Images vues de face présentes dans la base FERET. Les images de la catégorie fb présentent différentes expressions par rapport à celles de la catégorie fa. Les images de la catégorie fc ont été acquises avec une caméra différente et sous différentes conditions d'illumination. Les images appartenant aux catégories duplicate I et II ont été enregistrées à différentes sessions à plusieurs jours d'intervalle.*

Dans cette section nous décrivons la base de visages FERET [PMRR97] que nous avons utilisée pour valider l'ensemble de nos expériences, que ce soit pour l'identification comme pour l'estimation de la pose.

La base FERET (The Facial Recognition Technology) est une base publique dont les images ont été collectées à l'université George Mason aux États Unis. C'est une base qui contient un nombre très important d'images de visages (plusieurs milliers) acquises sous différentes conditions. Ainsi, il existe 24 catégories différentes dans lesquelles les visages ont notamment été enregistrés sous différentes illuminations, poses ou expressions.

Nous n'allons pas décrire l'ensemble des catégories qui existent dans cette base mais seulement celles utiles à notre étude.

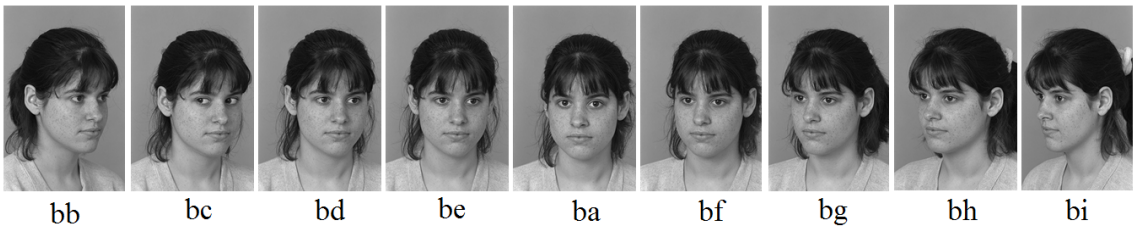

Figure 1.16 – *Les 9 poses disponibles dans la base Feret. Les bases de données bb, bc, bd, be, ba, bf, bg, bh et bi correspondent respectivement à une orientation du visage de* $+60°$, $+40°$, $+25°$, $+15°$, $0°$, $-15°$, $-25°$, $-40°$ *et enfin* $-60°$.

Toutes les images de la base FERET ont une taille de 256×384 pixels. Les images présentées sur la figure **Figure 1.15** correspondent aux cinq catégories contenant des images prises de face. Les images du jeu de données *fa* et celles du jeu *fb* ont été obtenues successivement. Il était en effet demandé aux sujets d'avoir une expression différente pour l'image de la classe *fb* par rapport à celle de la classe *fa*. Les images de la catégorie *fc* ont été acquises avec un appareil photo différent et sous des éclairages différents. Les images d'un certain nombre de sujets ont de nouveau été acquises dans une session ayant lieu à plusieurs jours d'intervalle pour former les catégories *duplicate I* et *duplicate II*. Pour *duplicate I*, entre 0 et 1031 jours séparent la seconde session de la première tandis qu'au moins 18 mois séparent la seconde de la première pour les images de la catégorie *duplicate II*. Dans le cadre de notre étude, nous avons utilisé uniquement les images des catégories *fa* et *fb*. En effet, nos expériences visent à montrer en quoi notre approche améliore le taux d'identification en présence d'image de mauvaise qualité. Il est donc important de ne pas faire intervenir plusieurs paramètres de variation qui pourraient fausser l'évaluation des performances.

Dans un second temps, nous avons travaillé sur l'estimation de la pose. C'est pourquoi, pour pouvoir valider notre méthode il nous a fallu utiliser une base d'images adaptée. Nous avons donc utilisée les images des catégories *ba* à *bi* dont nous présentons un exemple sur la **Figure 1.16**. Pour élaborer ces 9 catégories, il a été demandé au sujet de bouger la tête et le corps d'un angle allant de $-60°$ à $+60°$. Chacune des catégories contient 200 images appartenant à 200 sujets différents.

3.2 La base de donnée BIORAFALE

La base de données Biorafale est destinée, à terme, à simuler plusieurs scénarios pouvant être rencontrés à l'entrée d'un stade de foot. Les performances d'un algorithme de reconnaissance dépendent en effet de nombreux paramètres comme par exemple :

– L'éclairage : les matchs se déroulent généralement la nuit, il est donc important de tester les méthodes de reconnaissance lorsque les visages sont faiblement éclairés ou de façon peu uniforme.

– L'occultation : les personnes peuvent porter des bonnets, des écharpes ou des lunettes qui peuvent masquer partiellement le visage rendant l'identification plus difficile.
– La pose : les visages ne sont pas forcément face à la caméra.

L'acquisition de ces vidéos réclament un dispositif très lourd. Ce dispositif fait appel à plusieurs centaines de personnes pour simuler une foule mais également les personnes que l'on souhaite reconnaître (les interdits de stade). L'organisation de ces campagnes d'acquisition est donc à la fois complexe et très coûteuse. Par ailleurs, la mise en place de ces campagnes doit recevoir l'accord de la Commission Nationale de l'Informatique et des Libertés (CNIL) qui peut prendre plusieurs mois d'attente. C'est la raison pour laquelle nous n'avons pas pu obtenir jusqu'à maintenant des images prises dans un réel contexte de vidéosurveillance. La base de données qui a été mise à notre disposition et que nous avons utilisée dans cette thèse a été élaborée au stade Gabriel Montpied de Clermont Ferrand à l'aide d'une caméra fixe et de figurants issus des différents laboratoires travaillant pour le projet. Une nouvelle campagne d'acquisition plus complète et plus adaptée à notre contexte est prévue pour 2012.

Description de la base Biorafale utilisée :

Le scénario mis en place pour cette base de données est simple. Il s'agit d'enregistrer plusieurs personnes passant, à des temps différents, dans le champ de la caméra. Celle-ci est située dans un plan intermédiaire en hauteur par rapport aux sujets (comme c'est souvent le cas dans un contexte de vidéosurveillance).
Plusieurs vidéos (d'une minute environ) sont enregistrées, une par personnes à reconnaître et par session. Nous disposons des vidéos de deux de ces sessions, une dizaine de personnes pour chacune d'entre elles. Les visages ont été détectés au préalable avec un dispositif adapté. Nous disposons donc, pour chaque personne, d'un ensemble d'images de visages issues de la vidéo dont la qualité et la définition est très variable. Cela dépend en effet de la distance à laquelle le sujet a été filmé au cours de son passage devant la caméra, de ces mouvements (rapides, lents) etc. La **Figure 1.17** illustrent plusieurs images obtenues pour une des participantes.

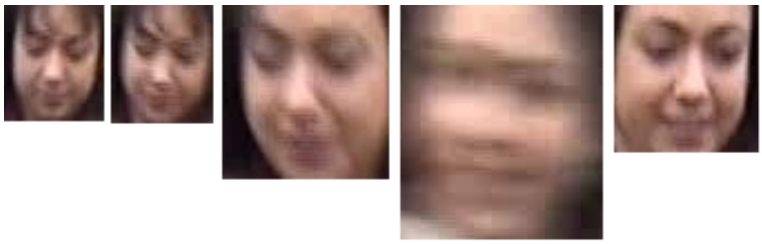

Figure 1.17 – *Illustration de la base Biorafale. Images acquises avec une caméra vidéo dans un scénario pour lequel la personne doit passer dans le champ de vision de la caméra pendant un intervalle d'une minute environ. Les numéros des images correspondantes de gauche à droite sont : 344, 347, 402, 823, 803.*

Comme on peut le voir sur la **Figure 1.17**, les images acquises présentent un niveau de flou très important et la taille des images varie énormément d'une image à l'autre. La définition de ces images est donnée dans le **tableau 1.1**. On remarque également une grande variabilité de la pose des images acquises.

Table 1.1 – *Définition des images présentées sur la* ***Figure 1.17****.*

Variation de la définition des images		
Base	Numéros	Taille
Object6	344	42×52
	347	46×53
	402	75×78
	803	65×67
	823	91×107

Ces images présentant des artéfacts multiples, elles ont été utilisées dans le cadre de cette thèse pour mesurer le niveau des artéfacts présents et non pour tester les performances des algorithmes.

3.3 Précision de vocabulaire vis-à-vis des bases de données

Pour la reconnaissance de visages, au moins deux bases d'images sont nécessaires :

- La **galerie** est la base contenant les images des visages des personnes à reconnaître. Les images de cette base sont généralement prises dans de bonnes conditions d'acquisition contrôlées.
- La **base de test** regroupe les images qui permettent de tester les performances de l'algorithme de reconnaissance. Ce sont les images qui, dans notre contexte, présentent divers artéfacts d'acquisition (flou, effets de bloc) ou des variations de la pose du visage.

  Enfin, certaines méthodes de reconnaissance nécessitent d'être optimisées au préalable. Cela signifie que plusieurs paramètres doivent être appris avant l'utilisation de l'algorithme. Ceci est fait lors d'une phase dite d'apprentissage :
- La **base d'apprentissage** contient donc l'ensemble des images nécessaires à l'apprentissage des paramètres et à l'optimisation de l'algorithme. Aucune image de la base d'apprentissage ne doit figurer dans la base de test. Pour tester un algorithme, il convient en effet de tester un algorithme sur des images qui n'ont jamais été présentées au système.

4. Analyse des performances des systèmes de reconnaissance de visages sur des images de vidéosurveillance

Nous souhaitons étudier l'évolution des performances de plusieurs algorithmes d'identification de visages lorsqu'ils sont utilisés dans un contexte de vidéosurveillance. Nous présentons dans un premier temps les taux d'identification obtenus pour plusieurs algorithmes de l'état de l'art lorsqu'ils sont appliqués sur des images prises dans des conditions contrôlées ou présentant des variations d'illumination. Puis nous présentons dans un second temps l'évolution des performances de ces algorithmes en présence d'artéfacts couramment rencontrés dans un contexte de vidéosurveillance à savoir en présence de flou, d'effets de bloc et de variations de pose.

4.1 Performances des systèmes de reconnaissance en conditions contrôlées

Pour cela, nous présentons dans le **tableau 1.2** un récapitulatif des méthodes d'identification de visages les plus performantes de l'état de l'art sur la base FERET [PMRR97]. Ces méthodes de reconnaissance utilisent une combinaison de plusieurs extracteurs de caractéristiques. Les principaux descripteurs de visage considérés sont LBP [OPH96] et les ondelettes de Gabor. Ainsi, dans le **tableau 1.2**, LBP fait référence à la méthode de reconnaissance développée par Ahonen et al. [AHP04] qui se base sur le descripteur LBP. LGBPHS est l'algorithme de reconnaissance développé par Zhang et al [ZSG$^+$05] qui se base sur le descripteur LGBP de Ma et al [MZS$^+$06], lui même basé sur une combinaison du descripteur LBP avec un ensemble d'ondelettes de Gabor. HGPP fait quant à lui référence à la méthode d'identification développée par Zhang et al [ZSCG07] et qui se base sur l'information de phase issue d'un ensemble d'ondelettes de Gabor. Enfin, $POEM$ fait référence à la méthode proposée par Vu et al. dans [VC10] qui se base sur un calcul de gradient combiné au descripteur LBP.

Table 1.2 – *Comparaison des performances obtenues pour plusieurs algorithmes d'identification sur plusieurs jeux de données de la base FERET.*

Comparaison des taux d'identification				
Méthodes	Fb (%)	Fc (%)	Dup1 (%)	Dup2 (%)
LBP [AHP04]	93	51	61	50
LGBPHS [ZSG$^+$05]	94	97	68	53
HGPP [ZSCG07]	97.6	98.9	77.7	76.1
POEM [VC10]	97.6	96	77.8	76.5

Comme on peut le voir sur le **tableau 1.2**, l'ensemble des méthodes d'identification qui sont mentionnées obtiennent de très bons résultats sur le jeu de données Fb de la base FERET. Exceptée la méthode d'identification basée sur le descripteur LBP, les méthodes obtiennent également de bons résultats sur le jeu de données Fc qui présente des variations d'illumination. En revanche, les performances chutent pour toutes les méthodes lorsqu'elles sont appliquées sur les jeux de données Dup1 et Dup2 où les images ont été acquises dans des conditions différentes des images de la galerie (le jeu de données Fa). Les performances des algorithmes d'identification vis-à-vis des variations d'illumination ont donc fait l'objet de nombreuses études dans l'état de l'art. L'analyse de ce tableau montre que le problème

de la reconnaissance de visages est très bien résolu sur des images de visage de face acquises en conditions contrôlées. Les performances sont déjà un peu moins bonnes lorsque plusieurs mois séparent la session d'acquisition des images de la galerie de celle des images test comme c'est la cas des classes Dup1 et Dup2.
L'utilisation de la vidéosurveillance pour l'acquisition d'images fait intervenir d'autres artéfacts dont les conséquences sur les performances des algorithmes ont été très peu étudiées jusqu'à présent. C'est pourquoi nous proposons d'évaluer les performances de ces algorithmes sur des images présentant des artéfacts de flou ou de blocs.

4.2 Performances des systèmes de reconnaissance dans un contexte de vidéosurveillance

4.2.1 Influences des artéfacts de flou sur la reconnaissance de visages

Nous présentons sur la **Figure 1.18** les performances des deux méthodes de reconnaissance basées sur le descripteur LBP pour l'une et les ondelettes de Gabor pour l'autre. En effet, comme nous l'avons vu précédemment, plusieurs méthodes d'identification se basent sur ces deux descripteurs. Il est donc intéressant d'étudier leur robustesse en présence de différentes perturbations. Nous avons utilisé pour nos tests les images de la base FERET. 1194 images du jeu de données Fa forment la galerie. 1194 images du jeu de données Fb forment la base de test. Pour simuler un flou de mise au point dans une image, nous avons dégradé artificiellement la base de test en appliquant sur chacune des images un filtre gaussien d'écart type croissant.

La méthode de reconnaissance basée sur le descripteur LBP est présentée dans [AHP04]. Nous avons calculé un vecteur de caractéristiques basé sur le descripteur LBP_{u2} (LBP uniforme) pour chaque image de la galerie et pour chaque image de la base de test. Pour procéder à la reconnaissance nous avons mesuré les similarités entre le vecteur de caractéristiques associé à l'image test et celui associé à chaque image de la galerie à l'aide d'un calcul de distance $\chi 2$.

Pour la méthode d'identification basée sur les ondelettes de Gabor, nous avons calculé pour une image donnée 40 convolutions de cette image avec 40 noyaux de Gabor (8 orientations et 5 échelles différentes) généralement utilisés comme c'est le cas dans [ZSG+05]. Avant de concaténer l'ensemble de ces images pour former le vecteur de caractéristiques de l'image, nous avons sous-échantillonné chacune d'elle pour diminuer la taille du vecteur final. Pour procéder à la reconnaissance, le vecteur de caractéristiques associé à l'image test est comparé à chacun des vecteurs de la galerie à l'aide d'une distance de similarité cosinus, souvent utilisée dans ce cas.

Comme on peut le voir sur la **Figure 1.18**, les performances de ces deux méthodes d'identification sont fortement dégradées lorsque l'intensité du flou augmente. Ce n'est pas étonnant dans la mesure où ces deux descripteurs sont basés sur une information spatiale (intensité des pixels) de l'image. Par ailleurs, on peut également remarquer que la chute du taux d'identification n'est absolument pas régulière mais au contraire très brutale. Il

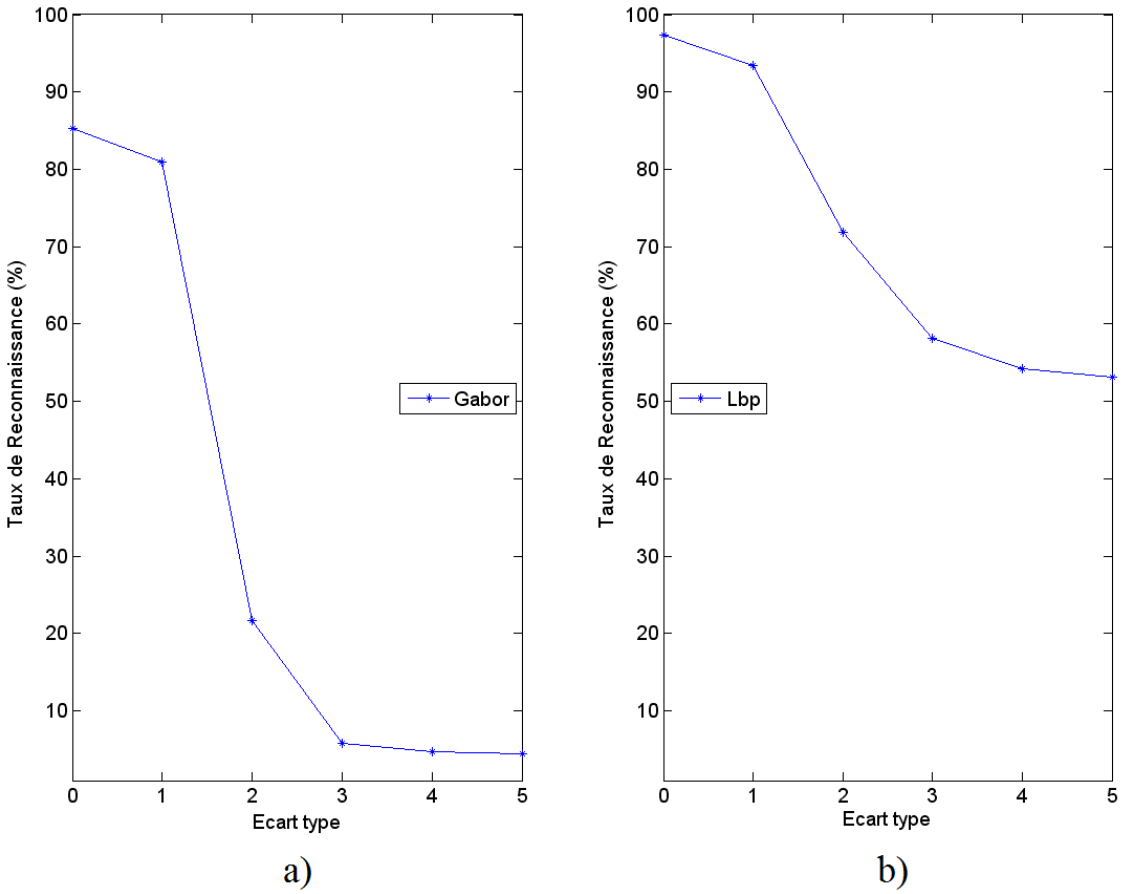

Figure 1.18 – *Influence de l'artéfact de flou sur la reconnaissance. Performances des méthodes d'identification basée sur les ondelettes de Gabor (a) et sur le descripteur LBP [AHP04] (b).*

y a donc clairement un seuil limite à partir duquel les performances de ces algorithmes ne sont plus acceptables. Nous pouvons notamment remarquer que pour des niveaux de flou élevés les taux avoisinent les 50% d'identification pour la méthode LBP voire même autour de 20% pour la méthode de reconnaissance basée sur les ondelettes de Gabor. Il est donc important de proposer une solution pour la reconnaissance des visages acquis sur de telles images.

Dans un second temps, nous avons testé les performances du descripteur LPQ proposé par Ojansivu et al. dans [OH08] et dont nous détaillons le principe au chapitre 3. Ce descripteur se veut en effet robuste au flou. Néanmoins, les résultats obtenus avec la méthode d'identification proposée par Ahonen dans [AROH08] le sont pour de faibles quantités de flou. Nous avons donc souhaité connaître les performances de cet algorithme de reconnaissance pour de plus fortes intensités de flou. Les résultats obtenus sont reportés sur la **Figure 1.19**.

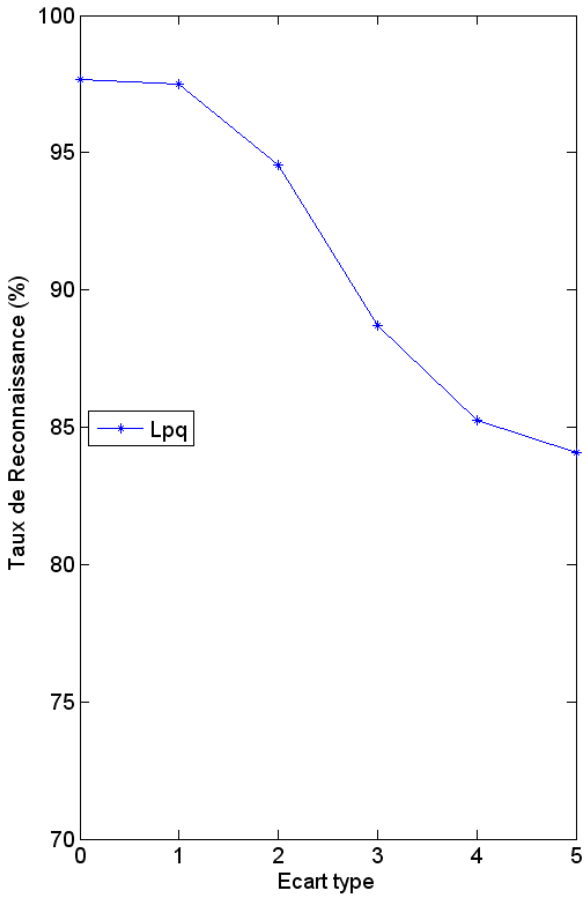

Figure 1.19 – *Influence de l'artéfact de flou de mise au point sur la reconnaissance. Performances de la méthode d'identification basée sur le descripteur LPQ [OH08] présentée dans [AROH08].*

Comme on peut le voir sur cette figure, les performances de cet algorithme sont meilleures que celles présentées sur la **Figure 1.18** mais on observe malgré tout une nette diminution des performances pour de forts artéfacts de flou. Nous avons également reporté sur la **Figure 1.20**, les résultats obtenus avec les deux méthodes de reconnaissance basées sur les descripteurs LBP et LPQ respectivement lorsque le flou présent dans l'image est un flou de bougé dans la direction horizontale (0°). L'abscisse correspond au nombre croissant de pixels qui ont été décalés selon la direction 0°. Plus ce nombre est important, plus le niveau de flou est élevé.

Tout comme pour le flou de mise au point, les performances des deux algorithmes diminuent lorsque le niveau de flou augmente. Les deux méthodes semblent néanmoins moins sensibles à ce type de flou et le descripteur LPQ est de nouveau plus robuste que ne l'est le descripteur LBP.

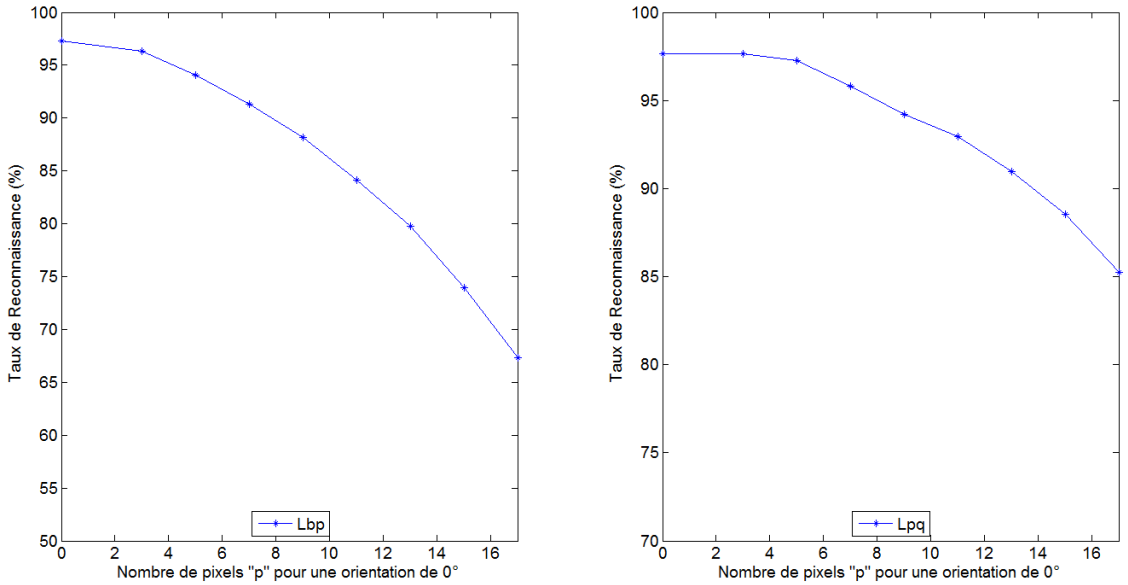

Figure 1.20 – *Influence de l'artéfact de flou de bougé sur la reconnaissance. Performances de la méthode d'identification basée sur le descripteur LBP à gauche et LPQ à droite.*

4.2.2 Influence des artéfacts de blocs sur la reconnaissance de visages

De même que précédemment, nous avons utilisé pour nos tests les images de la base FERET. 200 images du jeu de données Fa forment la galerie. 200 images du jeu de données Fb forment la base de test. Pour simuler l'effet de blocs dans une image, nous avons dégradé artificiellement la base de test en la compressant avec un facteur de qualité de compression $JPEG$ variable.

Les résultats de l'identification sur des images présentant des effets de blocs sont présentés sur la **Figure 1.21**. Comme on peut le voir, les performances des deux algorithmes basés sur les descripteurs LBP et LPQ respectivement sont fortement diminuées pour des taux de compression élevés. On note cependant une meilleure résistance de l'algorithme basé sur le descripteur LPQ comparé à celui basé sur le descripteur LBP.

4.2.3 Influences des artéfacts de pose sur la reconnaissance de visages

Nous souhaitons montrer la robustesse des algorithmes de reconnaissance de visages lorsque la pose varie. Pour cela, nous présentons sur la **Figure 1.22**, les résultats obtenus pour 3 méthodes de reconnaissance basées sur les descripteurs de visages POEM, LBP et Gabor respectivement. Ces résultats sont tirés de la thèse de S. Vu dans [Vu10]. Comme nous l'avons expliqué précédemment, l'identification d'une personne s'est faite par comparaison des vecteurs de caractéristiques à l'aide d'une mesure de distance $\chi 2$ pour les méthodes basées sur les descripteurs LBP et POEM et avec une mesure de distance cosinus pour la méthode de reconnaissance utilisant les ondelettes de Gabor.

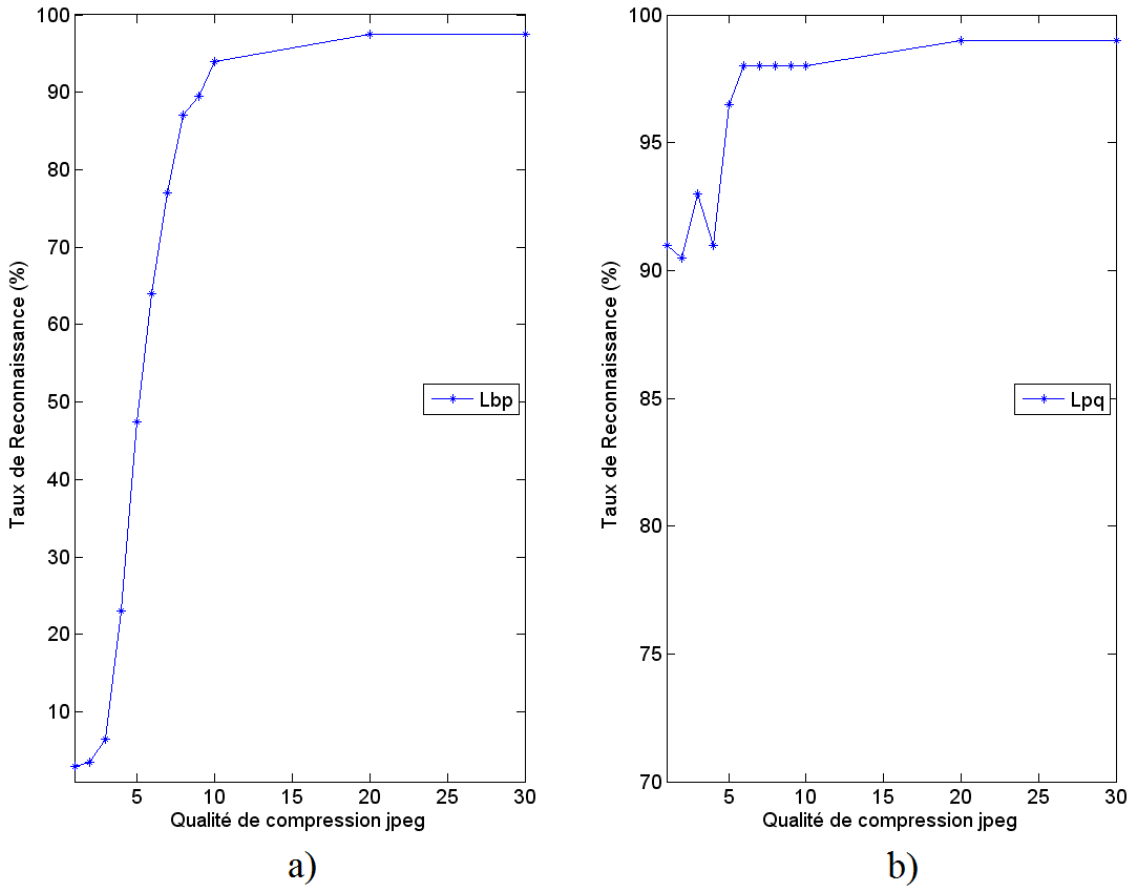

Figure 1.21 – *Influence de l'artéfact de blocs sur la reconnaissance. Performances des méthodes d'identification basée sur le descripteur LBP [AHP04] (a) et sur le descripteur LPQ [AROH08] (b).*

Comme nous pouvons le voir sur cette figure, les méthodes sont robustes à une variation de pose lorsque celle-ci varie peu. Dés que nous dépassons une valeur de $\pm 15°$ pour les descripteurs LBP et Gabor, les performances des deux algorithmes chutent considérablement. POEM est plus robuste car les performances de la méthode de reconnaissance chutent à partir d'un angle de $\pm 40°$.

Néanmoins, dans tous les cas, l'influence de la pose sur les performances des algorithmes de reconnaissance est très marquée dés lors que celle-ci présente de larges variations.

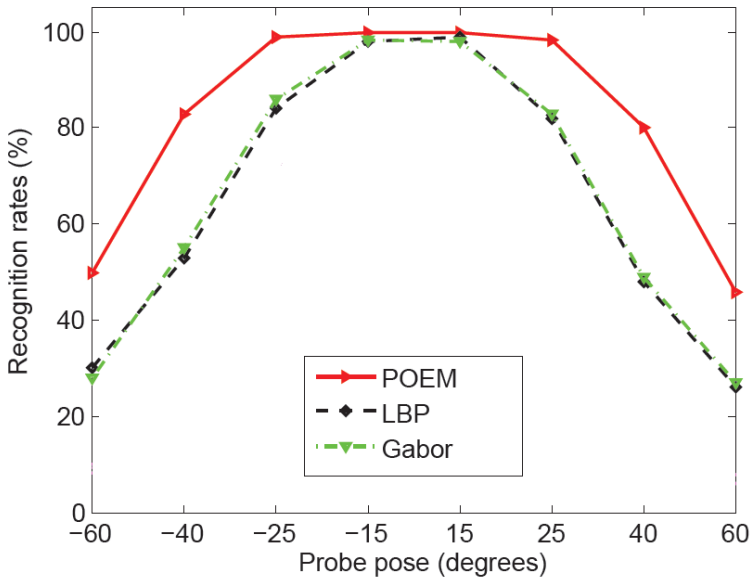

Figure 1.22 – *Influence de l'artéfact de pose sur la reconnaissance. Performances des méthodes d'identification basée sur les descripteurs LBP, Gabor et POEM.[?]*

Conclusion

Dans ce chapitre, nous avons présenté dans un premier temps l'ensemble des méthodes de reconnaissance principalement utilisées ou ayant permis une avancée significative dans le domaine. Récemment, de nouvelles méthodes d'identification basées sur des descripteurs de visages ont été proposées. Nous avons présenté ceux qui dans l'état de l'art semblaient les plus performants. Puis, nous avons étudié les performances de certains de ces descripteurs de visages vis-à-vis de plusieurs artéfacts (flou, blocs et pose) régulièrement rencontrés sur des images acquises avec une caméra de vidéosurveillance. Nous avons ainsi montré que les taux d'identification étaient fortement diminués en présence de ces artéfacts. Il est donc intéressant de connaître la qualité d'une image avant de procéder à l'identification. En effet, dans un contexte de vidéosurveillance, nous disposons de vidéos et donc de plusieurs images d'un même sujet. Ces images peuvent être de qualité variables et il est intéressant de pouvoir sélectionner celles qui présentent le moins d'artéfacts. Pour cela, l'utilisation de métriques de qualité est nécessaire. Dans le chapitre suivant, nous présentons les artéfacts régulièrement rencontrés dans un contexte de vidéosurveillance et nous décrivons les métriques de flou et de blocs qui existent dans l'état de l'art.

Chapitre 2

La qualité dans les images

Introduction

La vidéosurveillance a commencé à se développer dans les années 70, d'abord au Royaume-Uni puis en France, et a connu un essor dans les années 90. Parallèlement, ces dernières années, les outils numériques se sont développés de façon extrêmement rapide et ont pris progressivement le pas sur l'analogique. Avec l'arrivée de cette nouvelle technologie de nouveaux problèmes sont apparus auxquels il a fallu faire face. Le nombre de caméras de vidéosurveillance croissant, la quantité d'information recueillie devient de plus en plus importante et se pose le problème de la transmission et du stockage. Des standards de compression comme entre autres MPEG2, font leur apparition et sont intégrés dans le processus d'utilisation des caméras. Cela permet alors de diminuer considérablement la taille des fichiers à transmettre mais diminue dans le même temps la qualité des images à traiter. Les images acquises sont alors de qualité variable et certains artéfacts tels que les effets de bloc apparaissent en plus des problèmes d'acquisition déjà existants comme le flou de mise au point ou de bougé. Nous présentons sur la **Figure 2.1** la chaine d'acquisition d'un signal. Celle-ci se décompose en une phase d'acquisition puis une phase de compression avant envoi du signal par le canal de transmission et décodage pour permettre son affichage sur un moniteur.

Ce n'est qu'à partir de 1995 que l'utilisation de la vidéosurveillance en France est encadrée par une loi (loi n°95-73 du 21 janvier 1995 relative à la sécurité) et ce n'est qu'à partir de 2007 (Arrêté du 3 août 2007 portant sur la définition des normes techniques des systèmes de vidéosurveillance) que certaines normes techniques relatives à l'utilisation des

Figure 2.1 – *Chaine d'acquisition d'un signal.*

caméras de vidéosurveillance ont été définies. Malheureusement, ces normes ne sont pas forcément adaptées à la problématique de reconnaissance de visages. C'est d'ailleurs l'objet du projet de recherche QuIAVU débuté entre 2008 et 2009 qui vise à définir les critères de qualité minimum nécessaires au traitement des images au moment de la réception. Ces critères devraient en effet permettre que l'analyse puisse se faire dans les meilleures conditions possibles et qu'elle puisse être ensuite validée de façon objective dans un cadre juridique. A l'heure actuelle ces critères n'existent pas et de nombreuses images issues de caméras de vidéosurveillance présentent de multiples artéfacts.

Dans ce chapitre, nous commençons par décrire les principaux artéfacts qui peuvent apparaître sur une vidéo ayant subi une phase de compression en nous focalisant sur les deux principaux à savoir le flou et l'effet de bloc. Puis nous faisons un état de l'art sur les métriques d'estimation de la qualité des images (estimation du flou et de l'effet de blocs principalement). En effet, nous avons mis en évidence dans le chapitre 1 l'effondrement des performances des méthodes de reconnaissance de visages en cas d'images dégradées. Il est donc nécessaire de pouvoir avoir une idée de la qualité d'une image de visage avant de procéder à son identification.

1. Les principaux artéfacts présents dans une image ou une vidéo

Étant donné que nous avons basé notre étude sur les images fixes (ce qui revient, dans une vidéo, à baser toute l'étude sur des images de type I) nous ne nous intéresserons dans ce chapitre qu'aux artéfacts introduits par la compression spatiale. Pour plus de précisions sur le principe de la compression spatiale et temporelle, se référer à **l'annexe C**. Nous présentons 6 types de distorsions spatiales en commençant par le flou et l'effet de blocs qui sont les plus importants car ce sont généralement les deux plus visibles et donc les plus gênants pour la perception d'une image :

1. Le flou
2. L'effet de blocs
3. L'effet de ringing
4. L'effet de motif

5. Les faux contours
6. L'effet d'escalier

1.1 Les artéfacts liés aux effets de blocs

Durant la compression, l'image est découpée en blocs de 8*8 pixels sur lesquels on applique une matrice de quantification. Ce traitement étant réalisé sur un bloc et de façon indépendante des autres, une discontinuité peut apparaître au niveau des frontières. L'effet de blocs est donc fortement lié au taux de compression de l'image mais il dépend également de la variation des informations existant entre les blocs. Par conséquent plus on privilégie la composante continue au détriment des hautes fréquences moins les variations d'informations seront importantes et plus l'effet de blocs sera visible. On illustre l'effet de blocs sur la **Figure 2.2**.

Figure 2.2 – *Illustration de l'**effet de blocs** particulièrement visible au niveau des zones homogènes de l'image (ciel).*

1.2 Les artéfacts liés au flou

La particularité de cet artéfact est qu'il peut être causé par plusieurs facteurs :

1. *La compression* : l'étape de quantification privilégie les coefficients basses fréquences aux coefficients de hautes fréquences. Cela a pour principal conséquence la diminution de la netteté des contours ce qui rend l'image floue. Ainsi, plus la quantification est importante, plus l'effet de flou sera visible. Nous illustrons cette classe de flou sur la **Figure 2.3**.
2. *L'interpolation* : sur une image, il arrive que l'on ait besoin d'effectuer un zoom d'une partie de l'image. Cela entraîne donc l'utilisation d'un algorithme d'interpolation qui estime la valeur d'un pixel inconnu en fonction des pixels voisins par pondération de leur intensité. Cela entraine donc, pour la plus grande partie des algorithmes d'agrandissement, un effet de moyenne sur l'image qui est alors perçue comme floue.
3. *L'acquisition de l'image* : en effet, il arrive que la mise au point de l'appareil d'acquisition ne soit pas bien réglée par rapport à la scène que l'on souhaite observer.

Figure 2.3 – *Illustration de l'**effet de flou causé par une forte compression de l'image**. Comme on peut le voir sur l'image du haut, le premier plan de l'image est net ce qui n'est plus le cas sur l'image compressée du bas.*

Ce flou est généralement modélisé par une fonction gaussienne donnée par :

$$G(x,\sigma) = \frac{1}{\sqrt{(2\pi\sigma)}} exp^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{2.1}$$

où σ représente l'écart-type de la fonction.

Cet effet de flou est illustré sur la **Figure 2.4**.

4. Le flou peut également apparaître lorsque le sujet filmé bouge par rapport à la caméra (ou lorsque la caméra bouge par rapport au sujet), c'est ce que l'on appelle le flou de bougé. Nous illustrons cet effet de flou de bougé sur la **Figure 2.5**.

1.3 Les artéfacts liés aux effets de ringing

Il est équivalent au phénomène de Gibbs en traitement du signal. La représentation d'un signal carré à partir d'une somme finie de sinusoïdes introduit des oscillations sur les bords du créneau. On peut faire la même observation pour l'effet de ringing que l'on voit apparaître principalement au niveau des contours ou des ombres de l'image. En effet,

Figure 2.4 – *Illustration du **flou de mise au point** : la mise au point de l'appareil est faite sur les gouttes d'eau présentes sur la vitre alors que l'information intéressante se trouve à l'arrière plan que l'on voit flou.*

un bloc de l'image est représenté par une somme de motifs de la DCT dont la pondération varie en fonction de la quantification. Dans le cas du contour d'un objet plusieurs motifs sont utilisés et par conséquent, selon la quantification réalisée, certaines composantes peuvent être atténuées au détriment d'autres. Cela explique pourquoi cet effet est principalement observé au niveau des contours de l'image proche de zones homogènes (composante fréquentielle nulle). Nous illustrons l'effet de ringing sur la **Figure 2.6**.

1.4 L'effet de motif

Cet artéfact fait apparaître, comme son nom l'indique, des motifs sur des zones de l'image après compression. Ils correspondent aux motifs élémentaires associés à la DCT. Un bloc est représenté par une combinaison de ces motifs. Un coefficient est donc attribué à chacun de ces motifs en fonction de l'information contenue dans le bloc. Or, lorsque la compression est élevée, la quantification est telle qu'il ne reste que très peu de coefficients

Figure 2.5 – *Illustration du **flou de bougé** : le sujet est en mouvement. Le temps d'obturation de l'appareil est trop long et on perçoit un effet de flou.*

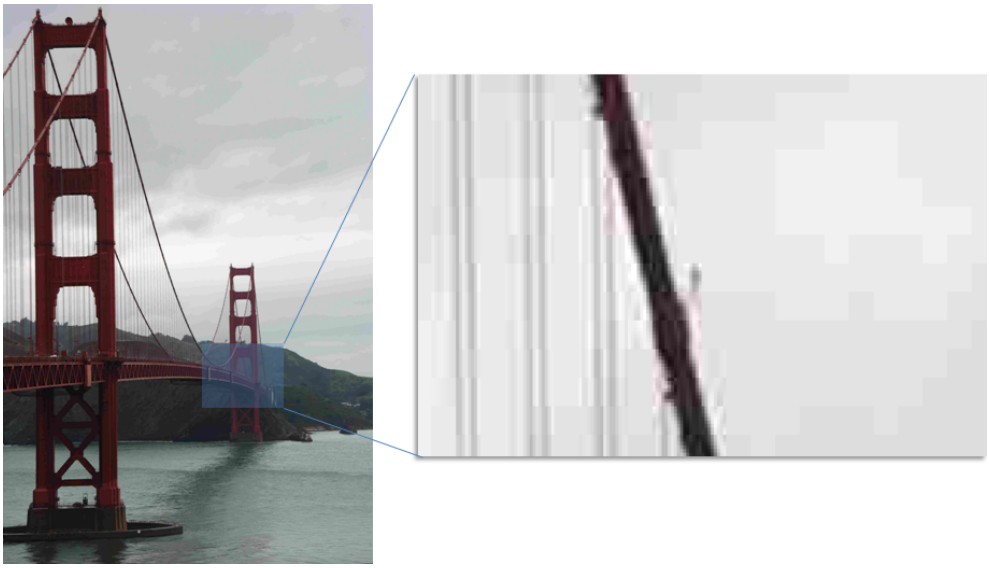

Figure 2.6 – *Illustration de l'**effet de ringing**. On perçoit bien les oscillations le long des contours de l'image.*

représentés voire un seul. Dans ce cas, le bloc représente alors le motif auquel correspond le coefficient restant. **Figure 2.7**.

Figure 2.7 – *Illustration de l'**effet de motif**. On perçoit clairement sur le bloc jaune les motifs correspondant à ceux de la DCT.*

1.5 Les faux contours

Celui-ci se voit au niveau des zones faiblement texturées de l'image. Ces zones contiennent peu d'information et l'étape de quantification tend à les rendre homogènes en fonction du niveau de compression appliqué. La quantification se faisant sur un bloc, indépendamment de l'information contenue dans les blocs voisins, les valeurs attribuées peuvent varier d'un bloc à l'autre faisant apparaître de faux contours. Cet effet est illustré sur la **Figure 2.8**.

1.6 L'effet d'escalier

Cet effet s'observe au niveau des contours rectilignes qui ne sont ni horizontaux, ni verticaux. Or chaque contour d'une image est représenté par une combinaison de motifs élémentaires associés à la DCT. Plus la combinaison est forte, moins il y a de motifs disponibles. La quantification éliminant préférentiellement les motifs hautes fréquences il ne reste généralement que des motifs basses fréquences. Autrement dit, pour représenter un tel contour après compression, seuls des motifs basses fréquences horizontaux et verticaux sont disponibles ce qui entraîne l'effet de marche d'escalier observé lors de fortes compression. Ceci est illustré sur la **Figure 2.9**.

Nous venons de présenter les principales distorsions spatiales que nous pouvons rencontrer dans une image après compression. Ces artéfacts contribuent tous à dégrader de façon plus ou moins importante la qualité de l'image. Néanmoins, certains sont plus gênants que d'autres pour la reconnaissance d'un visage. En particulier, les artéfacts de blocs et le flou dégradent l'ensemble de l'image contrairement aux autres effets qui sont plus localisés.

Figure 2.8 – *Illustration de l'**effet de faux contours** : l'image du bas correspond à l'image du haut compressée. Nous pouvons voir sur l'image du bas des zones homogènes qui apparaissent sur l'image au niveau du ciel et que nous ne voyons pas sur l'image du haut. L'apparition de ces contours est gênante car particulièrement visible sur l'image.*

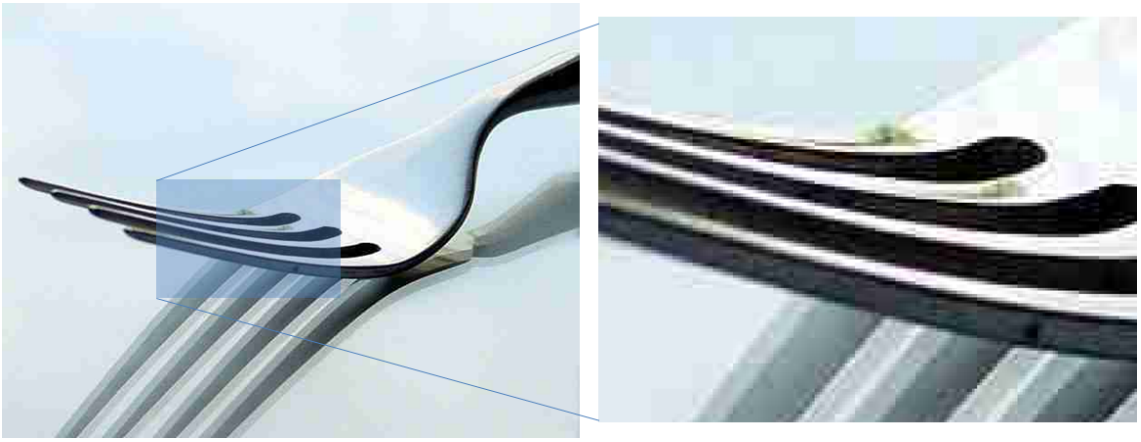

Figure 2.9 – *Illustration de l'**effet d'escalier** : celui-ci apparaît clairement au niveau des dents de la fourchette.*

2. Estimation de la qualité des images : état de l'art

Les algorithmes de reconnaissance de visages étant très sensibles aux dégradations de type flou ou blocs, il est important de pouvoir quantifier l'importance de ces dégradations avant tout traitement. Plusieurs métrique ont été développées afin d'estimer la qualité d'une image en fonction des artéfacts présents. Parmi l'ensemble de ces métriques nous nous focalisons sur celles de flou et de bloc dont nous présentons ici un bref état de l'art.

Il existe plusieurs types de métriques que l'on peut regrouper sous deux catégories : les métriques avec référence et les métriques sans référence. Les métriques avec référence estiment la qualité d'une image (ou d'une vidéo) en comparant l'image ou la vidéo dégradée avec l'originale de bonne qualité. Or pour de nombreuses applications, il est souvent très difficile voire impossible de disposer de l'image originale. C'est pourquoi, pour ce type d'application il est indispensable de faire appel à des métriques sans référence. Ce terme signifie que le résultat de la métrique n'est pas relatif à l'image originale mais correspond à une mesure objective associée à une image donnée. Ce type de métrique permet de comparer la qualité de plusieurs images entres-elles indépendamment des images originales. Dans le cadre de cette thèse, nous ne disposons que des images issues de caméras vidéo surveillance dont la qualité a été dégradée suite à une compression souvent excessive et une qualité d'acquisition entrainant de multiples artéfacts de flou. C'est pourquoi nous ne nous intéresserons par la suite qu'aux métriques sans référence.

2.1 Les Métriques de flou

Nous présentons dans cette section les principales méthodes d'estimation du niveau de flou. Puis nous décrivons en détail le principe de la méthode d'estimation choisie, le BluM [CRDLN07] ainsi que les raisons qui ont motivées notre choix.

2.2 État de l'art des métriques de flou

La dégradation d'une image par des artéfacts de flou peut être détectée aussi bien dans le domaine spatial que dans le domaine fréquentiel. Dans le domaine spatial, le flou apparaît principalement au niveau des contours de l'image. En effet, le flou a tendance à atténuer la perception des contours en les lissant. Dans le domaine fréquentiel, une image apparaît floue lorsque les hautes fréquences qui composent son spectre sont atténuées.

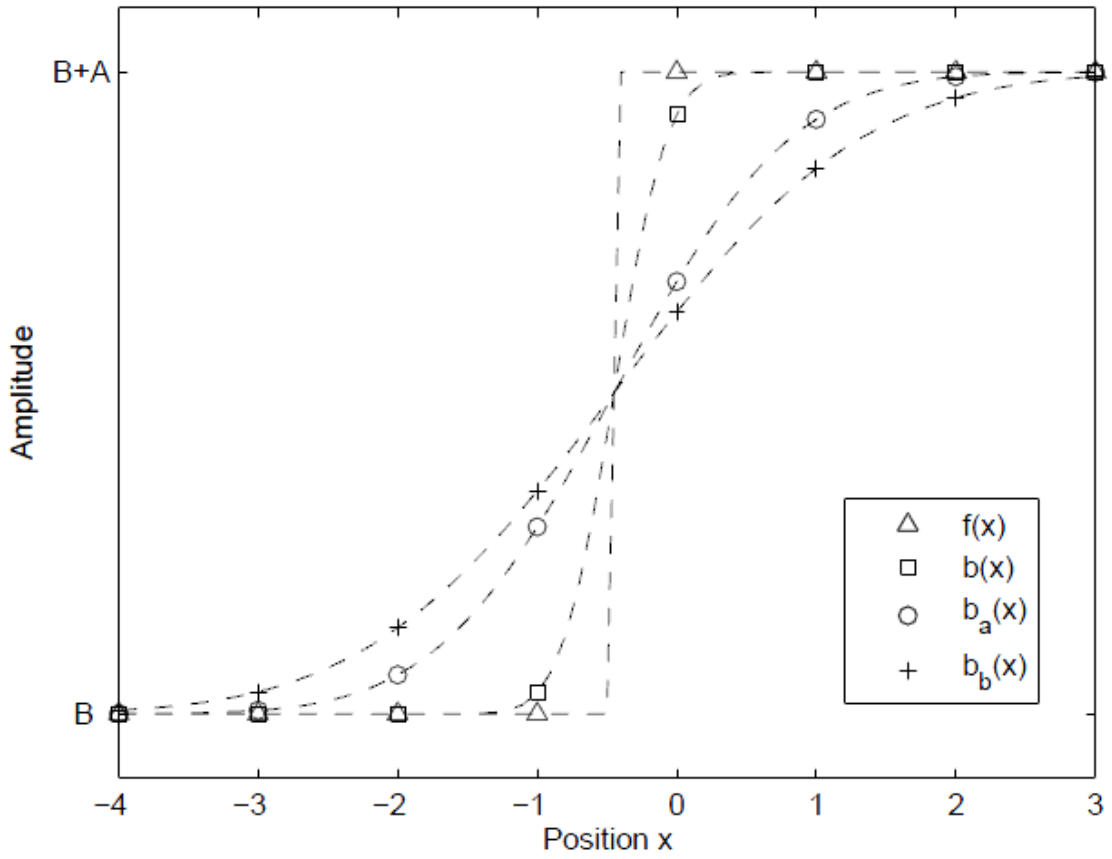

Figure 2.10 – *Evolution d'un contour appartenant à une image qui présente des artéfacts de flou de plus en plus importants. $f(x)$ représente le contour original et $b(x)$ sa version rendue floue. b_a et b_b sont les deux versions de $b(x)$ qui a été rendu flou à nouveau après convolution par un filtre Gaussien d'écart type σ_a et σ_b. [HH06].*

De nombreuses méthodes travaillent dans le domaine spatial pour inférer la qualité d'une image. Ces méthodes se basent généralement sur une détection de contours avant de procéder à une estimation de leur taille. Comme présenté dans [HH06], un contour peut être schématisé par une fonction escalier d'amplitude A et d'offset B. Un contour $f(x)$ peut donc s'exprimer comme suit :

$$f(x) =\in \begin{cases} A+B & \text{si} \quad x \geq 0 \\ B & \text{si} \quad x < 0 \end{cases}$$

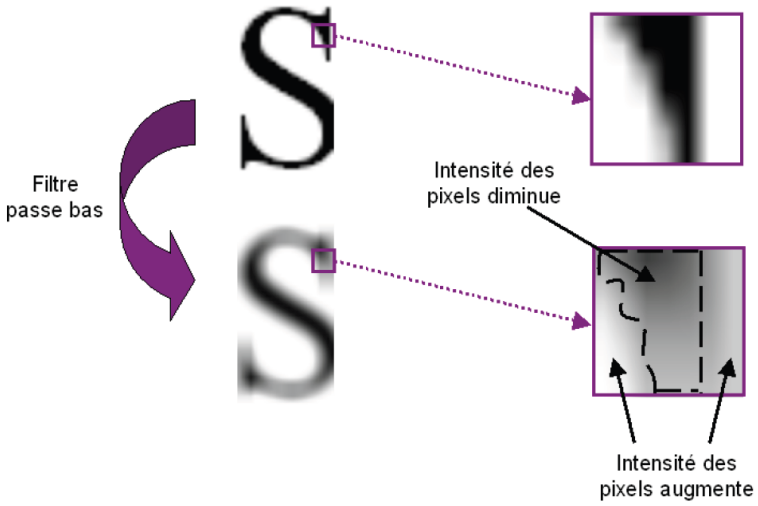

Figure 2.11 – *Illustration de l'étalement des contours et de l'atténuation de l'amplitude d'une image après application d'un filtre passe bas. [CRDLN07].*

Et comme on peut le voir sur la **Figure 2.10**, plus l'image est floue, plus le contour s'étale et l'amplitude s'atténue. Sur la figure on note $b(x)$ la version floue du contour $f(x)$ et b_a et b_b sont les deux versions de $b(x)$ qui a été rendu floue à nouveau après convolution par un filtre gaussien d'écart type σ_a et σ_b où $\sigma_b > \sigma_a$. Ceci est également illustré sur l'image de la **Figure 2.11**. L'application d'un filtre passe bas sur l'image de la lettre « S » lisse les contours de l'image. La frontière entre l'arrière plan de l'image et la lettre est beaucoup moins franche. Cela correspond bien aux observations faites sur la **Figure 2.10**.

A partir de cette observation, plusieurs méthodes permettant d'estimer les artéfacts de flou ont été proposées [FDW+02], [MDWE04], [Ong03]. L'idée de toutes ces méthodes est de corréler la quantité de flou d'une image à l'épaisseur des contours présents dans l'image.

Dans [Ong03], les auteurs proposent de calculer le gradient d'une image et de définir les orientations de chacun de ces gradients. Puis en combinant cette information à l'aide d'un détecteur de contour de Canny, ils peuvent définir l'orientation de chacun des contours contenus dans l'image. Cela leur permet de mesurer la taille d'un contour (et donc son étalement si celui-ci est sujet à du flou) en mesurant le nombre de pixels dans la direction du gradient correspondant au contour et dans la direction opposée. Cette mesure permet d'obtenir l'information relative au flou d'une image. Mais l'expression de la mesure finale obtenue dépend également de paramètres qui ont été estimés à partir d'une base de test

contenant des images qui ne sont pas forcément représentatives de l'ensemble des cas que nous pourrons rencontrer. Par conséquent, cette méthode peut ne pas s'appliquer dans tous les cas.

Dans [FDW+02] et [MDWE04], les auteurs appliquent un détecteur de contours sur l'image à l'aide d'un filtre de Sobel qui peut s'appliquer aussi bien dans les directions verticales qu' horizontales puis ils mesurent la taille d'un contour entre les deux minimums locaux les plus proches et présents de part et d'autre de ce contour. Le score final est obtenu après avoir sommé l'ensemble des largeurs de chaque contour et normalisé cette valeur par rapport au nombre de contours contenus dans l'image. Bien que cette méthode présentent de bons résultats, l'étape de recherche des maximums locaux n'est pas négligeable en temps de calcul, c'est la raison pour laquelle cette méthode n'a pas été retenue.

Les métriques de netteté (sharpness metric) permettent également d'obtenir une information sur le niveau de flou présent dans l'image. Dans [FK07], les auteurs proposent une estimation de la netteté d'une image également basée sur une mesure de l'étalement des contours. A l'aide d'une expérimentation subjective, les auteurs disposent d'une valeur de la taille d'un contour à partir duquel le flou devient perceptible dans une image. Cette valeur est notée w_{jnb}. Ils modélisent ensuite en fonction de cette valeur la probabilité de détecter une distorsion dans l'image causée par le flou et en déduisent une estimation du niveau de flou perçu dans une région R de l'image. L'inverse de ce score peut donc être utilisé en tant que métrique de flou. D'autres métriques de netteté ont été proposées dans l'état de l'art, notamment celle présentée dans [HWS10] basée sur une information obtenue dans le domaine fréquentiel. Un calcul de transformée en ondelettes est nécessaire pour obtenir une information de cohérence locale de phase. Par rapport aux méthodes proposées ci-dessus, cette dernière méthode nécessite des calculs plus complexes et nous ne considérerons donc pas cette méthode car elle est plus lente en terme de temps de calcul.

Plusieurs méthodes combinent à la fois l'information spatiale et l'information fréquentielle d'une image. C'est par exemple le cas de [CG02] qui se base sur le calcul du coefficient de Kurtosis pour mesurer la présence de l'artéfact de flou dans une image. On rappelle que le coefficient de Kurtosis est une mesure statistique qui estime l'aplatissement de la distribution $p(u, v)$ d'une variable aléatoire. Après avoir détecté les contours présents dans une image avec un détecteur de Canny, les auteurs passent dans le domaine fréquentiel en calculant la transformée en cosinus discrète au niveau d'un bloc qui entoure le contour détecté. C'est cette fonction, qui après normalisation, représente la distribution de probabilité pour laquelle le coefficient de Kurtosis est calculé. De même que précédemment, cette méthode fait appel à des calculs complexes qui demandent un temps de calcul plus longs ce qui n'est pas souhaitable dans le cadre de notre projet.

Enfin, d'autres méthodes appliquent un filtre passe-bas sur l'image originale pour la rendre plus floue. L'information recueillie au niveau de l'image originale et de l'image rendue floue artificiellement est traitée puis comparée afin d'estimer la qualité de l'image originale. C'est en effet le cas des méthodes présentées dans [HH06] et [CRDLN07]. Dans

[HH06], les auteurs modélisent une image floue par la convolution de l'image originale avec une fonction gaussienne d'écart-type σ. Le but de la méthode est alors d'estimer ce coefficient σ. Pour cela, ils utilisent deux autres filtres gaussiens d'écart-type σ_a et σ_b connus et à partir desquels ils expriment l'écart-type σ recherché. La méthode proposée dans [CRDLN07] se base quant à elle sur l'information de niveaux de gris contenue dans l'image originale et dans cette même image qui a été floutée artificiellement. Cette dernière méthode (BluM) est de mise en œuvre très simple ce qui permet de l'appliquer en temps réel. Par ailleurs, elle présente de très bon résultats par rapport aux méthodes de l'état de l'art. Très récemment, une métrique d'estimation de flou se basant sur le BluM et sur les méthodes de mesure de contour a récemment été publiée dans [LYBW11]. Les auteurs de cet article cherchent à tirer profit des deux façons de voir la mesure du flou. L'amélioration n'est pas à la mesure de la complexité introduite. Mais cette piste reste intéressante si le besoin d'améliorer encore la mesure du flou se fait à l'avenir sentir. C'est pourquoi nous avons souhaité utiliser la métrique de flou BluM dans le cadre de notre travail. Nous présentons les détails de son principe dans la section 3.2.1.

2.2.1 La métrique de flou (BluM)

La métrique de flou utilisée dans cette thèse est la métrique proposée par Frédérique Crête [CRDLN07]. Nous en résumons brièvement les principales étapes.
A partir d'une certaine limite, l'oeil humain n'est plus capable de différencier divers niveaux de flou [CRDLN07]. En effet, plus une image est floue, plus les pixels d'une même région de l'image vont converger vers une valeur moyenne commune. Ainsi, en rendant floue une image nette, les intensités des pixels d'une même zone de l'image vont varier de façon significative contrairement au cas où l'on rend floue une image qui l'est déjà et dont on aura alors beaucoup plus de mal à déterminer le niveau de dégradation. Ceci est illustrée sur la **Figure 2.12**. C'est principalement sur cette observation que se base la métrique que nous utilisons dans cette thèse bien que d'autres caractéristiques liées à notre perception du flou aient également été prises en compte. En termes de niveaux de gris cette première observation se traduit à la fois par une perte et une génération d'informations. En effet, lors de l'application d'un filtre passe-bas, les pixels nets ayant diminué d'intensité sur l'image floue, il y a une perte de variations d'informations. En revanche, cette diminution d'intensité des pixels s'est répartie sur l'image autour des zones initialement nettes. L'intensité des pixels autour du contour n'est plus nulle, c'est donc une variation d'information positive dans le sens où il est apparu de l'information. Ceci est illustré sur la **Figure 2.13**. Une seconde observation concerne le flou directionnel, c'est-à-dire l'étude des variations d'intensité entre pixels voisins selon une direction donnée. Cela signifie que pour une image rendue floue dans la direction horizontale, seules les variations entre pixels voisins dans cette direction sont faibles. Ce flou de mouvement est en effet très mal perçu par l'oeil humain qui n'arrive pas en déterminer la direction. Cela doit donc être pris en compte dans l'estimation de la dégradation. Ces deux observations ont leur importance dans la définition de l'algorithme comme nous allons le voir par la suite.

Celui-ci se divise en 4 étapes :

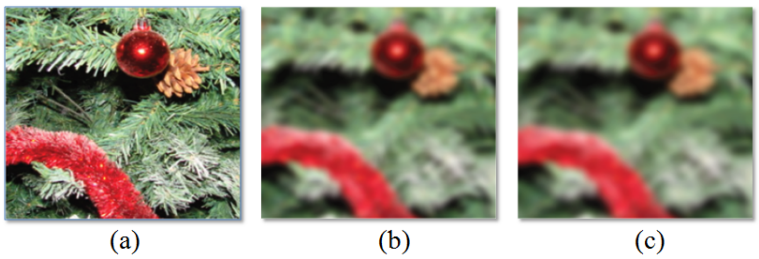

Figure 2.12 – *L'image (a) est l'image originale nette. L'image (b) correspond à l'image (a) après application d'un filtre passe-bas. L'image (c) correspond à l'image (b) après application d'un second filtre passe-bas. [FR07]*

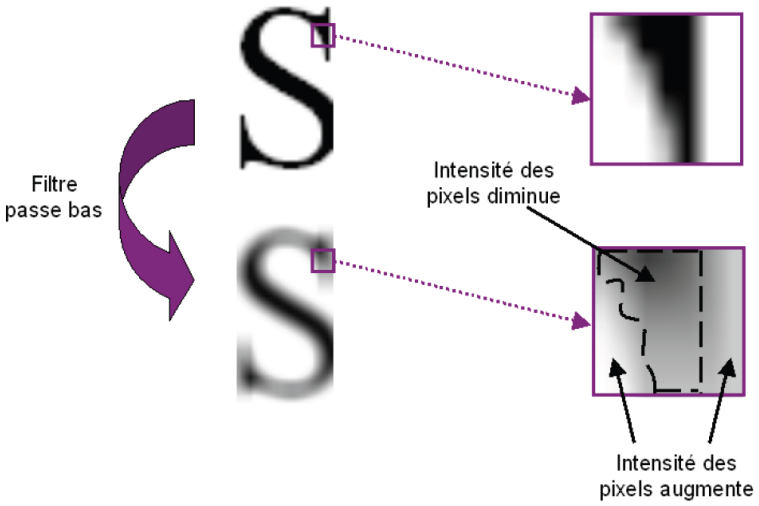

Figure 2.13 – *Illustration de la perte et du gain d'informations après application d'un filtre passe-bas sur une image, [CRDLN07].*

La **première étape** consiste, par application d'un filtre adapté, à rendre floue l'image que l'on souhaite évaluer pour obtenir une image qui sera utilisée comme référence par la suite. Cependant, pour pouvoir prendre en compte dans les prochaines étapes le flou de mouvement, l'image d'origine F est rendue floue selon les directions horizontales et verticales par application d'un filtre moyenneur à une dimension. Les images B_H et B_V ainsi obtenues sont alors utilisées toutes les deux comme référence.

La **seconde étape** détermine la différence absolue d'intensité entre pixels voisins pour chacune des images. Celle-ci se calcule pour l'image originale F dans les deux directions ainsi que pour les images floues B_H et B_V. Nous obtenons donc quatre variations de distances calculées entre un pixel $P(i,j)$ donné et le pixel juste avant lui selon l'une ou l'autre des directions ($P(i-1,j)$ ou $P(i,j-1)$) :

$$dF_{Ver} = Abs(F(i,j) - F(i-1,j)) ; \quad 1 \leq i \leq m-1 \ , \ 0 \leq j \leq n-1 \ . \tag{2.2}$$

$$dF_{Hor} = Abs(F(i,j) - F(i,j-1)) ; \quad 1 \leq j \leq n-1 \ , \ 0 \leq i \leq m-1 \ . \tag{2.3}$$

$$dB_{Ver} = Abs(B_V(i,j) - B_V(i-1,j)) ; \quad 1 \leq i \leq m-1 \ , \ 0 \leq j \leq n-1 \ . \tag{2.4}$$

$$dB_{Hor} = Abs(B_H(i,j) - B_H(i,j-1)) ; \quad 1 \leq j \leq n-1 \ , \ 0 \leq i \leq m-1 \ . \tag{2.5}$$

La **troisième étape** analyse les variations d'une image à l'autre entre ces distances en ne prenant en compte que les distances ayant diminué entre l'image originale et l'image floue pour satisfaire la première observation illustrée à la **Figure 2.13**. Seule la perte d'information est prise en compte.

$$V_{Ver} = Max(0, dF_{Ver}(i,j) - dB_{Ver}(i,j) ; \quad 1 \leq i \leq m-1 \ , \ 0 \leq j \leq n-1 \ . \tag{2.6}$$

$$V_{Hor} = Max(0, dF_{Hor}(i,j) - dB_{Hor}(i,j) ; \quad 1 \leq j \leq n-1 \ , \ 0 \leq i \leq m-1 \ . \tag{2.7}$$

La **quatrième et dernière étape** statue sur le niveau de flou de l'image selon l'importance des variations. Si les variations sont importantes, cela signifie que l'image originale est nette tandis que de petites variations traduisent le fait que l'image originale présentait déjà un niveau de flou non négligeable. Pour comparer ces variations, une somme des distances dF_{Ver}, dF_{Hor}, V_{Ver} et V_{Hor} est calculée comme suit :

$$SF_{Ver} = \sum_{i=1}^{m-1} \sum_{j=1}^{n-1} dF_{Ver}(i,j) \tag{2.8}$$

$$SF_{Hor} = \sum_{i=1}^{m-1} \sum_{j=1}^{n-1} dF_{Hor}(i,j) \ . \tag{2.9}$$

$$SV_{Ver} = \sum_{i=1}^{m-1} \sum_{j=1}^{n-1} V_{Ver}(i,j) \tag{2.10}$$

$$SV_{Hor} = \sum_{i=1}^{m-1} \sum_{j=1}^{n-1} V_{Hor}(i,j) \ . \tag{2.11}$$

Enfin, une normalisation est effectuée pour obtenir une estimation du flou sans référence avec un indicateur compris entre 0 et1 (respectivement la meilleure et la pire qualité en terme de flou) :

$$b\_F_{Ver} = \frac{SF_{Ver} - SV_{Ver}}{SF_{Ver}} \text{ and } b\_F_{Hor} = \frac{SF_{Hor} - SV_{Hor}}{SF_{Hor}} \ . \tag{2.12}$$

La valeur finale de flou $blur_F$, correspond à la valeur maximale de ces deux variables, b_F_{Ver} and b_F_{Hor} :

$$blur_F = Max\left(b\_F_{Ver}, b\_F_{Hor}\right) \ . \tag{2.13}$$

On indique sur la **Figure 2.14** l'indice de flou obtenu pour deux images dont la netteté est différente.

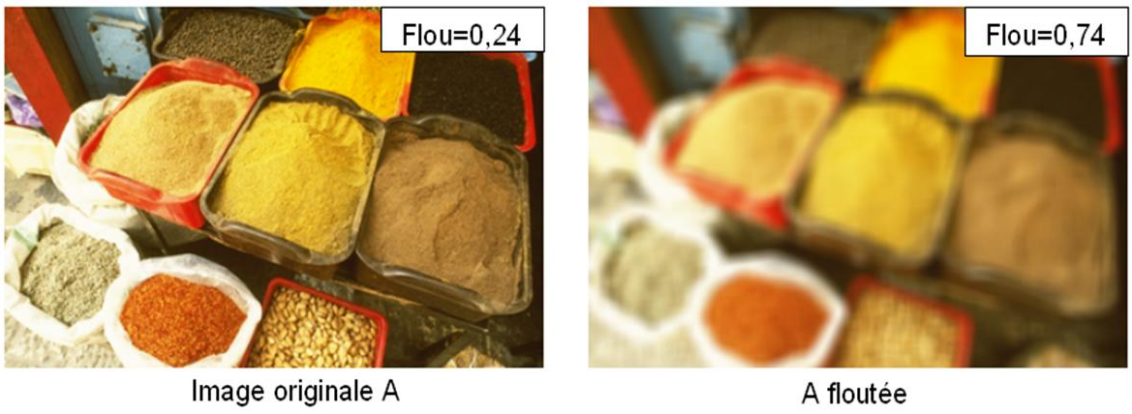

Figure 2.14 – *Indice de flou obtenu avec la métrique de flou BluM [CRDLN07].*

Une illustration de la méthode est présentée sur les figures **Figure 2.15** et **Figure 2.16** avec respectivement une image nette et une image floue.

2.3 Métrique de bloc

2.3.1 État de l'art des métriques de bloc

Les artéfacts de bloc sont essentiellement dus à l'algorithme de compression utilisé pour compresser l'image. Le standard MPEG2 nécessite de diviser l'image en blocs de 8 pixels sur 8 et une étape de quantification est ensuite appliquée sur chacun d'entre eux. Malheureusement ce type de traitement ne prend pas en compte les corrélations éventuelles entre les blocs étudiés puisque chaque bloc est traité de façon indépendante. Il apparaît alors une discontinuité de l'information au niveau des frontières entre bloc. Ce phénomène se résume par le terme d'effet de bloc que l'on illustre sur la **Figure 2.17**.

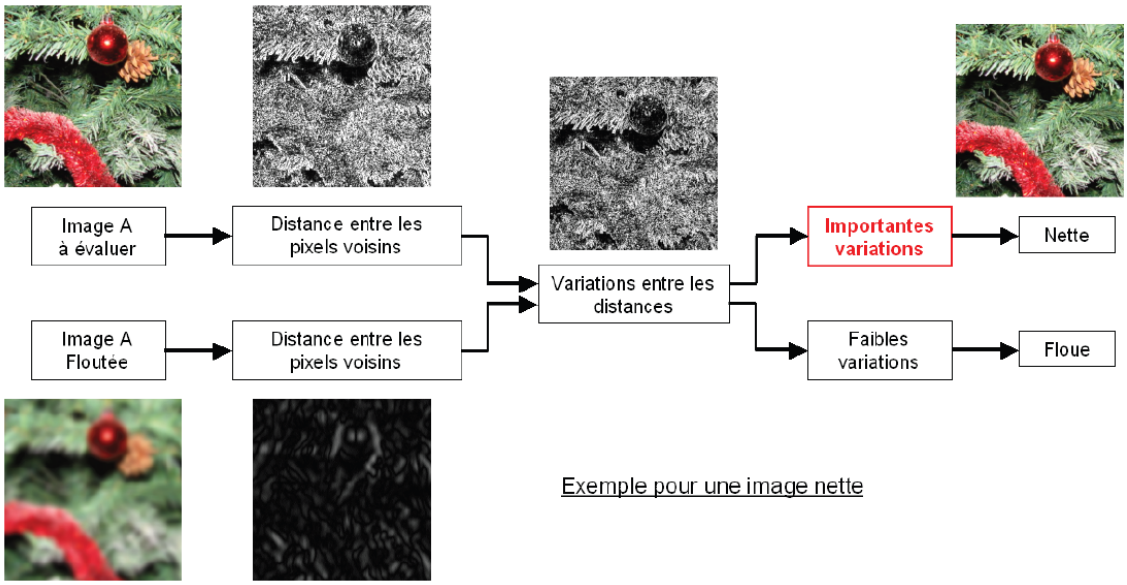

Figure 2.15 – *Illustration de la méthode proposée dans [CRDLN07] pour une image nette. [FR07]*

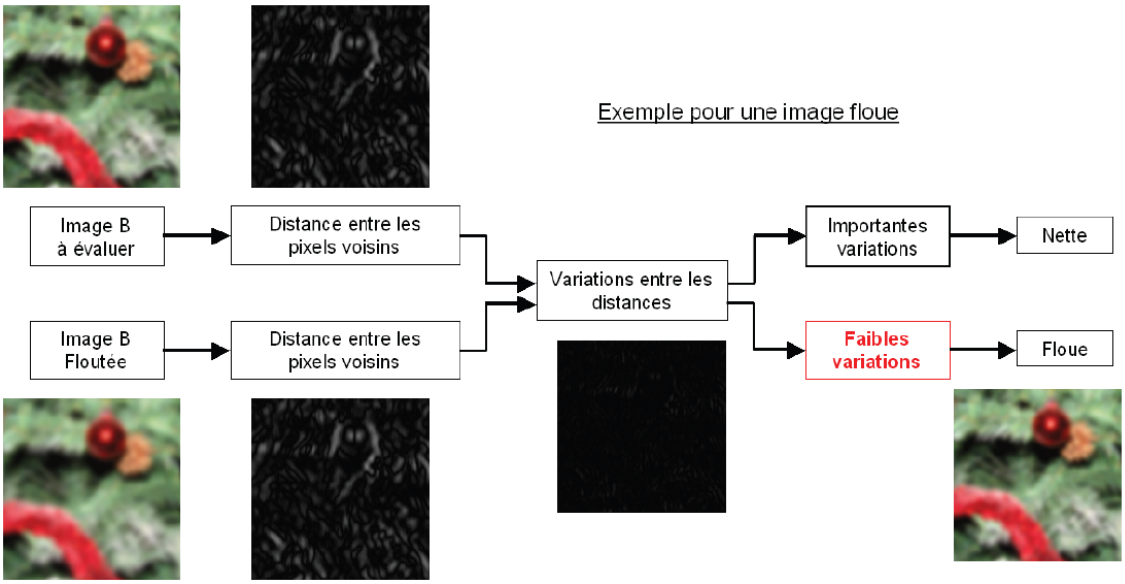

Figure 2.16 – *Illustration de la méthode proposée dans [CRDLN07] pour une image floue. [FR07]*

Figure 2.17 – *Illustration de l'effet de bloc. On voit bien l'apparition des frontières entre blocs au niveau de la mer ou du bateau comme illustré sur le zoom de l'image. [FR07]*

L'une des principales difficultés lors de l'élaboration d'une métrique de bloc est de ne pas confondre les frontières induites par effet de blocs avec les contours réels de l'image.

Les artéfacts de compression sont détectables aussi bien dans le domaine spatial que fréquentiel. Dans le domaine spatial, les effets de bloc font apparaître des discontinuités au niveau des frontières de bloc régulièrement réparties dans l'image. Cela signifie que ce motif est répété de façon régulière à des fréquences constantes ce qui se traduit dans le domaine fréquentiel par des pics réguliers sur le spectre du signal comme illustré dans [WBE00] sur la **Figure 2.18**. Le spectre représenté correspond à celui d'une image à deux dimensions que l'on a transformé en une image à une dimension pour ne considérer que les effets de blocs dans la direction verticale. Un spectre similaire peut être obtenu si l'on considère la direction horizontale.

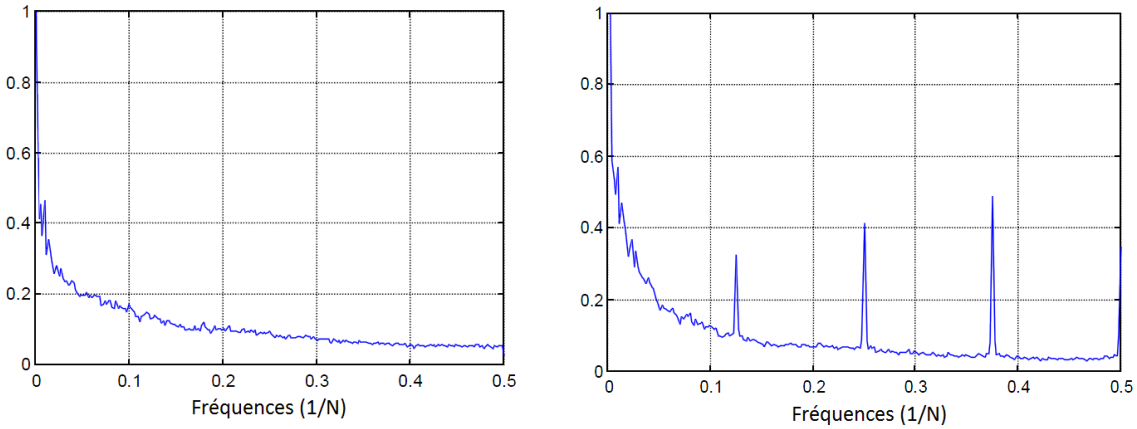

Figure 2.18 – *A gauche le spectre d'une image non compressée et à droite le spectre de la même image compressée avec un facteur de qualité Jpeg [WBE00]*

Plusieurs métriques utilisent l'information fréquentielle pour estimer la qualité d'une image en terme d'effet de blocs. C'est notamment le cas des méthodes présentées dans [WBE00] et [CB10]. Dans [WBE00], les auteurs proposent une modélisation de l'image comme étant la somme d'une image sans effet de bloc avec un signal représentant uniquement l'effet de bloc. Le but est donc d'isoler le signal associé à l'effet de bloc afin de pouvoir l'enlever de l'image et de procéder à une comparaison entre l'image présentant des effets de bloc et celle qui n'en contient pas. Cela nécessite le calcul de la transformée de Fourier du signal. Le spectre obtenu contient alors les pics caractéristiques de l'effet de bloc qui sont enlevés à l'aide d'un filtre médian. Une comparaison entre le spectre de l'image filtrée et celui de l'image originale donne une information sur les artéfacts de blocs présents dans l'image de départ. Dans [CB10], les auteurs proposent une méthode permettant d'estimer la qualité de l'image sans avoir aucun a priori sur la nature des bloc présents dans l'image (en particulier leur taille). Néanmoins, cette méthode nécessite également le calcul de la transformée de Fourier souvent lourde en terme de temps de calcul.

Les méthodes basées sur l'information spatiale ne s'accompagnent pas d'un coût de calcul important. La difficulté réside plutôt sur la définition des frontières. En effet, il est difficile de distinguer une frontière de bloc d'un contour de l'image. Et c'est généralement le principal problème de ce type de métrique de bloc. Dans [WSB02], l'effet de bloc est mesuré dans un premier temps directement au niveau des frontières de bloc. La connaissance de la taille du bloc permet de cibler directement ses frontières. Dans un second temps, l'information est recueillie au niveau des blocs eux mêmes. En effet, la compression entraînant également un effet de flou, il est important de regarder la valeur des pixels au niveau de chaque région de l'image. Pour renforcer l'évaluation de cet effet, les auteurs estiment également une mesure du nombre de passages à zéro au niveau de chaque frontière. Ces trois informations sont ensuite combinées et pondérées expérimentalement. Dans [PLR$^+$04], les auteurs recueillent également l'information au niveau des frontières et au niveau des blocs mais proposent une méthode qui permet de ne parcourir l'image qu'une seule fois en considérant uniquement, pour un bloc donné, que les frontières qui ont déjà été parcourues ce qui augmente la rapidité de la méthode. Néanmoins, les deux méthodes citées ne permettent pas de faire la distinction entre une frontière de bloc et un contour de l'image puisque toutes les frontières sont considérées. Ceci tend à fausser le résultat de l'estimation. Pour résoudre ce problème, la méthode proposée dans [PMG05] permet de définir trois régions différentes dans lesquelles l'analyse des frontières est réalisée après avoir procédé à une détection de contours à l'aide d'un filtre de Sobel. Dans [HYX08], les auteurs utilisent également les filtres de Sobel pour détecter les contours de l'image et procéder à une analyse des artéfacts de bloc. Pour cela ils utilisent l'information au niveau des blocs, comme dans [PLR$^+$04] tout en prenant en compte d'autres effets liés à la compression auxquels le système visuel humain est sensible. En particulier, il prend en compte les effets de blocs étendus qui apparaissent au niveau de région relativement homogène lorsque le taux d'échantillonnage est bas. Mais il prend également en compte les distorsions au niveau de la luminance et de la texture auxquelles le système visuel humain est également sensible. Pour notre étude, la prise en considération de ce type de distorsion n'est pas fondamental puisque, dans notre cas, nous souhaitons évaluer la qualité d'une image à travers des algorithmes de reconnaissance de visages. C'est pourquoi

nous avons choisi d'utiliser dans le cadre de notre projet la métrique de bloc proposée dans [FR07]. D'une part, contrairement aux métriques proposées dans [WSB02], [PLR+04] et [PMG05], celle-ci arrive à mieux définir les variations qui existent au niveau d'une frontière et sa visibilité ce qui permet d'affiner l'estimation de l'effet de blocs. D'autre part, le coût d'implantation de cette méthode est très faible ce qui permet son utilisation dans un contexte de vidéosurveillance. Nous présentons en détails cette métrique dans la suite de ce paragraphe.

2.3.2 La métrique de bloc : bloc Level Estimator (BLE)

Pour estimer l'intensité des artéfacts de bloc sur une image, l'idée de l'algorithme BLE est de détecter les frontières de bloc dans un premier temps puis d'évaluer leur visibilité dans un second temps. De façon intuitive, il est facile de comprendre que l'effet de bloc est beaucoup plus visible au niveau des zones homogènes d'une image qu'au niveau des parties fortement texturées comme illustré sur la **Figure 2.19**. Il est donc cohérent de centrer le travail sur ce type de régions. Néanmoins, lorsqu'une compression est faite de façon excessive, elle peut faire apparaître une suite de blocs homogènes dans l'image entre lesquels les frontières ne sont pas visibles. Ces blocs se regroupent et fusionnent formant une large zone homogène dans laquelle les frontières ne sont pas détectables par une détection de contour classique (cf. **Figure 2.20**). Pour autant, ces zones sont causées par une compression excessive et il est donc intéressant de pouvoir les détecter. C'est pourquoi, la métrique que nous utilisons dans ce travail et qui est présentée dans l'article [FR07] s'attache dans un premier temps à détecter deux types de frontière : celles existants entre deux blocs adjacents et celles que l'on qualifie d'invisible avant d'en évaluer leur impact sur la qualité de l'image elle-même.

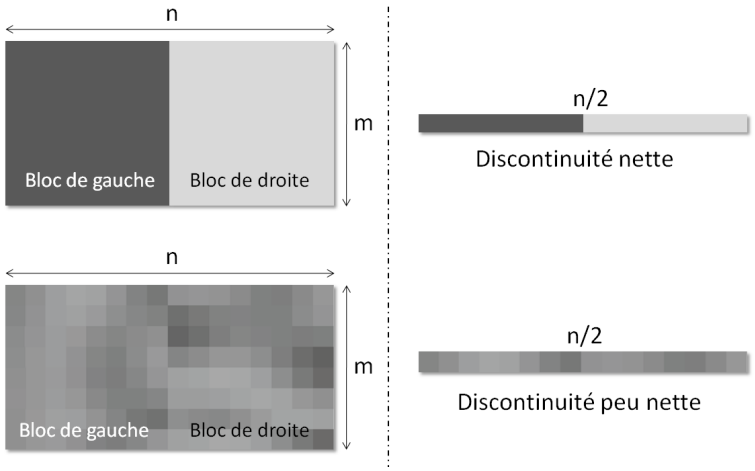

Figure 2.19 – *Illustration de l'effet de bloc au niveau des frontières entre deux blocs. [FR07]*

Figure 2.20 – *Illustration des larges zones homogènes perçues pour des taux de compression élevés. [FR07]*

Détection des frontières de blocs

Pour détecter une frontière de bloc, quelle qu'elle soit, Crete et al proposent d'étudier les variations d'informations entre deux blocs adjacents en mesurant la différence d'intensité existant entre deux pixels voisins $p(i,j)$ et $p(i,j+1)$. Pour cela les auteurs calculent dans un premier temps les coefficients $d_{i,j}$ de la matrice de variation d'information D :

$$d_{i,j} = |p(i,j) - p(i,j+1)| \tag{2.14}$$

Ainsi, si on considère deux blocs adjacents de taille 8×8 chacun, la matrice D sera composée de 8×15 termes dont les coefficients centraux $d_{i,8}$, i $\in$ {1,...,8} représentent l'information au niveau de la frontière. Dans ce cas, sur la **Figure 2.19**, $m = \frac{n}{2} = 8$. Pour préciser le type de la frontière considérée, les auteurs déterminent dans un second temps l'information contenue dans les blocs situés de part et d'autre de celle-ci qu'ils comparent, après en avoir fait la moyenne pour le côté gauche (moy_g) puis pour le côté droit (moy_d), aux coefficients centraux. En fonction des inégalités obtenues, un compteur *cpt* est incrémenté ou non. Ainsi :

$$\begin{cases} \text{Si } d_{i,8} > moy_g(i) \text{ et/ou } d_{i,8} > moy_d(i), \text{ cpt=cpt+1} \\ \text{Sinon cpt=cpt} \end{cases} \tag{2.15}$$

Les variables $moy_g(i)$ et $moy_d(i)$ sont définies comme suit :

$$moy_g(i) = \frac{1}{\frac{n}{4} - 1} \times \sum_{j=\frac{n}{4}}^{\frac{n}{2}-1} d_{i,j} \tag{2.16}$$

$$moy_d(i) = \frac{1}{\frac{n}{4} - 1} \times \sum_{j=\frac{n}{2}+1}^{\frac{3}{4}n} d_{i,j} \tag{2.17}$$

C'est la valeur finale du compteur qui détermine l'existence ou non d'une frontière visible. Les auteurs ont fixé cette valeur seuil à $cpt = \frac{3}{4}n$ pour s'assurer qu'il existe bien une discontinuité au niveau de la frontière séparant deux blocs. On définit donc le type de la frontière par la relation :

$$\begin{cases} \text{Si } cpt > \frac{3}{4}m, \text{ la frontière est visible.} \\ \text{Sinon cpt=0, la frontière n'est pas visible.} \end{cases} \tag{2.18}$$

Une fois le type de frontière déterminée, il s'agit maintenant d'attribuer un coefficient à cette dernière en fonction de la puissance de l'effet de bloc introduit dans l'image.

Attribution d'un poids par frontière détectée en fonction de la gêne visuelle qu'elle occasionne

Le coefficient que l'on associe à une frontière se fait en fonction de la visibilité de cette dernière. On pourrait donc penser que le simple critère de différence absolue défini dans le paragraphe précédent suffit à lui seul pour caractériser la visibilité d'une frontière. Malheureusement, il ne permet pas de prendre en compte à la fois le contenu des blocs et leur variabilité. En effet, comme on peut le voir sur la **Figure 2.21**, il existe des phénomènes de masquage qui diminuent la perception de la frontière de bloc alors même que la variation d'informations au niveau de la frontière est élevée. Celle-ci est en effet

Figure 2.21 – *Illustration de l'effet de masquage. Bien que la frontière soit nettement plus visible entre les deux blocs homogènes de droite, la différence absolue entre les deux blocs est plus importante à gauche, 208>160. [FR07]*

beaucoup plus évidente lorsque l'on considère deux blocs adjacents homogènes bien que la différence absolue soit inférieure. Les auteurs proposent donc un second critère que l'on associe à chaque frontière F(I,J) de l'image : la force de visibilité V_F définie à partir de deux pseudos variance V_g et V_d :

$$V_g(i) = \frac{1}{\frac{n}{4} - 1} \times \sum_{j=\frac{n}{4}}^{\frac{n}{2}-1} |moy_g(i) - d_{i,j}| \tag{2.19}$$

$$V_d(i) = \frac{1}{\frac{n}{4} - 1} \times \sum_{j=\frac{n}{2}+1}^{\frac{3}{4}n} |moy_d(i) - d_{i,j}| \tag{2.20}$$

telle que :

$$V_F = V_F + V(i) \text{ pour i=1 à m} \tag{2.21}$$

où l'on définit V(i) comme suit :

$$V(i) = \left\{ \begin{array}{ll} \frac{d_{i,j}-(moy_g(i)+moy_f(i))/2}{(V_g(i)+V_d(i))/2} & \text{Si } d_{i,j} > moy_g(i) \text{ et } d_{i,j} > moy_d(i) \\ \frac{d_{i,j}-moy_g(i)}{V_g(i)} & \text{Si } d_{i,j} > moy_g(i) \\ \frac{d_{i,j}-moy_d(i)}{V_d(i)} & \text{Si } d_{i,j} > moy_d(i) \end{array} \right\} cpt = cpt + 1 \tag{2.22}$$

$$V(i) = \left\{ \begin{array}{ll} 0 & \text{Si } cpt < \frac{3}{4}n \end{array} \right\} cpt = 0 \tag{2.23}$$

Ceci est schématisé sur la **Figure 2.22** où l'on représente chaque frontière de l'image en fonction du score obtenu pour chacune d'elle. Ainsi, une fois le coefficient de visibilité défini pour chacune des frontières de l'image, on peut définir l'estimateur de bloc final :

$$BLE = \frac{\sum_F V_F}{cpt_F r} \tag{2.24}$$

Le coefficient $cpt_F r$ permet de normaliser l'estimateur en fonction du type de frontière rencontré. En effet, une métrique doit être capable de différencier le contenu de l'image d'une frontière de bloc tout en distinguant le phénomène de bloc étendu qui apparaît sur les images fortement compressées et que nous illustrons sur la **Figure 2.20** . Ainsi ce compteur ne sera pas incrémenté lorsque la frontière étudiée appartiendra à une zone homogène de l'image pour ne pas fausser le résultat final de la métrique.

On représente sur la **Figure 2.23** une illustration des coefficients BLE obtenu pour deux images, l'une nette et l'autre fortement compressée.

La gestion des zones homogènes et des blocs étendus

Comme on peut le voir sur la **Figure 2.20**, dans le cas de fortes compressions, plus le niveau de compression est élevé, plus le nombre de frontières perceptibles diminue. C'est pourquoi si l'on normalise l'indice de bloc par le nombre de frontières perceptibles, celui-ci sera dans ce cas plus faible alors que la compression est plus élevée. C'est la raison pour laquelle il faut pouvoir détecter ces blocs étendus et attribuer un poids à chacune des frontières "invisibles". Pour autant, cette métrique étant une métrique sans référence, il est difficile de différencier un contour d'une frontière de bloc. Pour limiter les erreurs,

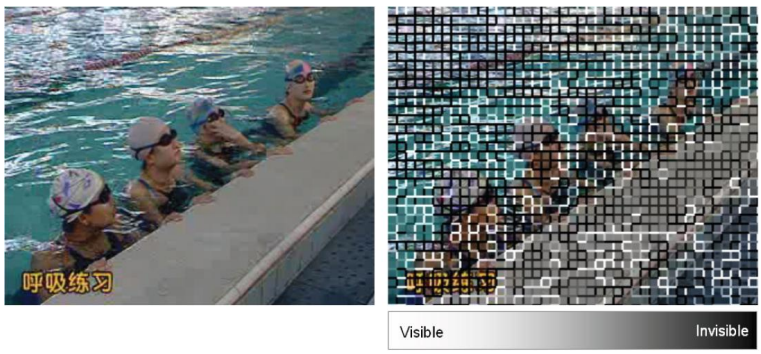

Figure 2.22 – *Illustration du critère de visibilité au niveau de chaque frontière de bloc de l'image. [FR07]*

Figure 2.23 – *Indice BLE obtenu pour deux images de niveau de compression différent. La compression est appliquée sur l'image du haut. L'effet de cette compression est observé au niveau du cadre rouge indiqué sur l'image mais le calcul du coefficient BLE est fait sur la globalité de l'image. L'indice BLE est de 5 pour l'image de gauche et de 26 pour l'image compressée de droite. [FR07]*

les auteurs se placent donc au niveau des zones fortement compressées de l'image pour procéder à la détection des blocs étendus.

Pour cela, ils définissent un bloc étendu comme étant une suite de frontières invisibles I_F qui suivent une frontière visible V_F elle-même encadrée par deux blocs homogènes. Ceci est illustré sur le schéma **Figure 2.24**.

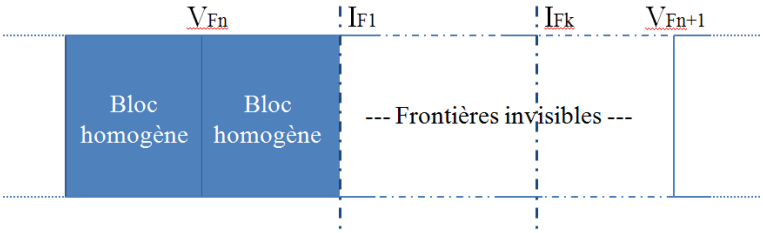

Figure 2.24 – *Un bloc étendu est toujours compris entre deux blocs homogènes et est séparé de ces blocs par deux frontières visibles V_{Fn} et V_{Fn+1}. [FR07]*

Connaissant la taille des blocs qui est généralement de 8×8 pixels, il est facile de déterminer le nombre de frontières invisibles dans le bloc étendu. Une fois ce nombre déterminé, il reste à associer un poids à chacune de ces frontières invisibles. Pour cela, les auteurs ont choisi de leur associer le poids de la première frontière visible V_{Fn}.

Conclusion

Nous avons montré au chapitre 1 que les performances des algorithmes de reconnaissance chutent lorsque l'image du visage à reconnaître présente des artéfacts de flou ou de bloc. Il est donc important, avant l'étape d'identification, d'estimer la qualité des images d'entrée afin d'évaluer la fiabilité de l'identification proposée par l'algorithme.

Pour estimer la qualité d'une image, nous nous sommes focalisés sur les artéfacts de flou et de bloc car ce sont en effet les plus visibles et les plus gênants pour des personnes non initiées. Nous avons utilisé pour cela deux métriques sans référence, respectivement le BluM [CRDLN07] pour le flou et le BLE [FR07] pour l'effet de bloc. Il est important de choisir une métrique sans référence car, dans un contexte de vidéosurveillance il est impossible de disposer de l'image originale (sans artéfact) acquise avant l'étape de compression.

Le BluM présente plusieurs avantage pour notre application. D'une part, il présente de bons résultats par rapport aux autres méthodes proposées dans l'état de l'art. D'autre part il permet de mesurer de façon objective le niveau de flou dans une image en proposant un score normalisé compris entre 0 et 1. Enfin, l'implantation de cet algorithme est très

rapide ce qui permet son utilisation dans un contexte de vidéosurveillance.

Le BLE permet d'estimer les effets de blocs présents dans une image. Cette métrique présente de bons résultats dans l'état de l'art et permet de prendre en compte les zones étendues qui apparaissent sur l'image lorsque la compression est importante. Or, nous avons vu dans le chapitre 1 que les algorithmes de reconnaissance étaient fortement dégradés pour des taux de compression élevés. Ceci est donc un point important pour notre application. De plus c'est un algorithme rapide qui peut être utilisé en temps réel et donc dans un contexte de vidéosurveillance.

Dans le chapitre 3, nous présentons l'intérêt de ces métriques pour la reconnaissance de visages et l'approche proposée pour la reconnaissance de visages sur des images floues ou avec des artéfacts de bloc.

Chapitre 3

Algorithme d'identification faciale sur des images compressées - Gestion du flou et des effets de blocs

Introduction

Comme on a pu le voir dans le chapitre 1, l'identification d'un visage impose l'utilisation d'images de référence qui forment ce que l'on appelle la galerie des visages de toutes les personnes connues par le système. Celles-ci sont prises dans de bonnes conditions et ne présentent généralement aucun artéfact. Mais il n'en est pas de même des images de test. De nombreuses études ont déjà été faites à ce sujet et les effets néfastes sur les taux de reconnaissance de la pose, de l'expression ou de l'éclairement d'un visage ont été largement étudiés [ZCPR03]. De plus en plus d'applications liées à la reconnaissance de personnes font appel à des images issues de caméras vidéo pour lesquelles les opérations d'encodage sont précédées d'un prétraitement destiné à adapter le flux d'information à la bande passante disponible [CBB+08]. L'utilisateur peut en effet réduire aussi bien la vitesse d'acquisition des images que la résolution ou le taux de compression de la vidéo. Ceci entraîne inévitablement des artéfacts, principalement de flou et des effets de bloc. L'effet de flou apparaît également lorsque la caméra ou le sujet bouge alors que le diaphragme de l'objectif est encore ouvert ou lorsque la mise au point est mal réglée ou bien lorsque la résolution de l'image est trop faible et qu'un zoom sur l'image nécessite une interpolation. L'effet de bloc apparaît quant à lui lorsque le taux de compression est trop élevé. Relativement à notre problème de reconnaissance de visages, il s'avère que des images floues ou

avec des effets de bloc sont des images pour lesquelles une partie de l'information décrivant un visage a été perdue.

Jusqu'à présent peu de travaux de recherche ont été menés sur ce type d'images. Par ailleurs, peu d'entre eux s'intéressent aux images présentant de multiples artéfacts. Pourtant, il est rare qu'une image issue d'un système de vidéo-surveillance ne présente qu'un seul artéfact. Dans l'idéal, il faudrait que toutes les dégradations générées à un instant t durant le processus d'acquisition (i.e. les problèmes d'illumination, le flou, l'effet de bloc, les problèmes de résolution etc...) puissent être maîtrisées en même temps. Dans [CMH+11], les auteurs proposent une architecture basée sur l'extraction de caractéristiques locales du visage issues d'images statiques ou de vidéos particulièrement adaptées aux applications de vidéo-surveillance car robustes aux variations de pose, d'illumination et d'expression. Néanmoins, les auteurs ne traitent ni du flou ni des effets de blocs explicitement. De plus, leur méthode est adaptée pour un protocole d'évaluation dont le but est de dire si deux images de visages qui n'ont jamais été vues appartiennent ou non à une même personne. C'est donc un travail de classification binaire : soit la paire d'images appartient à la même personne, soit non. Pour l'identification d'une personne, la tâche est plus difficile dans la mesure où il s'agit de dire si une personne se trouve dans la galerie et si oui, qui est-ce ? Le travail d'identification est d'autant plus complexe que l'on ne dispose dans la galerie que d'une seule image par personne prise dans de bonnes conditions d'acquisition.

Par conséquent, on se propose dans ce chapitre de traiter le problème de l'identification de visages sur des images acquises dans un environnement non contrôlé et donc potentiellement perturbées par deux artéfacts particuliers, le flou et l'effet de bloc.

Notre approche repose sur une stratégie en deux temps :

1. Nous estimons l'ampleur de la dégradation, le niveau de flou et la quantité d'effet de bloc.

2. Nous adaptons les images de la galerie conformément au niveau estimé de la dégradation.

Une telle approche intervient en amont de la reconnaissance faciale et permet d'être utilisée avec n'importe quel algorithme de reconnaissance.

Dans ce chapitre, notre travail s'est avant tout focalisé sur l'étude du bien fondé de l'adaptation de la galerie aux images de test pour ce type d'artéfact. Nous présentons dans un premier temps un état de l'art de la reconnaissance de visage utilisant des images présentant divers artéfacts d'acquisition. Puis dans un deuxième temps, nous détaillerons un extracteur de caractéristiques particulièrement adapté à la reconnaissance d'images floues, le descripteur LPQ [AROH08]. Enfin, nous présentons nos approches adaptées aux images floues et à l'effet de bloc ainsi que les résultats obtenus dans les sections 3 et 4.

1. État de l'art

Dans [HBK08], les auteurs ont proposé une méthode d'identification de visages permettant de prendre en compte l'ensemble des problèmes liés à une faible résolution de l'image. Néanmoins, la plupart des études menées jusqu'à maintenant traitent principalement de la reconnaissance de visage sur des images floues [AROH08], [SI00], [BK97] ce qui résulte également d'une mauvaise résolution de l'image. Très peu de travaux considèrent les problèmes liés aux effets de blocs dans une image.

On distingue de manière générale 3 approches différentes pour faire de la reconnaissance de visage sur des images dégradées :

1. Restaurer l'image à traiter
2. Estimer une dégradation limite et ignorer les images trop dégradées.
3. Traiter toutes les images dégradées ainsi que celles de bonne qualité.

La première stratégie, la plus intuitive, consiste à essayer de supprimer les dégradations et donc à essayer de restaurer l'image à traiter. Plusieurs méthodes basées sur ce principe existent déjà dans la littérature et permettent de traiter plus ou moins efficacement la reconnaissance de visages à partir d'images floues. Une première méthode consiste à restaurer l'image avant l'étape d'identification mais cela entraîne l'apparition de nouveaux artéfacts dans l'image [BK97]. Stainvas et Intrator [SI00] ont donc proposé une seconde méthode en essayant de s'affranchir de ce problème. Dans un premier temps, ils proposent dans [SMI00] une architecture de type réseaux de neurones capable de gérer aussi bien la classification que la restauration des images dégradées. L'apprentissage des paramètres du réseau se fait grâce à un modèle de type Bayésien. Néanmoins, bien qu'une combinaison de ces deux réseaux conduise à une amélioration significative du taux de reconnaissance, cette méthode nécessite d'utiliser plusieurs images par personne pour paramétrer correctement le problème. Très récemment, Nishiyama et al. [NTS$^+$09] ont proposé une méthode permettant d'inférer la fonction d'étalement du point (PSF) qui représente le processus de flou. Pour cela, un espace de représentation est crée dans lequel on peut distinguer plusieurs ensembles d'images selon les caractéristiques du flou (ou de la PSF) qui leur a été appliqué. Cette distinction entre plusieurs types de flou est apprise lors d'une phase d'apprentissage. Lors de la phase de reconnaissance, l'image d'entrée est projetée dans cette espace afin de déterminer la PSF qui se rapproche le plus de celle qui a été appliquée à l'image d'entrée. Puis, l'image est ensuite restaurée en fonction. Ce processus a été combinée à plusieurs méthodes de reconnaissance de visages dans [NHT$^+$11] et [HNS10]. Bien que la méthode proposée soit très performante, elle nécessite cependant une phase de calcul assez lourde due à l'étape de restauration de l'image.

La **seconde stratégie** pour gérer le problème de la reconnaissance sur des images dégradées consiste à déterminer le point critique à partir duquel la dégradation est telle qu'elle ne permet pas une identification correct d'un visage. Au dessus de ce point cri-

tique, les images trop abimées sont ignorées. Dans [KO05] les auteurs ont estimé le niveau critique de compression d'une vidéo à partir duquel les algorithmes de détection et de tracking ne sont plus performants. A notre connaissance, il n'existe pas de telles études appliquées à la reconnaissance de visages. Néanmoins, cette catégorie de méthode présente un désavantage majeur car elles ne permettent pas l'identification des visages dont les images sont fortement dégradées. Or nous ne disposons pas toujours d'images de bonne qualité et il est important de proposer une solution dans ce cas. Nous illustrons ceci sur des images de la base Biorafale en considérant les artéfacts de flou.

Nous présentons sur la **Figure 3.1** l'estimation du niveau de flou avec la métrique de flou BluM sur l'ensemble d'images *Object6* de la base Biorafale. Comme on peut le voir, l'indice de flou est très variable oscillant entre 0.25 et 0.71.

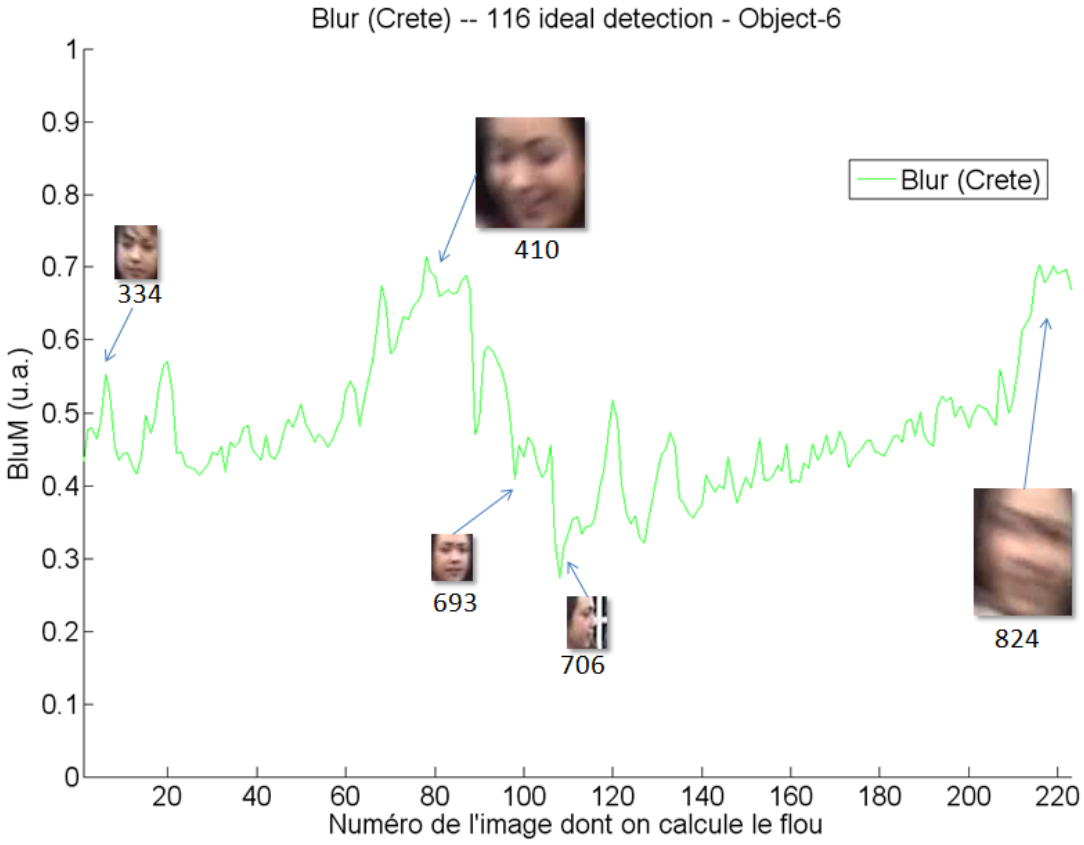

Figure 3.1 – *Étude du niveau de flou du jeu de données Object6 de la base Biorafale à l'aide de la métrique de flou proposée par Crête et al. dans [CRDLN07]. Le numéro des images dans la vidéo est indiqué sous chacune des images qui illustrent la courbe.*

Selon les auteurs qui ont développé la métrique de flou BluM, un indice de flou de 0.45 indique que cet artéfact est considéré comme légèrement gênant pour le SVH tandis qu'un indice de 0.55 indique que l'artéfact est gênant et très gênant à partir d'une valeur de 0.7.

Plusieurs images de la vidéo sont reportées sur la figure. Sur la **Figure 3.2** nous avons reporté les taux d'identification obtenus avec les méthodes basées sur les descripteurs LBP [AHP04] et LPQ [AROH08] appliquées sur des images où le flou de bougé était présent. Le détail de cette expérience est expliqué dans le chapitre 1. Nous avons reporté sur les

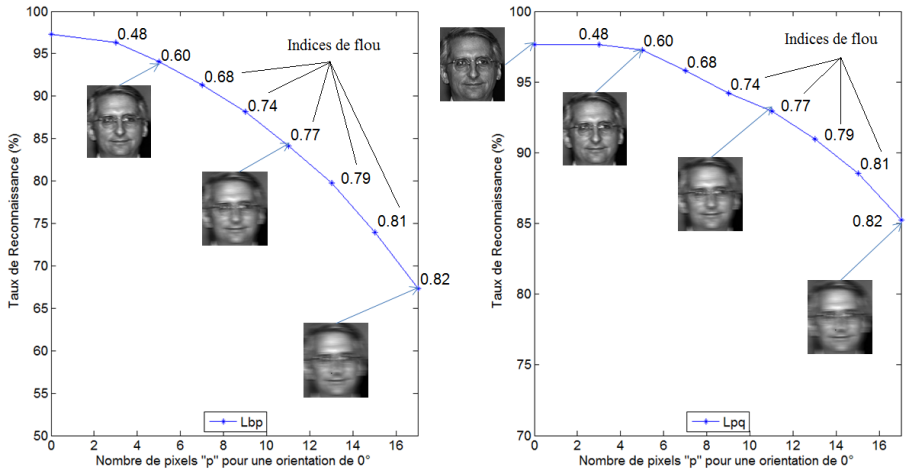

Figure 3.2 – *Taux d'identification obtenu avec les méthodes de reconnaissance basées sur les descripteurs LBP et LPQ lorsque le niveau de flou de bougé varie.*

courbes la valeur moyenne de l'indice de flou BluM obtenue sur l'ensemble des images de la base test en fonction du niveau de flou appliqué. On constate que les performances des deux algorithmes se dégradent à partir d'un niveau de flou compris entre 0.48 et 0.6 ce qui correspond à un niveau de flou perçu comme gênant selon les auteurs de la métrique. Autrement dit, nous pouvons fixer, a priori, un seuil autour de 0.6 à partir duquel nous ne considérons plus les images de la vidéo car elles sont considérées comme trop flou pour envisager une reconnaissance fiable. Ceci est illustré sur la **Figure 3.3**.

Néanmoins, cette approche présente assez rapidement des limites sur des vidéos prises dans un contexte de vidéosurveillance. D'une part, comme on peut le voir sur les images reportées sur la **Figure 3.1**, la résolution et la qualité des images est très variable. Pour pouvoir procéder à une reconnaissance de qualité, il a été montré dans [Mel09] que la distance entre les yeux du visage à identifier doit être au moins de 45 pixels ce qui nécessite une résolution de l'image supérieure. Or certaines images de la vidéo qui présentent un indice de flou relativement bas ont une résolution largement inférieure à ce seuil ce qui nécessite leur interpolation. Comme on l'a reporté dans le **Tableau 3.1**, cela entraîne une augmentation des artéfacts de flou dans l'image. Pour l'image 693 de l'ensemble de données *Object6* par exemple, lorsque la taille de l'image passe de 35*40 à 90*90, l'indice de flou passe de 0.56 à 0.8. Finalement, l'indice de flou de l'image est supérieure à 0.6 et est donc rejetée par le système. Dans ce cas, il peut arriver qu'aucune image de la vidéo ne soit suffisamment nette pour permettre la reconnaissance et il faut donc procéder à l'identification des visages même si le niveau de flou détecté dans l'image est supérieur au

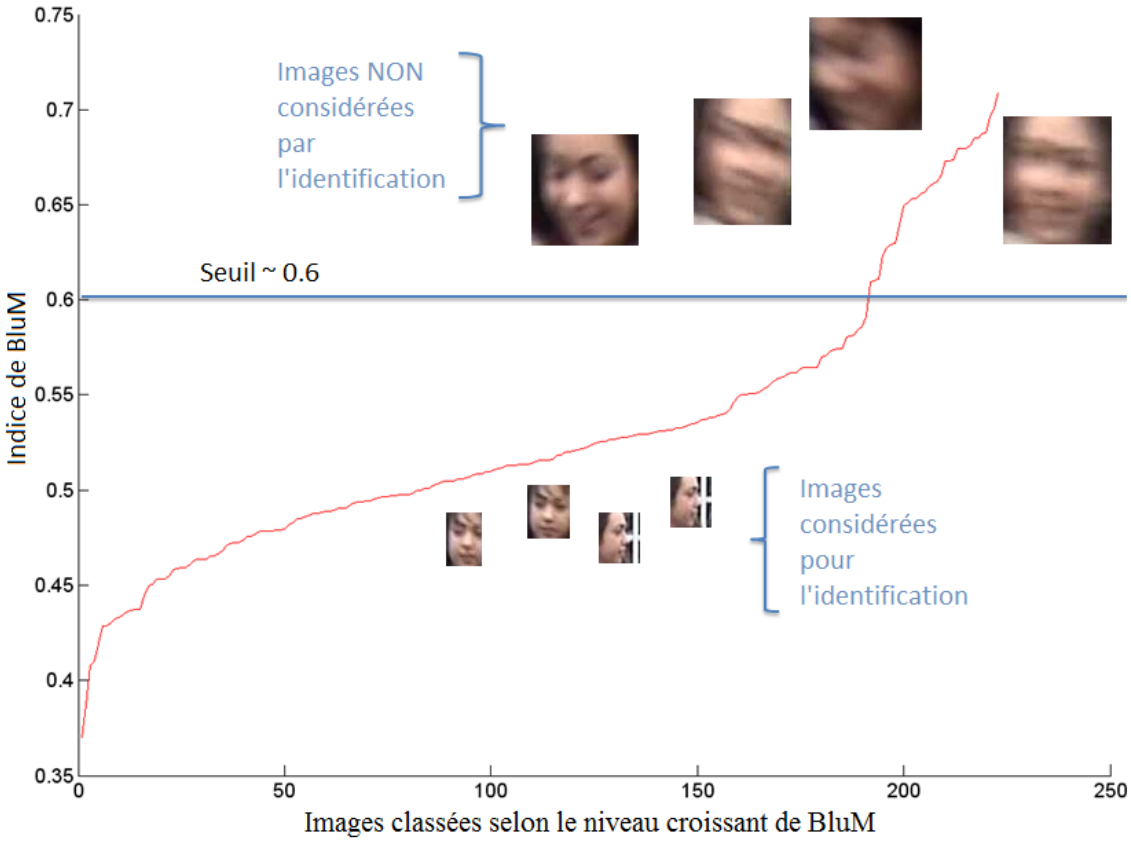

Figure 3.3 – *Définition d'un seuil de niveau de flou à partir duquel les images de la vidéo ne sont plus considérées. Ici, le seuil a été fixé à 0.6.*

seuil fixé.

D'autre part, comme illustré sur la **Figure 3.1**, certaines images présentent des portions d'images nettes causées par des objets ou du texte présent sur l'image ce qui tend à fausser l'estimation du niveau de flou.

Ainsi, définir un seuil à partir duquel le niveau de qualité de l'image n'est plus acceptable n'est pas suffisant pour prendre en compte tous les problèmes rencontrés dans un contexte de vidéosurveillance. Ce que nous venons de montrer pour le flou est aussi valable pour les artéfacts de blocs par exemple.

Il faut donc être capable de proposer une solution dans le cas où aucune image de bonne qualité n'est disponible. La qualité des données à notre disposition dépend en effet du système mais aussi de l'utilisateur et des contraintes de budget qui peuvent se poser à

Table 3.1 – *Evolution du niveau de flou lors d'agrandissement des images dont la résolution est trop faible pour la reconnaissance. Nous indiquons la taille de l'image, son numéro et la base à laquelle elle appartient ainsi que le niveau de flou d'origine et celui obtenu après agrandissement.*

Evolution du niveau de flou lors d'un agrandissement					
Base	Numéro	Taille	BluM	Taille	BluM
Object1	183	51*53	0.54	90*90	0.68
	216	72*77	0.62	90*90	0.67
	531	65*68	0.58	90*90	0.63
Object6	334	36*45	0.48	90*90	0.77
	693	35*40	0.56	90*90	0.80

l'achat [KS08].

Une **troisième stratégie** pour faire de la reconnaissance sur des images dégradées consiste à pouvoir traiter aussi bien les images de bonne qualité que des images où le niveau d'artéfacts est très important. Pour se faire, la façon la plus efficace et la plus adaptée consiste à trouver un descripteur de visage insensible aux perturbations que l'on souhaite étudier. Il s'agit donc dans notre cas de trouver un descripteur suffisamment robuste au flou ou à l'effet de bloc. C'est l'objet de la méthode d'Ahonen et al. [AROH08] qui considère l'artéfact de flou en utilisant le descripteur de texture LPQ (Local Phase Quantization) proposé dans [OH08] par Ojansivu et al. et que nous présentons en détails dans la section 2. Ce descripteur présente en effet de très bons résultats sur la base CMU et sur une base de visages dégradés volontairement de façon contrôlée. Les résultats publiés dans [AROH08] ont en effet été comparés à plusieurs descripteurs de reconnaissance récents dont le descripteur LBP [AHP04] que le descripteur LPQ surpasse sur ce genre d'images. On se propose donc, dans la section 2 d'expliquer en détail le fonctionnement de l'extracteur LPQ avant de présenter dans les sections 3 et 4 la méthode que nous proposons pour pallier les problèmes de flou et de bloc.

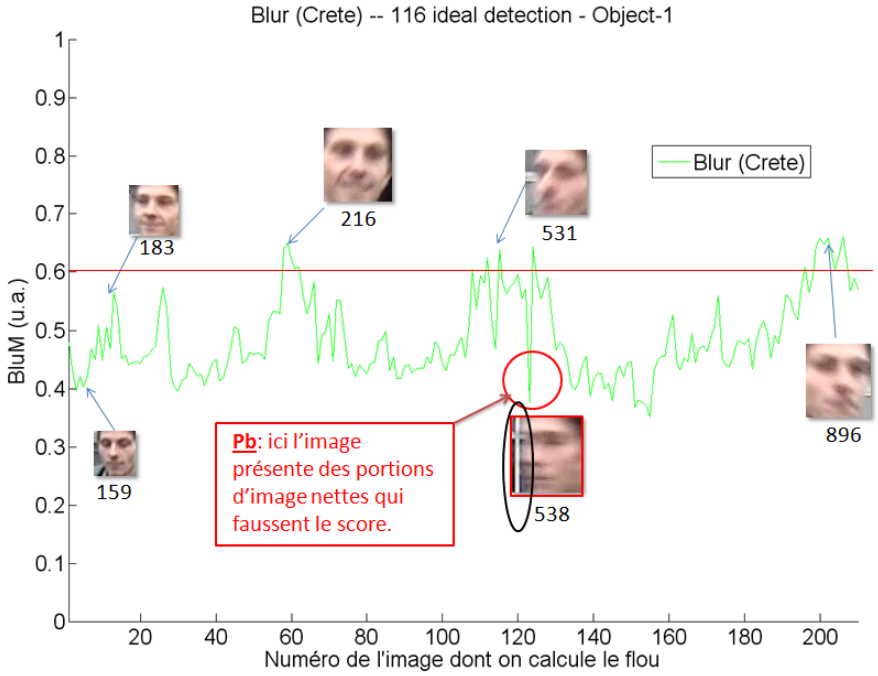

Figure 3.4 – *Étude du niveau de flou du jeu de données Object1 de la base Biorafale à l'aide de la métrique de flou proposée par Crête et al. dans [CRDLN07]. Le numéro des images dans la vidéo est indiqué sous chacune des images qui illustrent la courbe. Certaines images contiennent des régions nettes qui faussent l'évaluation de la qualité de l'image plaçant l'image en dessous du seuil critique de qualité fixé alors qu'elle est manifestement très floue.*

2. Caractérisation d'un visage par le descripteur LPQ (Local Phase Quantization)

2.1 Principe

Le descripteur de texture LPQ a été introduit pour la première fois par Ojansivu et al. [OH08]. Il permet d'améliorer la classification de textures tout en étant robuste aux artéfacts générés par différentes formes de flou présents dans une image. Pour cela, le descripteur est construit de façon à ne retenir dans une image que l'information locale invariante à un certain type de flou. Les auteurs ne considèrent en effet que les flous pouvant être représentés par une fonction d'étalement du point (PSF) présentant une symétrie centrale. Cette hypothèse sur la PSF ne limite pas pour autant l'utilisation de cette méthode étant donnée que la réponse à une source ponctuelle de la majorité des capteurs et des systèmes d'imagerie peut être modélisée par ce type de fonctions mathématiques qui peuvent également présenter des symétries d'ordre supérieure (axiale ou radiale par exemple) [FS98]. Une fois les conditions sur le flou définies, une transformée

de Fourier à fenêtre glissante est calculée pour plusieurs fréquences **f** choisies pour respecter les critères de la fonction d'étalement que nous définissons plus en détails dans la suite de cette section. Les coefficients ainsi obtenus sont quantifiés afin d'obtenir un mot de 8 bits puis, de la même façon que pour le descripteur LBP [OPM02], un histogramme est formé et est utilisé comme descripteur de texture.

2.2 Détails de la méthode

Définition du critère d'invariance au flou

Soit $t(\mathbf{z})$ une image et $d(\mathbf{z})$ sa version dégradée par du flou et définie comme étant le produit de convolution à deux dimensions entre l'image originale et la fonction d'étalement du système $g(\mathbf{z})$. La PSF présentant une symétrie centrale, on peut donc écrire que $g(\mathbf{z}) = g(x, y) = g(-x, -y)$ et on peut exprimer $d_f(\mathbf{z})$ telle que :

$$d(\mathbf{z}) = t(\mathbf{z}) \star g(\mathbf{z}). \tag{3.1}$$

ce qui correspond à la relation suivante dans le domaine de Fourier, **f** étant le vecteur de coordonnées $[f_x, f_y]$:

$$D(\mathbf{f}) = T(\mathbf{f}) \times G(\mathbf{f}). \tag{3.2}$$

où $D(\mathbf{f})$, $T(\mathbf{f})$ et $G(\mathbf{f})$ sont les transformées de Fourier discrète des fonctions $d(\mathbf{z})$, $t(\mathbf{z})$ et $g(\mathbf{z})$ respectivement. Cette dernière expression peut également être exprimée en fonction des amplitudes et des phases de ces fonctions :

$$|D(\mathbf{f})| = |T(\mathbf{f})| \times |G(\mathbf{f})| \tag{3.3}$$

$$\phi_d(\mathbf{f}) = \phi_t(\mathbf{f}) + \phi_g(\mathbf{f}) \tag{3.4}$$

Or la PSF est supposée être à symétrie centrale ce qui ajoute une condition sur la phase de la fonction d'étalement ϕ_g qui s'annule alors pour toutes valeurs positives de $G(\mathbf{f})$:

$$\phi_g(\mathbf{f}) = \begin{cases} 0 & \text{Si G}(\mathbf{f}) \geq 0 \\ \pi & \text{Si G}(\mathbf{f}) < 0 \end{cases} \tag{3.5}$$

Autrement dit, lorsque la transformée de Fourier de la PSF est positive sa phase ϕ_g s'annule. La phase de l'image originale ϕ_t et celle de l'image floue ϕ_d sont alors égales. Autrement dit, puisque ϕ_d ne dépend plus de ϕ_g, la phase de l'image floue ϕ_d devient indépendantes de la dégradation. Cette condition peut être obtenue pour certaines valeurs de la PSF associée à un flou de bougé ou un flou de mise au point pour lesquels la réponse impulsionnelle g(**z**) peut être modélisée par une porte dans le domaine temporel et donc un sinus cardinal dans le domaine fréquentiel. Le sinus cardinal étant positif autour de zéro par exemple.

Application de la transformée de Fourier à fenêtre glissante et quantification des coefficients

L'objectif de cette étape est d'extraire les coefficients spectraux indépendants du flou.

Une transformée discrète de Fourier définie comme suit est réalisée sur un voisinage carré N_z de $M \times M$ voisins pour chaque pixel $z = (x, y)$ de l'image t.

$$T(\mathbf{f}, \mathbf{z}) = \sum_{\tau \in N_z} t(\mathbf{z} - \tau) e^{-i2\pi \mathbf{f}^T \tau} = \mathbf{w_f}.\mathbf{t_z} \tag{3.6}$$

Où $\mathbf{w_f}$ correspond aux vecteurs de base de la décomposition à la fréquence $\mathbf{f}$ et $\mathbf{t_z}$ contient toutes les valeurs de l'image appartenant au voisinage N_z. La transformée de Fourier est alors calculée pour seulement 4 fréquences $\mathbf{f_i}$ pour lesquelles la phase de la DFT est invariante au flou symétrique [OH08] : $f_1 = [a, 0]$, $f_2 = [0, a]$, $f_3 = [a, a]$ and $f_4 = [a, -a]$. où a est un scalaire choisi pour satisfaire ces conditions. Par la suite, le signe de la partie réelle et celui de la partie imaginaire de chaque coefficient de Fourier sont extraits pour former un coefficient binaire $q_j(\mathbf{z})$:

$$q_j(\mathbf{z}) = \begin{cases} 0 & \text{si } s_j(\mathbf{z}) \geq 0 \\ 1 & \text{sinon} \end{cases} \tag{3.7}$$

où $s_j(\mathbf{z})$ est la $j^{ème}$ composante du vecteur $S_z = [Re\{T(\mathbf{f}, \mathbf{z})\}, Im\{T(\mathbf{f}, \mathbf{z})\}]$ où $j \in \{1, ..., 8\}$ Le mot formé des 8 bits obtenus donne l'information de phase recherchée. Enfin, un label de l'image est construit en représentant le code comme une valeur entière comprise entre 0 et 255. La **Figure 3.5**, a), b), résume l'ensemble de ces étapes.

$$F_{LPQ}(\mathbf{x}) = \sum_{j=1}^{8} q_j(\mathbf{x}) 2^{j-1} \ . \tag{3.8}$$

Néanmoins, pour préserver au maximum l'information lors de la quantification, l'indépendance entre les échantillons doit être maximisée. Cela revient à ré-exprimer les coefficients de S_z dans une matrice E_z où les coefficients sont dé-corrélés les uns des autres. Dans ce cas, l'étape de quantification (3.7) se fait sur la matrice E_z et non plus S_z. Deux pixels voisins dans une image étant fortement corrélés on peut exprimer la covariance entre deux pixels voisins $\mathbf{z_i}$ et $\mathbf{z_j}$ comme suit :

$$\sigma_{i,j} = \rho^{||\mathbf{z_i} - \mathbf{z_j}||} \tag{3.9}$$

ρ étant compris entre 0 et 1, cela signifie que plus les deux pixels sont éloignés l'un de l'autre, plus la covariance diminue. Dans cette méthode, ρ est fixé à 0.9. L'ensemble de ces coefficient $\sigma_{i,j}$ permet de déterminer la matrice de covariance C de l'ensemble des pixels contenus dans le voisinage N_z à partir duquel on détermine la matrice E_z. Une extension de la méthode présentée ici a été proposé dans [CKP+09] où les auteurs combinent l'information multi-échelle recueillie pour plusieurs voisinages N_z.

Formation du descripteur de visage

En 2008, Ahonen et al. [AROH08] ont introduit un algorithme d'identification de visages qui utilise ce descripteur de texture LPQ. Dans toute la suite du manuscrit, nous

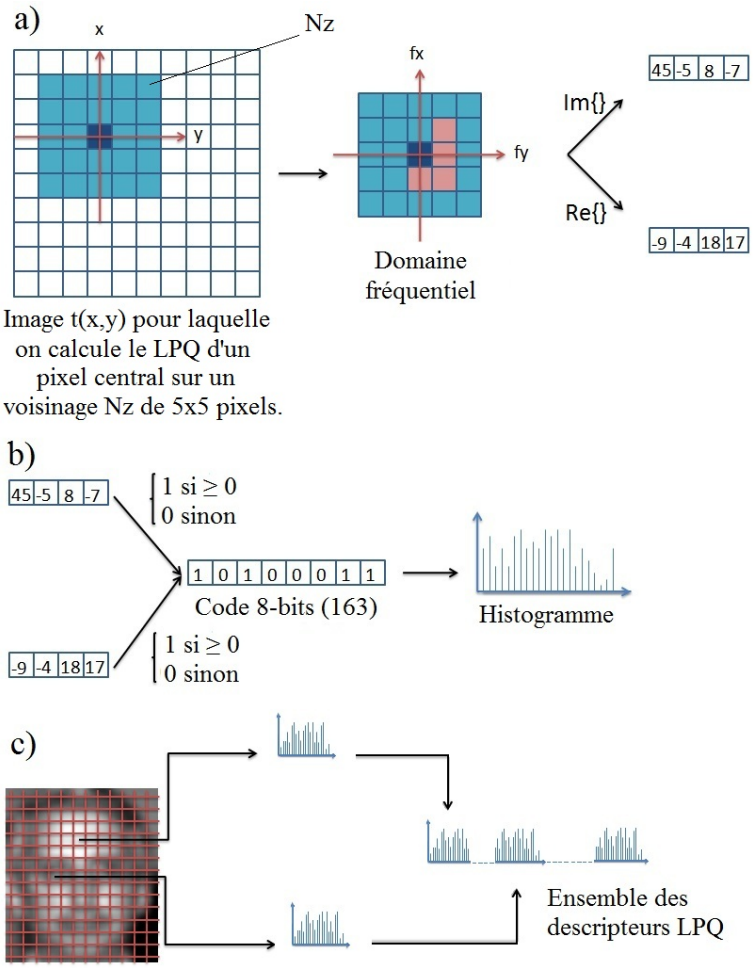

Figure 3.5 – *Organigramme de l'ensemble des étapes nécessaire à la construction du descripteur de visage LPQ. Schéma adapté de [PH11]*

faisons référence à cette méthode de reconnaissance de visages sous l'appellation abrégée « méthode LPQ ». De la même façon que pour la méthode LBP, la méthode LPQ peut être résumée en 4 étapes distinctes. Dans un premier temps, l'opérateur est appliqué sur l'image d'entrée pour obtenir l'image labelisée. Ensuite, l'image obtenue est divisée en petites régions. Pour chacune d'entre elles, un histogramme des étiquettes est construit afin d'obtenir des descripteurs locaux de visage. La représentation globale du visage est obtenue par combinaison de tous les descripteurs. La partie *c*) de la **Figure 3.5** résume l'ensemble des étapes nécessaires à la formation de ce descripteur.

Nous venons de détailler dans cette partie le seul descripteur de visages robuste aux variations de flou existant à notre connaissance dans l'état de l'art. Les résultats obtenus par la méthode LPQ sur la base CMU PIE sont présentés dans l'article [AROH08]. Les

performances de cet algorithme sont comparées au descripteur LBP. On constate que le descripteur LPQ est beaucoup moins sensible aux variations de flou que ne l'est le descripteur LBP. Néanmoins, le test a été effectué sur des images présentant un flou gaussien dont l'écart type varie entre 0 et 2. Or, les images issues de caméra vidéo surveillance peuvent présenter des flous beaucoup plus importants. Par ailleurs, le flou présent dans une image peut aussi bien être modélisé par un flou de bougé. C'est pourquoi dans la suite de ce projet nous avons souhaité tester les performances de ces deux algorithmes pour différents flous d'intensité variable.

Par ailleurs, outre ses qualités vis-à-vis du flou, ce descripteur présente deux avantages intéressants pour notre problématique. D'une part, il a été montré dans [AROH08] que les performances de ce descripteur sont meilleures que le descripteur LBP qui a été utilisé dans de nombreuses méthodes de reconnaissance en présence et en absence de flou. D'autre part, il présente également de très bons résultats sur des images de la base CMU présentant de larges variations d'illumination. Autrement dit, ce descripteur semble pouvoir s'adapter à au moins deux des artéfacts que nous rencontrons sur des images de vidéo surveillance. Par conséquent, il est également intéressant de connaître dans quelle mesure ce descripteur semble robuste aux autres artéfacts que nous rencontrons dans le cadre de notre projet et en particulier l'artéfact de bloc.

3. Approche proposée pour une reconnaissance de visage sur des images floues

Un système de reconnaissance doit être capable de reconnaître une personne donnée lorsqu'on lui présente son visage en entrée. Pour cela, les systèmes de reconnaissance automatiques doivent être capables de comparer une image de visage acquise avec une caméra vidéo, à un fichier de référence contenant les images de personnes à reconnaître et proposer une classification. Généralement, l'analyse de visage se fait à l'aide d'un descripteur capable de représenter un visage de façon à pouvoir le distinguer d'un autre tout en s'affranchissant au maximum des problèmes liés à l'acquisition de l'image. La classification se fait en comparant le vecteur de caractéristiques de l'image d'entrée ainsi obtenu à chacun de ceux du fichier de référence. Or, la galerie de référence est généralement formée d'images de bonne qualité contrairement aux images d'entrée à partir desquelles on souhaite effectuer la reconnaissance ce qui fausse les résultats. La gamme de fréquences des images test ne correspond pas à celle des images de la galerie. Autrement dit, les visages test et les visages contenus dans la galerie de référence ne contiennent pas le même type d'informations. Il s'agit ainsi de trouver l'information utile à la reconnaissance commune aux deux bases pour que les caractéristiques extraites par le descripteur de visage soient semblables et comparables afin d'obtenir une bonne identification.

C'est pourquoi nous proposons un prétraitement qui permette d'adapter la galerie de référence à l'image test. Ce prétraitement est divisée en une étape de construction et une étape de reconnaissance. L'étape de construction consiste à filtrer avec plusieurs filtres passe-bas d'intensité variable la galerie de référence originale afin d'obtenir plusieurs sous galeries de référence, chacune d'entre elles regroupant les images de la galerie originale qui présentent un degré de flou similaire. Cette étape permet ainsi d'éliminer le trop plein d'information contenu dans le jeu de données de référence pour garder les caractéristiques du visage les plus pertinentes en fonction du degré de flou de l'image test. L'étape de reconnaissance quant à elle utilise une métrique de flou qui permet d'évaluer le degré de flou de l'image test et de choisir la galerie de référence qui permettra de maximiser le taux de reconnaissance, à savoir celle qui contient les images de degré de flou similaire. Cette méthode est simple à mettre en œuvre et s'avère très efficace car elle peut être utilisée avec différents algorithmes de reconnaissance et ne demande aucune connaissance a priori sur le modèle de flou de l'image d'entrée.

3.1 Étape de construction

Il s'agit dans cette étape de construire un ensemble de galeries qui s'adapte au mieux aux images de test. C'est à dire constituées d'images de qualité similaire à celle de l'image test. Autrement dit, il s'agit dans cette étape d'introduire sur les images de la galerie le même flou que celui qui sera rencontré en conditions réelles. Pour cette étape, nous avons choisi un filtre gaussien. Simple à mettre en œuvre, il permet de simuler le flou de mise au point d'une caméra très souvent rencontré sur les images issues de caméra vidéo-surveillance. Toutes les images de la galerie ont donc été convoluées avec un filtre noté $Fg_{(s,h)}$ et représentant un filtre gaussien d'écart type s et de taille $h \times h$ tel que :

$$I_{i,s,h} = I_i \star Fg_{(s,h)} \tag{3.10}$$

Ainsi, à partir de ces paramètres, on ajoute à la galerie originale $T_{(0,0)}$ contenant les n images nettes I_i, $i \in \{1, ..., n\}$ t galeries $T_{(s,h)}$, chacune d'entre elles étant formée par les n images filtrées $I_{i,s,h}$. Une fois les $t+1$ galeries ainsi formées, un coefficient est alloué pour chacune d'entre elles afin de déterminer son niveau de flou moyen. Pour cela, nous utilisons la métrique de flou présentée au chapitre 1 : the Blur Metric (BluM). La détermination de ce coefficient se fait comme suit : dans un premier temps, la métrique est appliquée sur chacune des images $I_{i,s,h}$ de sorte que nous obtenons pour chacune d'entre elles une estimation sans référence $b_{i,s,h}$ du flou définie par :

$$b_{i,s,h} = max(b\_F_{Ver}, b\_F_{Hor}) \tag{3.11}$$

Ainsi, pour une base de galerie $T_{(s,h)}$ donnée, on peut associer un vecteur de taille n regroupant l'ensemble des coefficients associés à chacune des n images de la galerie considérée. Dans un second temps, chacun de ces jeux de coefficients est moyenné pour obtenir un seul coefficient $b_{s,h}$ à associer à une galerie donnée $T_{(s,h)}$ telle que :

$$b_{s,h} = \sum_{i=1}^{n} b_{i,s,h} \tag{3.12}$$

Finalement, nous obtenons un vecteur $B_{s,h}$ de taille $t+1$ contenant l'ensemble des coefficients $b_{s,h}$ pour lequel $b_{0,0}$ fait référence au coefficient de flou associé à la base originale (soit $b_{0,0} = 0$, pas de flou du tout). A noter que cette étape de construction n'a à être exécutée qu'une fois seulement. Un organigramme de cette étape est donné en **Figure 3.6**.

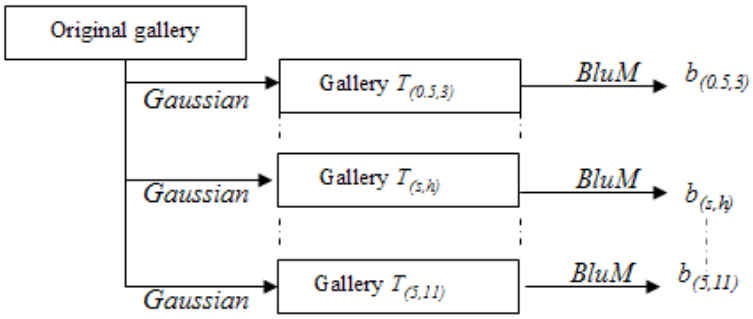

Figure 3.6 – *Organigramme de l'étape de construction.*

3.2 Étape de reconnaissance

Au cours de cette étape, nous utilisons une seule et même méthode de reconnaissance. Celle-ci consiste simplement à comparer le vecteur de caractéristiques associé à une image test aux vecteurs de caractéristiques correspondant à chacune des images de la galerie. Cette comparaison se fait grâce à une simple mesure de distance de type $\chi 2$. Nous utilisons en revanche deux méthodes permettant d'extraire les caractéristiques du visage. La première méthode d'extraction de caractéristiques est la méthode LBP détaillée dans le chapitre 2 et présentée par Ojala et al dans [OPM02]. La seconde méthode d'extraction de caractéristiques est la méthode LPQ détaillée en début de ce chapitre 3 et présentée par Ojansivu et al dans [OH08]. Le but de notre approche est d'adapter la galerie en fonction du degré de dégradation de l'image d'entrée pour permettre une extraction optimale des caractéristiques du visage considéré ce qui permettra d'améliorer la reconnaissance du visage. Dans un premier temps, nous estimons l'intensité du flou présent sur l'image test à l'aide de la métrique de flou BluM. Grâce à cette métrique, nous sommes capables d'estimer l'intensité de la dégradation quelque soit la forme du flou qui a été introduit dans l'image. Nous obtenons un coefficient b_t que nous attribuons à chacune des images test. Cette estimation est alors comparée à l'ensemble des coefficients $b_{(s,h)}$ que nous avons obtenus lors de la phase de construction. Pour cela nous avons utilisé la norme L2, $\|b_t - b_{(s,h)}\|$, que nous avons calculée pour chacun des t coefficients $b_{(s,h)}$. Le minimum est obtenu pour les paramètres s, h suivants :

$$(s,h) = \arg\min_{s,h} \|b_t - b_{(s,h)}\| \tag{3.13}$$

qui déterminent la galerie à utiliser pour la reconnaissance. Enfin, les vecteurs caractéristiques associés aux visages sont obtenus soit à partir de la méthode d'extraction de caractéristiques LBP, soit à partir de la méthode d'extraction de caractéristiques LPQ. Un organigramme de cette étape est donné en **Figure 3.7**.

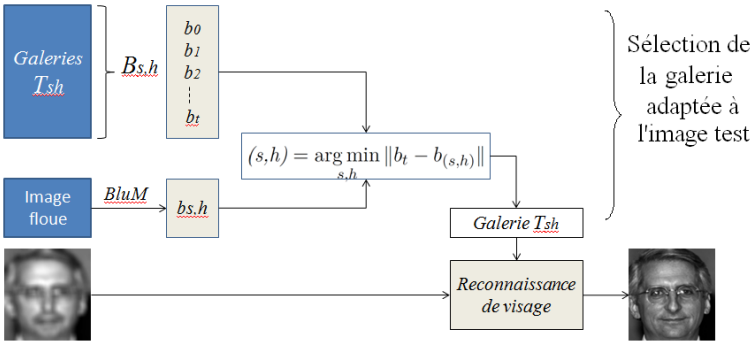

Figure 3.7 – *Organigramme de l'étape de reconnaissance.*

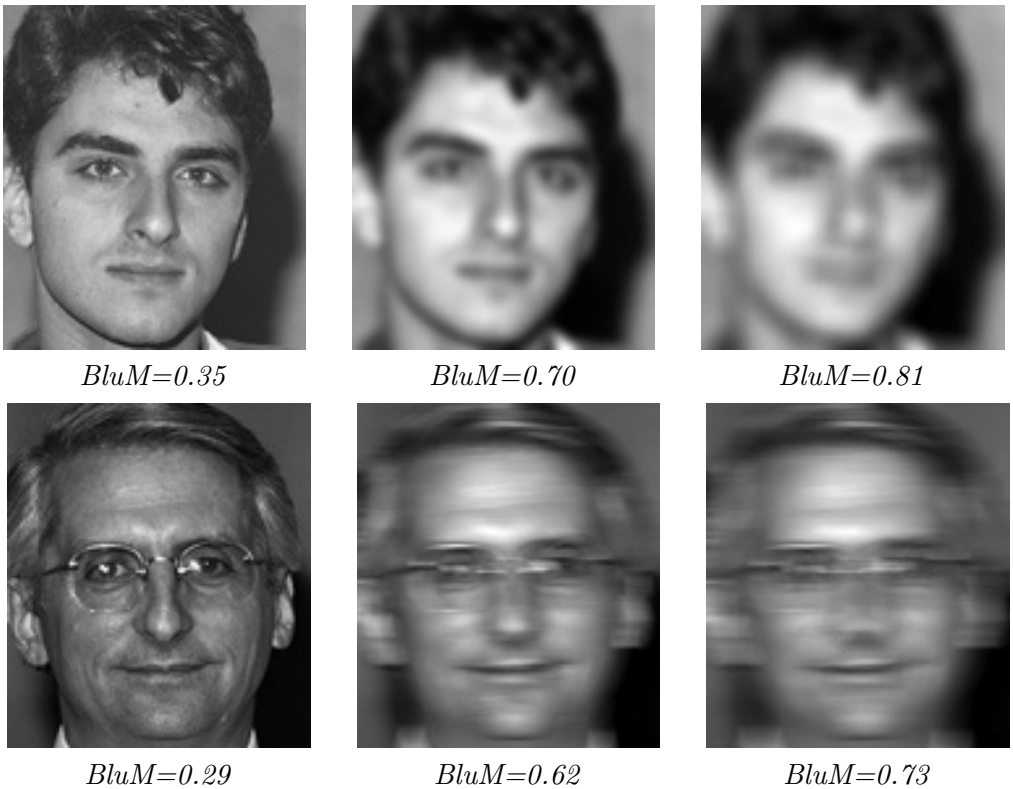

Figure 3.8 – *Première ligne de gauche à droite : image originale, image originale floutée avec un filtre gaussien $Fg_{(2,11)}$, image originale floutée avec un filtre gaussien $Fg_{(5,11)}$. Deuxième ligne de gauche à droite : image originale, image originale floutée avec un filtre de bougé $Fm_{(11,0)}$, image originale floutée avec un filtre de bougé $Fm_{(17,0)}$.*

3.3 Description de la base d'images utilisées

Pour valider l'efficacité de notre approche, nous avons utilisé un jeu de données construit à partir de la base FERET [PMRR97]. Cette base d'images est constituée de 14051 images en noir et blanc de visages. Dans le cadre de notre projet, nous avons utilisé 1194 images de visages vus de face issues des jeux de données fa et fb. Les performances de notre algorithme ont été testées sur des images appartenant au jeu de données fb. Chacune d'entre elles a été floutée artificiellement avec deux filtres, $Fg_{(s,h)}$ et $Fm_{(p,t)}$, différents. $Fg_{(s,h)}$ fait référence à un filtre gaussien d'écart type $s = \{1, 2, 3, 4, 5\}$ et de matrice carré de taille $h = \{3, 5, 7, 9, 11\}$. Autrement dit, nous avons testé notre approche sur les images provenant de 25 galeries différentes correspondant à 25 intensités différentes de flou gaussien. $Fm_{(p,t)}$, modélise quant à lui le flou de bougé souvent observé lorsque la caméra bouge de $p = \{3, 7, 9, 11, 13, 15, 17\}$ pixels avec un angle $t = \{0, 45, 90, 135\}$ degrés par rapport au sujet. De la même façon que pour le flou de mise au point, cela revient à dire que notre algorithme a été testé sur 6 galeries correspondant à 6 flous de bougé différents. Chacune de ces galeries contient 1194 images. Un exemple d'images originales et floutées est donné en **Figure 3.8**.

3.4 Analyse des résultats

Deux séries d'expériences ont été effectuées pour observer le comportement de notre algorithme face à différentes formes de flou. Dans la première, nous avons flouté artificiellement les images avec un filtre gaussien tandis que dans la seconde, les images ont été dégradées avec un filtre de flou de bougé. Pour chacune de ces expériences, notre approche a été testée avec deux méthodes de reconnaissance différentes afin de valider sa fiabilité.

Dans un premier temps, nous avons combiné notre approche avec la méthode de reconnaissance basée sur le descripteur de visage LBP. Il était intéressant d'utiliser cette méthode car c'est l'une des méthodes les plus couramment utilisées en reconnaissance de visage. En effet, elle permet d'extraire des données caractéristiques du visage invariantes à différents facteurs de distorsion tels que la pose, l'illumination et l'expression. En revanche, étant donné que la principale cause de flou dans une image est due à un filtrage passe-bas, c'est une méthode qui est a priori relativement sensible au flou. En effet, dans le domaine spatial, l'action de floutage entraîne de petites variations d'intensité entre pixels voisins. Étant donné que le descripteur LBP est un descripteur local de texture qui se base sur la valeur des pixels d'un voisinage donné, il ne peut, être invariant au flou.

Dans un deuxième temps, nous avons utilisé la méthode de reconnaissance basée sur le descripteur LPQ qui se veut robuste à de faibles variation de flou mais dont les performances diminuent lorsque le flou devient plus important.

Nous présentons les résultats obtenus pour un filtre Gaussien $Fg_{(s,11)}$ avec un écart type $s = \{0, 1, 2, 3, 4, 5\}$ croissant et un filtre de bougé $Fm_{(p,t)}$ avec un nombre de pixels $p = \{0, 3, 5, 7, 9, 11, 13, 15, 17\}$ croissant et un angle $t = \{0°, 45°, 90°, 135°\}$. Cela inclut les flous les plus forts que nous ayons appliqués à nos images.

3.4.1 Expérience 1

Les taux de reconnaissance obtenus avec les méthodes de reconnaissance basées sur les descripteurs LBP et LPQ pour les images artificiellement floutées avec un filtre gaussien Fg(s,11) sont présentés en **Figure 3.9** et **Figure 3.10** respectivement. Comme on peut le voir, le prétraitement que nous proposons améliore considérablement les performances quelle que soit la méthode de reconnaissance utilisée. Quand le flou croît, le taux de reconnaissance obtenu avec le descripteur LBP sans adaptation de la galerie aux images test diminue de 96,90% à 53,10% comme on peut le voir sur la **Figure 3.9**. Cela confirme que ce descripteur n'est pas du tout tolérant au flou. Lorsque la galerie de référence est adaptée à l'image d'entrée, seule une décroissance faible est observée. Combinée au descripteur LBP, notre approche permet d'obtenir un taux de reconnaissance ne diminuant pas en deçà de 96,15% ce qui est nettement supérieur à 53.10%. Quant à la méthode LPQ, non combinée à notre approche elle atteint des taux inférieurs à 60% alors qu'ils restent compris entre 97,32% et 96,15% avec adaptation de la galerie. La combinaison de notre approche avec la méthode basée LPQ a permis d'obtenir un taux quasiment constant compris entre 97,82% et 96,48% alors qu'il ne dépassait pas 85% pour les flous les plus forts avec la méthode utilisée sans adaptation de la galerie comme on peut le voir sur la **Figure 3.10**.

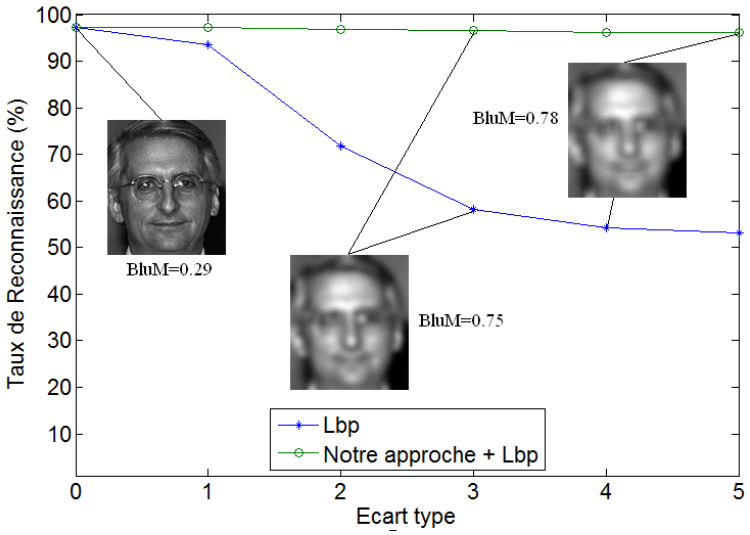

Figure 3.9 – *Taux de reconnaissance obtenus pour un filtre gaussien $Fg_{(s,11)}$ d'écart type s croissant (méthode de reconnaissance basée LBP).*

Expérience 2

Pour cette expérience, les images testées ont été floutées artificiellement avec un flou de bougé Fm(p,0), Fm(p,45), Fm(p,90) et Fm(p,135). Les résultats sont présentés sur les **Figure 3.11**, **Figure 3.12**, **Figure 3.13** et **Figure 3.14**. Comme dans l'expérience précédente, nous avons testé notre approche avec les méthodes d'identification basées sur

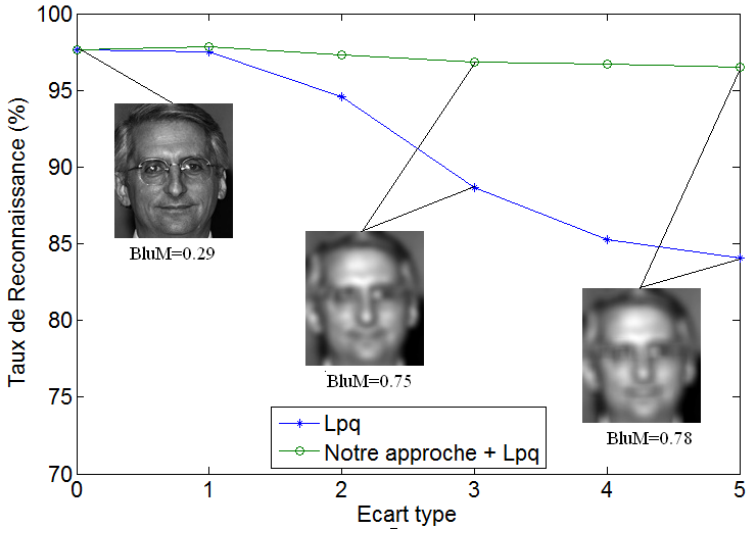

Figure 3.10 – *Taux de reconnaissance obtenus pour un filtre Gaussien $Fg_{(s,11)}$ d'écart type s croissant (méthode de reconnaissance basée LPQ).*

les descripteurs LBP et LPQ, des résultats similaires ont été obtenus. On observe en effet que combinée aux méthode de reconnaissance basées sur les descripteurs LBP ou LPQ, l'adaptation de la galerie à la qualité de l'image test augmente systématiquement le taux. Pour une orientation de 0° comme on peut le voir sur la **Figure 3.11**, le taux de reconnaissance obtenu avec le descripteur LBP diminue avec un flou de bougé d'intensité croissante pour atteindre 67,34% contre 85,26% avec le descripteur LPQ. Combinée à notre approche, ces méthodes permettent d'atteindre un taux de reconnaissance de 93,30% avec le descripteur LBP et de 93,13% avec le descripteur LPQ.

Par ailleurs, comme on peut le voir sur les **figures 3.13 et 3.14**, nous pouvons constater que pour les angles de 45° et de 135°, la diminution des performances de l'algorithme LPQ sans adaptation de la galerie à l'image test est moindre comparé à la diminution de ses performances pour un angle de bougé de 0° ou 90°. Il semble en effet beaucoup moins sensible aux effets de flou de bougé dans les directions de 45° et 135° pour lesquelles les taux d'ientification sont supérieurs. Cela est cohérent avec la valeur de l'indice de flou donné par le BluM. On peut également remarquer que globalement, l'intensité d'un flou de bougé semble moindre que celle d'un flou de mise au point pour les paramètres des filtres choisis.

Dans tous les cas, notre prétraitement permet d'améliorer les performance des algorithmes de reconnaissance. L'adaptation de la galerie à l'image test permet de trouver la gamme de fréquence qui contient l'information extraite par le descripteur commune à la galerie et à l'image test ce qui permet une bonne identification du visage.

Au final, ces résultats confirment l'efficacité de notre approche pour l'identification de personnes sur des images de visage floues, que le flou soit de type gaussien ou de type flou de bougé.

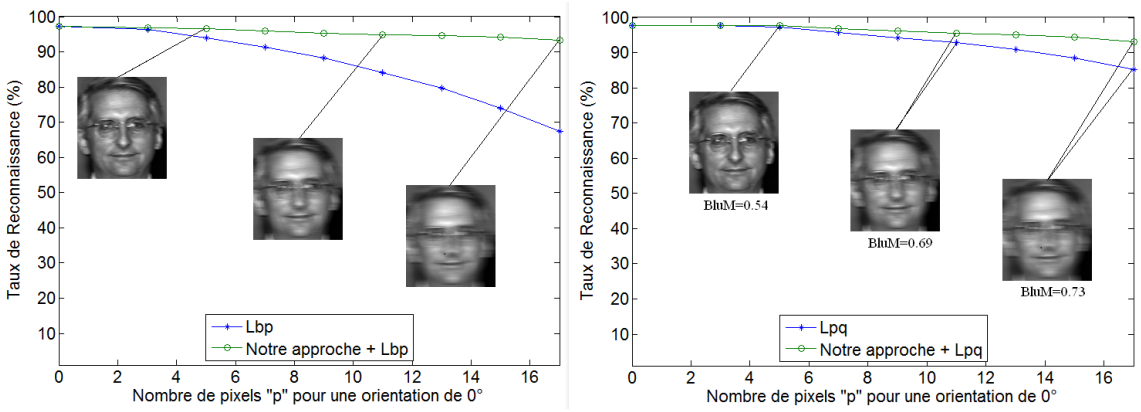

Figure 3.11 – *Taux de reconnaissance (méthode de reconnaissance basée LBP à gauche, LPQ à droite) obtenus pour des images dégradées par un filtre de bougé $Fm_{(p,0)}$ dont le nombre de pixels p est croissant.*

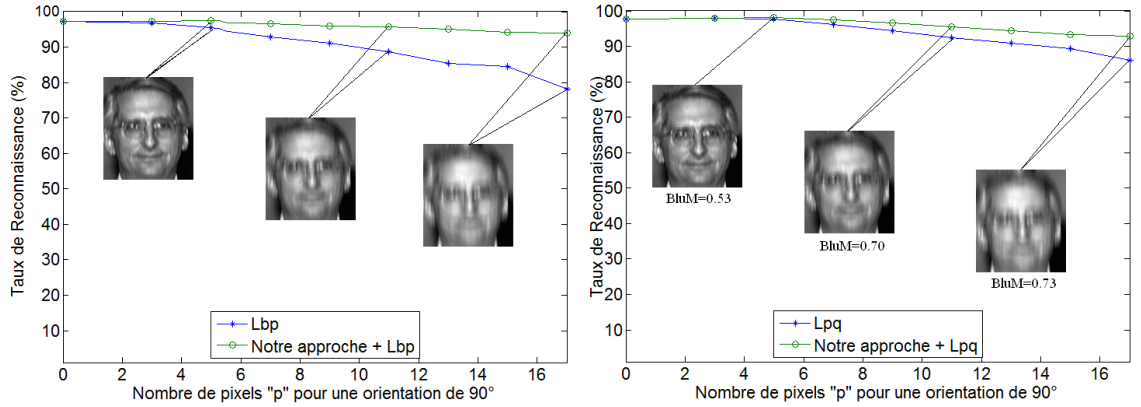

Figure 3.12 – *Taux de reconnaissance (méthode de reconnaissance basée LBP à gauche, LPQ à droite) obtenus pour des images dégradées par un filtre de bougé $Fm_{(p,90)}$ dont le nombre de pixels p est croissant.*

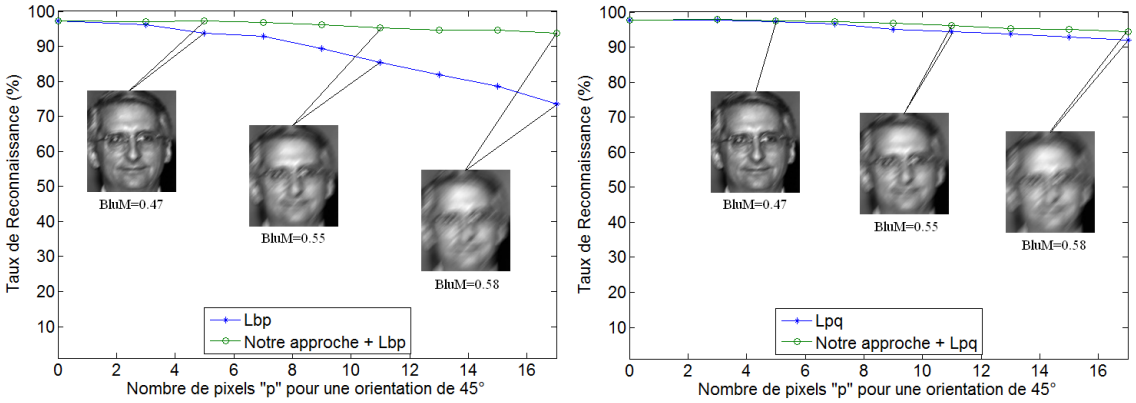

Figure 3.13 – *Taux de reconnaissance (méthode de reconnaissance basée LBP à gauche, LPQ à droite) obtenus pour des images dégradées par un filtre de bougé $Fm_{(p,45)}$ dont le nombre de pixels p est croissant.*

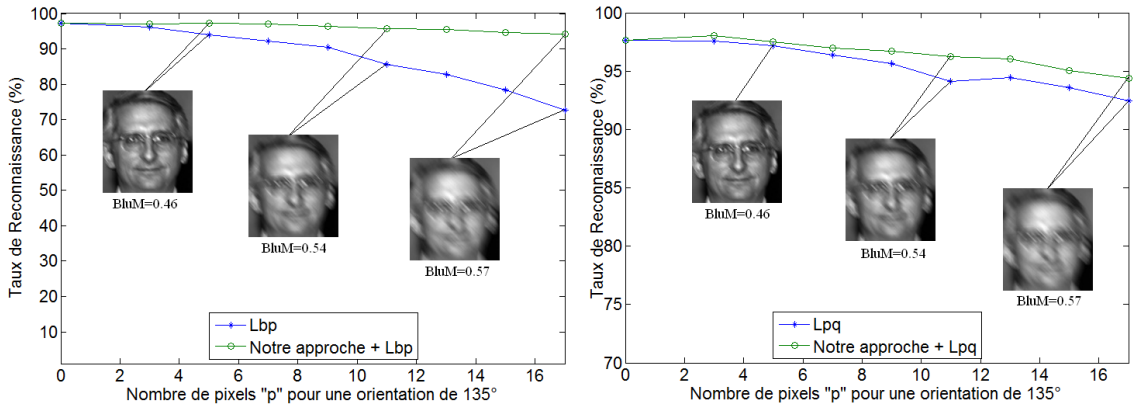

Figure 3.14 – *Taux de reconnaissance (méthode de reconnaissance basée LBP à gauche, LPQ à droite) obtenus pour des images dégradées par un filtre de bougé $Fm_{(p,135)}$ dont le nombre de pixels p est croissant.*

4. Méthode proposée pour une reconnaissance de visages sur des images avec artéfacts de bloc

Nous présentons dans ce paragraphe la méthode que nous avons développée pour améliorer les performances des algorithmes d'identification de visage lorsque les images d'entrée présentent des artéfacts de bloc. Autrement dit, on considère cette fois des images qui ont été compressées, voire fortement compressées, comme c'est le cas lorsque les images sont issues de caméras de vidéosurveillance. L'idée, similaire à celle proposée pour le flou, revient à trouver l'information utile pour la reconnaissance commune aux images de la galerie et à l'image test. Pour l'effet de bloc, tout comme pour le flou, il s'agit de trouver la gamme de fréquences qui contient les caractéristiques du visage nécessaires à sa reconnaissance. Pour cela, nous estimons dans un premier temps le niveau de l'artéfact de bloc présent dans l'image puis nous adaptons la galerie en fonction de ce niveau avant d'appliquer un algorithme d'identification de visage.

4.1 Choix des méthodes d'identification et base d'images utilisée

Pour chacune des expériences menées, nous utilisons, comme précédemment, les deux méthodes de reconnaissance basées sur le descripteur LBP et le descripteur LPQ. En effet, le descripteur LBP est largement utilisé par plusieurs méthodes de reconnaissance et constitue un des meilleurs descripteurs de l'état de l'art actuel. Il est donc intéressant de voir quelle est sa robustesse vis-à-vis de l'artéfact de bloc. LPQ a, quant à lui, été conçu pour permettre l'identification de visages dont les images présentent un niveau de flou relativement bas. On a ainsi pu voir dans la section 4 de ce chapitre que ses performances diminuaient lorsque le niveau de flou appliqué à l'image était fort. Néanmoins l'approche proposée a permis de palier ce problème tout en permettant d'obtenir de meilleures performances avec ce descripteur qu'avec le descripteur LBP. Il est donc intéressant, de connaître les performances de ce descripteur face aux artéfacts de bloc.

Pour constituer la galerie T_0, nous utilisons 200 images de l'ensemble fa de la base FERET. Pour la phase de test, nous utilisons 200 images de l'ensemble fb de la base FERET. Nous avons compressé la galerie d'origine T_0 avec plusieurs facteurs de qualité c de type *Jpeg* compris entre 1 et 90. Nous disposons exactement de 18 bases d'apprentissage T_c où $c \in \{1, 2, ..., 10, 20, 30, ..., 90\}$. La **Figure 3.15** présente un échantillon d'images de la base Feret qui ont été compressées avec une qualité de 5 et 10.

4.2 Méthode proposée et expériences

L'idée est d'appliquer la même stratégie que pour la gestion des images floues. On crée plusieurs galeries dégradées en appliquant une compression de niveau variable T_c. Nous obtenons ainsi plusieurs galeries compressées indexées par un coefficient de qualité c de type *Jpeg* compris entre 1 et 90, de la qualité la plus médiocre à la meilleure. A

l'issue de la phase de construction, les différentes galeries sont labellisées par une valeur du taux de compression appliqué. Dans un premier temps, nous démontrons l'amélioration des performances de reconnaissance de visage lorsque les images test sont comparées aux images de la galerie de taux de compression équivalent. Dans un second temps, nous proposons d'établir une correspondance entre la valeur du taux de compression et l'indice de bloc BLE décrit au chapitre 2. En effet, en conditions réelles d'utilisation, le taux de compression n'est pas connu.

4.2.1 Expérience 1 : mise en évidence de l'intérêt de la méthode proposée pour l'effet de blocs

Dans le chapitre 1, nous avons montré la dégradation des performances des deux méthodes de reconnaissance, LBP et LPQ, sur le taux d'identification des visages dont les images ont été fortement compressées. Dans cette première expérience nous souhaitons mettre en évidence l'intérêt de notre méthode d'adaptation de la galerie pour la reconnaissance de visages. Nous travaillons donc sur des images présentant des artéfacts de bloc connus.

Nous avons compressé les images test avec plusieurs niveaux de compression connus a priori. Puis, nous avons procédé à l'identification du visage test en adaptant la galerie en fonction du niveau de compression de cette image. Le vecteur de caractéristiques de l'image test a donc été comparé aux vecteurs de caractéristiques de la galerie compressée avec le même niveau de compression. Une image test compressée avec une qualité de 10 a été comparée à la galerie compressée avec cette même qualité (10). Les résultats sont présentés sur la **Figure 3.16**.

Comme nous pouvons le constater, pour les fortes compressions, les taux de reconnaissance sont meilleurs avec adaptation de la galerie que sans. Plus précisément, le descripteur LBP semble beaucoup plus sensible à cet artéfact que ne l'est le descripteur LPQ. Tous deux sont néanmoins sensibles à de très forts taux de compression. Nous commençons à voir une chute du taux de reconnaissance pour la méthode LBP à partir d'un coefficient c de qualité 20. A partir de cette qualité, sans adaptation de la galerie, les performances de la méthode LBP chutent considérablement passant de 97.5% (c=20) à 3% (c=1). En revanche, lorsque l'on adapte la galerie aux images test, le taux d'identification reste quasiment inchangé. Il est alors compris entre 98% (c=10) et 96% (c=1). En ce qui concerne la méthode LPQ, nous commençons à voir une chute du taux de reconnaissance à partir d'une qualité de compression de 10 passant de 98% à 91% pour une qualité de 1. Lorsque l'on adapte la galerie, on note une amélioration des performances pour les taux de compression élevés pour lesquels le taux de reconnaissance est compris entre 99.5% (c=10) et 96.5% (c=4). On note également une légère baisse du taux de reconnaissance pour les qualités de compression de 40 et 50 pour lesquelles le taux d'identification passe de 99.5% à 98.5% et de 99.5% à 99% respectivement.

Les résultats obtenus pour les qualités de compression les plus basses démontrent le bien-fondé de notre approche lorsqu'il s'agit de traiter le cas d'images dégradées par effets

c=100 *c=10* *c=5*

Figure 3.15 – *Image d'un visage de la base Feret compressé avec plusieurs qualités de compression jpeg. L'image originale est à gauche. Au centre, la même image compressée avec une qualité de 10 et à droite la même image compressée avec une qualité de 5.*

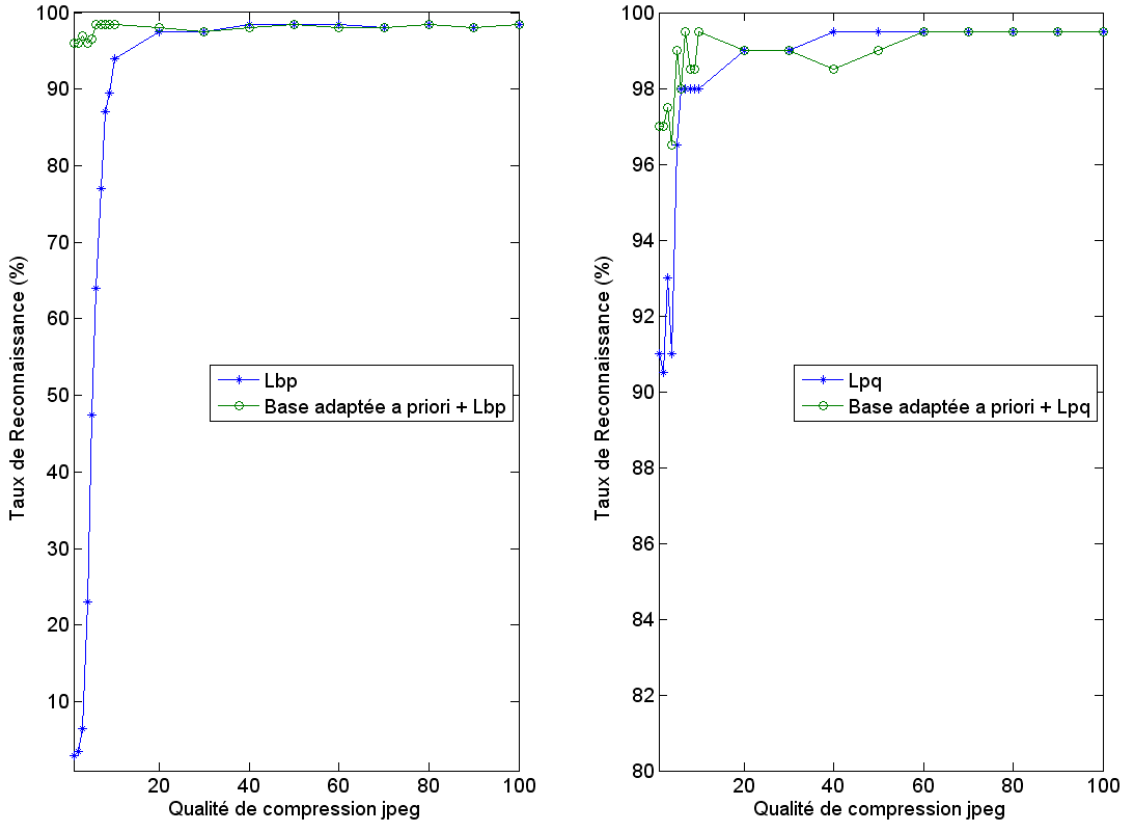

Figure 3.16 – *Expérience 1 : Résultats obtenus avec les deux méthodes d'identification basées sur les descripteurs LBP et LPQ avec et sans adaptation de la galerie lorsque le taux de compression de l'image test varie.*

de bloc. Mais cela nécessite de connaître a priori le taux de compression des images test. Or, mesurer ce taux de compression sur une image sans disposer de sa référence (ie. l'image sans compression) n'est pas possible. C'est pourquoi nous avons décidé d'essayer d'établir une correspondance entre le taux de compression et l'indice BLE qui permet d'évaluer la quantité de blocs sur une image.

4.2.2 Expérience 2 : Correspondance entre le taux de compression et l'indice de bloc BLE

Nous avons dans un premier temps estimé le niveau de bloc moyen de chacune des bases T_c pour $c \in \{1, 2, ..., 100\}$. Les résultats de cette estimation sont présentés dans le **Tableau 3.2**.

Table 3.2 – *Moyennes et variances obtenues pour la variable BLE en fonction du niveau de compression de type jpeg appliqué sur les images de la galerie.*

Moyennes et Variances de la variable BLE		
Compression *Jpeg*	Moyenne du BLE	Variance du BLE
1	8.61	32.06
2	8.58	32.35
3	10.32	37.36
4	13.01	33.26
5	14.28	27.65
6	14.53	21.40
7	14.66	16.76
8	14.73	12.84
9	14.83	12.32
10	13.9	12.34
20	11.93	6.48
30	10.33	5.22
40	9.73	5.18
50	8.9	5.33
60	8.42	4.45
70	7.80	5.1
80	7.03	4.02
90	5.83	2.99
100	4.73	2.55

Comme on peut le voir sur ce tableau, la moyenne de l'indice BLE augmente progressivement tout comme la variance de cet indice lorsque la qualité de compression diminue entre $c = 100$ et $c = 9$. Mais au delà de $c = 9$, l'indice BLE diminue alors qu'il devrait continuer à augmenter et sa variance devient de plus en plus importante. Cela est dû au fait que la métrique se base sur une mesure du nombre de blocs détectés dans l'image pour estimer le niveau des artéfacts présents. Lorsque la compression appliquée est à un niveau élevé, elle tend à lisser les contours de l'image et le nombre de blocs se stabilise ce qui explique pourquoi les coefficients BLE correspondant à une qualité comprise entre

5 et 9 n'augmentent plus. Lorsque la compression est très élevée, l'indice BLE diminue car il apparaît alors des blocs qui fusionnent avec leurs voisins créant de très gros blocs, mais dont le nombre diminue. L'algorithme confond alors ces larges zones homogènes avec des contours de l'image et ne les prend donc pas en compte dans l'estimation de la dégradation. Ceci est illustré sur la **Figure 3.17**. Ainsi, à des taux de compression élevés, l'indice BLE ne permet pas d'estimer la dégradation. Il est très performant pour estimer le nombre de blocs mais connaître ce paramètre n'est pas suffisant pour permettre une estimation fiable de la dégradation. A titre d'exemple, nous pouvons voir que pour une galerie dont le facteur de qualité est de 4, l'indice BLE moyen correspondant est autour de 9 comme c'est le cas des galeries dont le facteur de qualité est de 40 et 50. Un exemple de ce type d'estimation est également illustré sur la **Figure 3.17** où l'on peut voir que pour $c = 3$ l'indice BLE de 18 est inférieur à l'indice BLE trouvé pour $c = 6$.

Compte tenu de l'échelle de correspondance obtenue, nous décidons d'attribuer un indice BLE moyen à chacune des galeries pour $c \geq 10$. Par exemple, à la galerie compressée avec un facteur de qualité de 10, nous attribuons un indice BLE de 13.9. Nous faisons de même pour toutes les autres galeries.

Puis, nous procédons de nouveau à l'étape de reconnaissance de visages sans connaissance a priori du taux de compression des images test. Pour cela, nous estimons le niveau de blocs présents dans l'image test grâce à la métrique de bloc. L'indice BLE_t obtenu nous permet de sélectionner, via la table de correspondance **Tableau 3.2**, la galerie qui semble la mieux correspondre à l'image test. Pour cela nous considérons le BLE moyen qui a été attribué à chacune des galeries comme des seuils. La galerie dont l'indice BLE moyen se rapproche le plus de l'indice BLE_t est utilisée pour la reconnaissance. Les résultats de cette deuxième expérience sont présentés sur la **Figure 3.18**.

Comme nous pouvons le voir, les résultats ne correspondent pas à nos attentes. Il y a une dégradation des performances des deux méthodes de reconnaissance utilisées. En effet, pour un taux de compression faible ($c \geq 10$), le taux de reconnaissance est inférieur à celui obtenu lorsqu'il n'y a aucune adaptation de la galerie à l'image test. Nous notons néanmoins que les résultats obtenus avec la méthode LBP sont améliorés pour une qualité de compression située autour de $c = 10$. Pour ces images, nous tendons à nous rapprocher de la courbe idéale obtenue à l'expérience 1 pour laquelle nous connaissions le taux de compression a priori. Le taux d'estimation des images dont le BLE est supérieur à 13,9 a donc été amélioré. Pour la méthode LPQ, les résultats pour ces valeurs de qualité sont équivalents à ceux obtenus lorsqu'il n'y a pas d'adaptation de la galerie.

Il y a donc une « déficience » du BLE à fort taux de compression. En effet, comme il est indiqué dans le **Tableau 3.2**, l'indice BLE ne reflète pas correctement les dégradations de l'image et estime une qualité similaire à celle obtenue pour une qualité élevée. Il en résulte que certaines images très dégradées sont en fait considérées de la même manière que des images compressées à des niveaux de qualité voisins de 40 ou 50. Cela explique les taux de reconnaissance imparfaits pour ces coefficients. Pour avoir des résultats fiables, il

$c = 10$ $BLE = 13$ | $c = 6$ $BLE = 19$ | $c = 3$ $BLE = 18$

Figure 3.17 – *Évolution de l'effet de bloc lorsque le facteur de qualité diminue fortement. L'image a été compressée avec 3 coefficients Jpeg différents : 10, 6 et 3. Nous pouvons voir l'évolution du nombre de bloc et de l'indice BLE qui diminuent avec le facteur de qualité.*

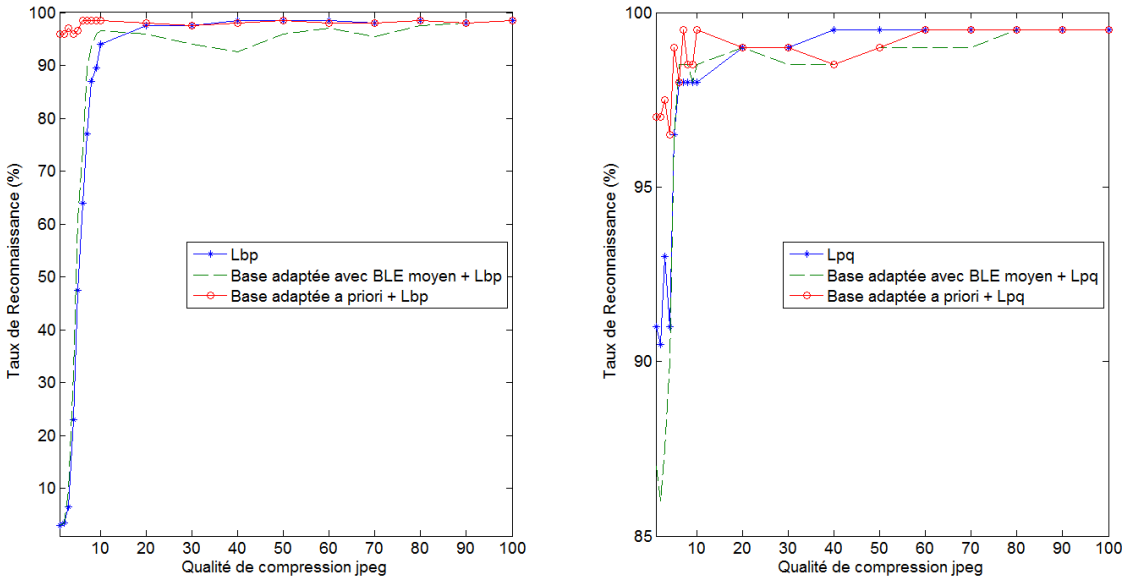

Figure 3.18 – *Expérience 2 : Résultats obtenus avec les deux méthodes d'identification basées sur les descripteurs Lbp et Lpq lorsque le taux de compression de l'image test varie et en adaptant la base d'apprentissage après avoir estimé, à l'aide de la métrique Ble, le niveau de bloc de l'image test d'entrée.*

est donc nécessaire de faire la distinction entre les images dont la compression est élevée ($c \leq 10$) et celles dont le coefficient de qualité est voisin de 40 ou 50.

4.2.3 Expérience 3 : différentiation des images peu compressées des images très compressées

Nous souhaitons dans cette expérience 3 classer les images test en fonction de la valeur du BLE selon 3 classes : compression *inexistante ou faible* ($classe_1$), compression *moyenne* ($classe_2$) et compression *très élevée* ($classe_3$).

Définition de la classe 2 : cette classe regroupe l'ensemble des images où il n'y a, a priori, pas d'incertitude importante sur la qualité de l'image. Ce sont donc l'ensemble des images qui ne sont ni très compressées, ni très peu, voire pas du tout compressées. Compte tenu de l'échelle de correspondance présentée sur le **Tableau 3.2**, nous définissons cette classe telle que :

$$Image \in \ Classe_2 \qquad \text{si} \quad Seuil_1 < Ble \leq Seuil_2$$

Avec $Seuil_1 = 10.3$ et $Seuil_2 = 13$ choisis pour prendre en compte l'incertitude de l'indice BLE. En effet, à partir d'une qualité de compression de 10, nous considérons les images comme étant fortement dégradées. La borne supérieure (ie. $Seuil_2$) doit donc être inférieure à 13.9. Pour les qualités de compression très basses, typiquement de 1 à 3, le niveau moyen de BLE est autour 9 comme c'est le cas pour les coefficients compris entre 40 et 60. Il faut donc choisir $Seuil_1$ supérieur à 10.3 qui correspond au BLE moyen obtenu pour $c = 30$.

Différentiation des classes 1 et 3 : D'un côté, nous avons des images très compressées et de l'autre des images peu compressées. Mais elles présentent des indices BLE similaires. Afin de les différencier, l'idée est d'appliquer une compression supplémentaire de qualité *Jpeg* connue et d'estimer la nouvelle valeur de l'indice BLE pour pouvoir le comparer avec celui de l'image d'entrée.

Soit une image pour laquelle on mesure un indice Ble : Ble_O. En théorie, le Ble_c de cette image après compression devrait être supérieur à Ble_O. En pratique, si l'image est déjà fortement compressée, Ble_C sera de valeur équivalente voire inférieure. La différence sera alors positive ou nulle. En revanche, si l'image d'entrée n'est pas compressée ou peu, le Ble_C associé sera supérieur à Ble_O. La différence sera alors négative.

Pour inférer le degré de dégradation de l'image d'entrée, on lui applique deux valeurs de compression moyenne : 30 et 50 ce qui conduit à deux images de qualité différente : I_{30} and I_{50} sur lesquelles nous appliquons la métrique de bloc et nous obtenons les coefficients Ble : Ble_{30} et Ble_{50}. Puis nous calculons la différence entre Ble_O et chacun des deux coefficients Ble obtenus :

$$Diff = Ble_0 - Ble_c \text{ où c } \in [30, 50] \ . \tag{3.14}$$

1. Si dans dans les deux cas (pour Ble_{30} et Ble_{50}) la différence est positive ou nulle alors la compression d'origine de l'image d'entrée est considérée comme étant très

forte et on associe l'image à la classe 3.

2. Si au contraire, dans les deux cas, la différence est négative ou nulle alors la compression d'origine de l'image d'entrée est considérée comme étant très faible et on associe l'image à la classe 1.

3. Si dans un des deux cas la différence est positive et dans l'autre elle est négative, alors la compression d'origine de l'image d'entrée est considérée comme étant très faible et on associe l'image à la classe 1 car pour des facteurs de qualité autour de 50, le taux d'identification n'est pas dégradé.

4.2.4 Expérience 4 : amélioration de la reconnaissance de visage par application d'une compression supplémentaire

Comme nous l'avons vu sur la **Figure 3.16**, lorsque le taux de compression de l'image test correspond à celui de la galerie, il permet d'améliorer les résultats de façon significative. Mais ce n'est plus le cas lorsque le taux de compression de l'image test n'est pas le même que celui de la galerie comme cela est illustré sur la **Figure 3.18**. En effet, plus la compression est élevée, plus on élimine des fréquences hautes et plus il manque d'informations dans l'image par rapport à l'image originale. Lorsque le taux de compression de l'image test est le même que celui de la galerie, l'information contenue dans la galerie est similaire à celle contenue dans l'image test. Autrement dit, cela revient à dire que les caractéristiques extraites par le descripteur sur l'image test et sur les images de la galerie ne dépendent plus des artéfacts de blocs mais seulement du visage ce qui explique les bons résultats obtenus même lorsque le taux de compression est élevé. C'est pourquoi, pour maîtriser le type d'information contenue sur l'image test par rapport à celle contenue sur les images de la galerie, nous proposons de compresser à nouveau l'image test d'entrée avec une qualité de compression c_s en fonction de la classe à laquelle elle appartient. Puis nous choisissons, pour la reconnaissance, la galerie ayant une qualité de compression c_s également.

Nous pouvons constater sur la **Figure 3.16** que le taux d'identification ne commence à diminuer notablement qu'à partir d'un facteur de qualité de 20. Les images de la classe 1 ne subissent donc aucune transformation et sont comparées, lors de la phase de reconnaissance, aux images de la galerie originale. Les images de la classe 2 correspondant à un niveau de compression moyen sont compressées avec un facteur de qualité de 30 pour traduire au mieux l'information contenue dans ces images sans en dégrader le contenu. Les images de la classe 3 sont quant à elles compressées avec un facteur de qualité de 5. Les images de la classe 2 sont alors comparées aux images de la galerie compressée avec un facteur de qualité de 30 et celles de la classe 3 sont comparées aux images de la galerie présentant un facteur de qualité de 5 ce qui correspond à une compression très forte.

$$Image \in \begin{cases} Classe_1 & \Rightarrow \text{ Pas de compression et galerie originale } T_0 \\ Classe_2 & \Rightarrow \text{ Compression de type } \textit{jpeg} \text{ de niveau 30 et galerie } T_{30} \\ Classe_3 & \Rightarrow \text{ Compression de type } \textit{jpeg} \text{ de niveau 5 et galerie } T_5 \end{cases}$$

Les résultats de cette dernière expérience sont présentés sur les **Figure 3.19** et **Figure 3.20**. Comme nous pouvons le voir sur la figure **Figure 3.19**, l'étape de classification permet de stabiliser les résultats. Pour les faibles taux de compression, nous nous rapprochons de la courbe obtenue lorsque le taux de compression est connu a priori. Néanmoins, nous observons une légère diminution des performances avec la méthode LBP pour une qualité de 50. Cela est certainement dû à un défaut de la classification. Certaines images appartenant à ce niveau sont classées dans la $Classe_2$ alors qu'elles devraient appartenir à la $Classe_1$ car leur BLE est proche de la limite inférieure ($Seuil_1 = 10.3$) de la classe 2. Elles sont donc comparées à la galerie de qualité 30 au lieu de celle de qualité 100. Il en est de même pour la méthode LPQ.

Pour les taux de compression élevés, comme nous pouvons le voir sur la **Figure 3.20**, notre prétraitement permet d'améliorer les performances de la méthode LBP. Nous obtenons un taux de 92.5% pour une qualité de compression de 6 contre seulement 64% lorsqu'il n'y a pas d'adaptation de la galerie. Les résultats sont plus mitigés pour la méthode LPQ où nous obtenons des résultats inférieurs à ceux obtenus sans notre prétraitement. Ainsi, même si l'ajout d'une étape de compression supplémentaire permet de se rapprocher de la courbe idéale, le problème de l'identification en cas de forts taux de compression demeure. Il convient donc d'analyser les résultats obtenus pour chacune des deux méthodes.

4.2.5 Analyses détaillées des résultats obtenus avec la méthode d'identification basée sur le descripteur LBP - Robustesse du descripteur LBP à l'artéfact de blocs

Le descripteur de texture LBP est basé sur des informations spatiales de l'image, en l'occurrence les valeurs des pixels voisins du pixel considéré. Les performances de l'algorithme diminuent en présence d'artéfacts de bloc lorsque la galerie n'est pas adaptée à l'image test et restent au contraire stables lorsque nous appliquons notre prétraitement. Il est donc intéressant de déterminer le degré de dégradation limite à partir duquel nous observons une dégradation des performances et les raisons pour lesquelles la correspondance entre le coefficient de qualité de compression et l'indice BLE est adaptée à cette méthode.

Comme on peut le voir sur la **Figure 3.19**, celui-ci se situe autour d'une qualité de compression de 20. Néanmoins, nous pouvons tout de même noter que ce descripteur est relativement robuste aux artéfacts de bloc puisque le taux d'identification passe de seulement 98.5% à 97.5% entre une qualité de 100 et une qualité de 20. Au delà, le taux diminue de façon très nette. Ainsi, le descripteur semble beaucoup plus sensible à cet artéfact lorsque la compression est élevée. Les effets de bloc qui apparaissent sur l'image

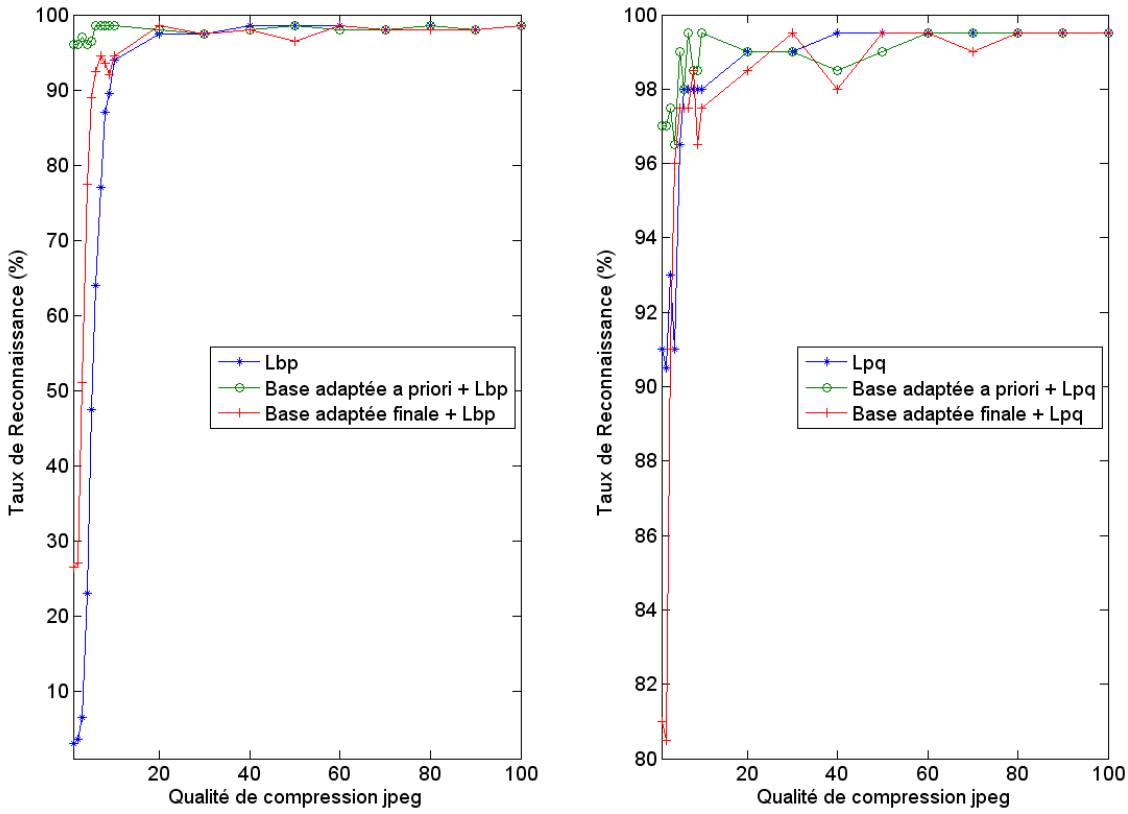

Figure 3.19 – *Expérience 4 : Intérêt de la classification (pour distinguer les images très compressées des images peu compressées) et de l'étape de compression pour l'amélioration des performances des algorithmes de reconnaissance.*

de façon beaucoup plus évidente à ce niveau, semblent modifier de façon importante le type de motifs recueillis au voisinage de chaque pixel et par conséquent l'histogramme des motifs uniformes s'en trouve complètement faussé. Le prétraitement que nous proposons permet donc une nette amélioration des performances de l'algorithme dans le cas où la compression appliquée est de mauvaise qualité. Nous obtenons par exemple un taux de reconnaissance de 94.5% pour une compression de qualité 7 contre 77% avec l'algorithme seul. Cela signifie que la compression introduit des artéfacts spécifiques dans l'image selon l'indice de qualité utilisé. Ces artéfacts modifient la forme des motifs recueillis au niveau de chaque pixel. Compresser une image avec un facteur de compression connue permet d'adapter la base d'apprentissage en conséquence et donc d'introduire les mêmes artéfacts et donc le même type d'information au niveau des motifs. C'est la raison pour laquelle appliquer une seconde compression de qualité c et comparer l'image obtenue avec la galerie de qualité c permet d'améliorer les performances.

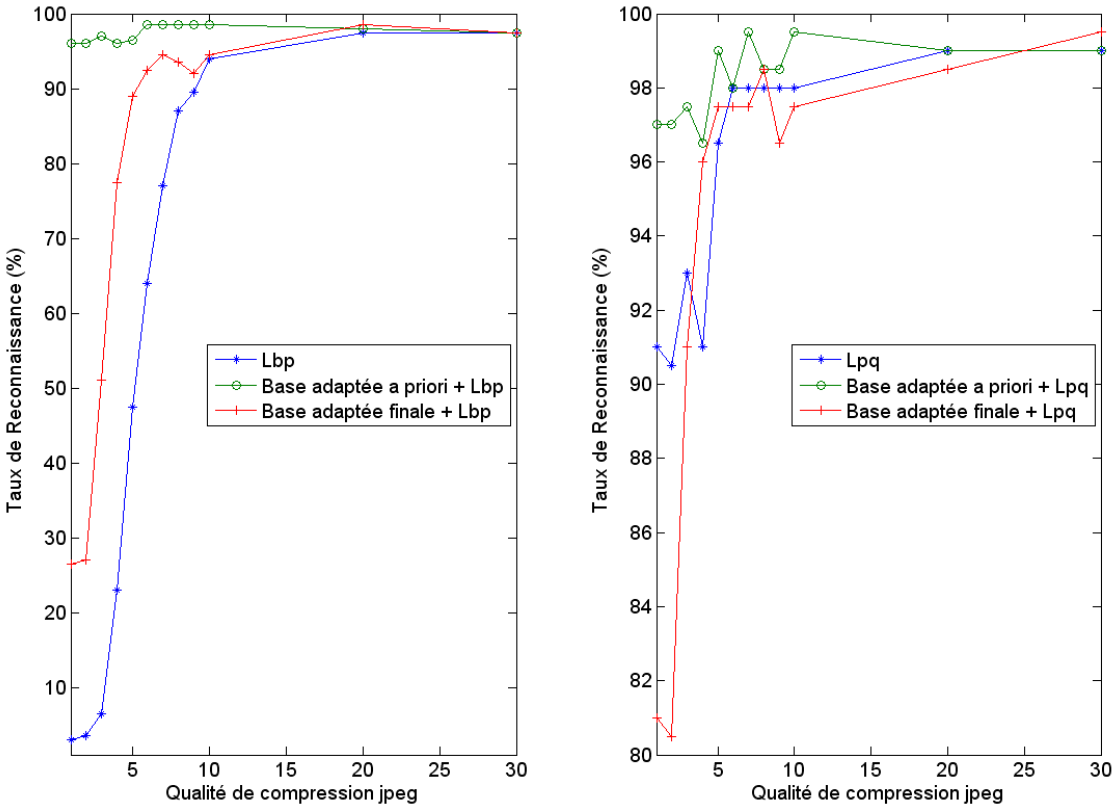

Figure 3.20 – *Expérience 4 : Intérêt de la classification (pour distinguer les images très compressées des images peu compressées) et de l'étape de compression pour l'amélioration des performances des algorithmes de reconnaissance. Zoom des résultats pour des qualités de compression comprises entre 1 et 30.*

4.2.6 Analyses détaillées des résultats obtenus avec la méthode d'identification basée sur le descripteur LPQ - Robustesse du descripteur LPQ à l'artéfact de blocs

Le descripteur de texture LPQ est basé sur des informations de type fréquentiel, il est également intéressant de connaître les limites de cet algorithme en présence d'effets de bloc. En effet, puisque la compression de type *Jpeg* consiste, lors de la phase de quantification, à éliminer les fréquences les plus hautes, le domaine de fréquence de l'image d'origine et celui de l'image compressée ne sont plus les mêmes.

Comme on peut le voir sur la **Figure 3.19**, les résultats sont à première vue surprenants. Étant donnée qu'une large gamme de fréquences est supprimée lors de l'étape de quantification et en particulier pour les compressions de basse qualité, on s'attendrait à ce que les performances de l'algorithme soient largement diminuées lorsque le taux de compression augmente. Or, comme on peut le voir sur la courbe correspondant aux taux

d'identification obtenus avec l'algorithme seul (sans adaptation de la galerie), il n'en est rien. Le taux d'identification reste constant jusqu'à une qualité de 10 puis diminue progressivement jusqu'à un taux de 91% pour une qualité de 1. Cet algorithme est donc extrêmement robuste aux artéfacts de blocs qui sont introduits dans les images. Dans le cadre d'une reconnaissance avec ce type de descripteur, cela signifie que l'étape de quantification ne commence à éliminer les fréquences indispensables à la reconnaissance d'un visage uniquement à partir d'une qualité de compression de 10. Il n'est donc pas étonnant qu'en adaptant la galerie les performances de cet algorithme augmente lorsque la qualité de compression est très basse comme nous l'avons vu lors de l'expérience 1. Après adaptation, l'information utile qui a été supprimée de l'image test a été également supprimée sur les images de la galerie. L'ensemble de l'information contenue dans les images de la galerie et celle de test est de nouveau commune. Néanmoins, lorsque nous appliquons notre prétraitement pour des images de qualité très faible, les performances de l'algorithme de reconnaissance diminuent, il est donc intéressant de comprendre pourquoi.

A partir de l'expérience 1, nous savons qu'appliquer exactement le même taux de compression de l'image test sur les images de la galerie permet d'améliorer les performances du descripteur LPQ. Or notre méthode ne permet pas d'améliorer le taux de reconnaissance des images pour une qualité $c \leq 5$ alors que nous les compressons avec une seconde compression de qualité 5 et que nous les comparons à la galerie qui a été compressée avec ce même coefficient. Comme on peut le voir sur la **3.20**, l'approche proposée fait diminuer le taux d'identification de 2 à 10 % en fonction de la qualité de compression appliquée sur les images lorsque celles-ci sont de basse qualité. Ceci s'explique par le fait que les images compressées avec un facteur de qualité $c \leq 5$ se sont vues supprimer encore plus d'informations utiles à la reconnaissance que celles dont la qualité est supérieure à 5. L'ajout d'une compression supplémentaire de 5 ne peut donc pas améliorer la reconnaissance. Mais pourquoi dans ce cas, le taux est-il meilleur lorsque les images fortement compressées sont comparées à la galerie originale (non compressées) ? Ceci s'explique par le fait que la galerie originale contient l'ensemble des informations utiles à la reconnaissance ce qui n'est pas le cas des images de la galerie compressées avec un facteur de 5. Autrement dit, cela signifie que selon la quantification appliquée et donc selon la qualité de la compression choisie, la gamme de fréquences nécessaire à la reconnaissance d'un visage change.

Conclusion

Dans certaines applications de vidéo surveillance, les images fournies par les caméras sont dégradées (flou, effet de blocs...) lors de l'acquisition. C'est pourquoi, dans ce chapitre, nous avons ciblé notre travail sur l'identification de visage lorsque les images sont de mauvaise qualité et nous avons proposé une approche permettant d'améliorer les performances de reconnaissance en cas d'images dégradées. Nous avons proposé dans ce chapitre une stratégie qui consiste dans chaque cas à adapter la galerie à la qualité de l'image test. L'originalité de notre approche repose sur le fait qu'elle est basée sur l'uti-

lisation de métrique sans référence, le BluM pour les artéfacts de flou et le BLE pour les artéfacts de bloc, qui fournissent l'information utile concernant la qualité de l'image d'entrée. L'efficacité de cette approche a été testée avec deux méthodes de reconnaissance différentes basées sur deux descripteurs très performants, LBP et LPQ, sur des images issues de la base FERET. Les visages sont comparés à un jeu de données pré-défini composé d'images de bonne qualité avec une seule image par personne en tant que référence. Les images à comparer ne présentant pas la même dégradation, les deux images ne possèdent pas la même bande de fréquences. Par conséquent, la difficulté est de trouver la gamme de fréquences commune aux deux vecteurs de caractéristiques de l'images et utile à la reconnaissance.

En ce qui concerne le flou, les résultats expérimentaux montrent clairement le bienfondé de notre approche pour différents types de flou quelque soit l'algorithme d'identification utilisé. Le BluM permet d'estimer avec précision l'intensité de la dégradation présente dans l'image d'entrée. Comme nous adaptons la galerie de référence en fonction de ce critère, cela revient à dire que la métrique permet de détecter les bandes de fréquences qui auront un intérêt au cours de l'étape d'identification et de supprimer celles qui seront inutiles lors du choix de la galerie.

L'artéfact de bloc altère l'identification d'un visage lorsque l'image de celui-ci présente un taux de compression relativement élevé. Nous avons montré que le principe de notre approche est validé mais cela requiert la connaissance a priori du taux de compression. Or, il n'existe pas à notre connaissance de métrique de qualité sans référence permettant d'estimer avec précision le taux de compression d'une image. C'est la raison pour laquelle l'utilisation de la métrique de bloc BLE était nécessaire. Néanmoins, celle-ci ne nous permet pas d'établir un lien entre la qualité de compression de l'image test et la quantité d'artéfacts de bloc pour tous les niveaux de compression. La correspondance entre la qualité de compression et l'indice BLE que nous avons proposée a permis d'améliorer néanmoins les performances de l'algorithme LBP basé sur l'information spatiale de l'image pour des qualités de compression très basse sans toutefois diminuer les performances lorsque la qualité est élevée. Les résultats obtenus avec notre prétraitement pour la méthode LPQ basée sur l'information fréquentielle de l'image sont plus mitigés même si le principe de notre approche a été validé pour cet algorithme également.

Dans ce chapitre nous nous sommes focalisés sur l'étape 2 de l'approche globale présentée en introduction. Nous avons ainsi montré l'intérêt d'une dégradation de la qualité des images de la galerie conformément aux degré des dégradations (liées au flou ou aux effets de bloc) estimées sur l'image test. Dans le chapitre suivant, nous considérons les perturbations liées à la pose. Plusieurs études ont déjà montré que si la galerie est composée d'images de visages ayant la même pose que l'image d'entrée, le taux d'identification est nettement amélioré par rapport au cas où il n'y a pas d'adaptation de la galerie à l'image test. Néanmoins, pour pouvoir choisir la galerie d'images dont la pose correspond à celle du visage d'entrée, il est nécessaire de pouvoir estimer ce paramètre. C'est la raison pour laquelle nous nous sommes cette fois-ci focalisés sur l'étape 1 et nous avons développé un nouvel estimateur de pose.

Chapitre 4

Estimation de la Pose appliquée à la Reconnaissance du visage

Introduction

La reconnaissance automatique de visages dans un environnement non contrôlé n'est pas une tâche aisée. Se voulant non intrusive et ne faisant d'hypothèse ni sur l'environnement dans lequel peuvent évoluer les sujets à reconnaître ni sur les sujets eux-même, ce type de méthodes se doit d'être le plus général possible. On parle ici de généralité dans le sens où dans un tel contexte, l'apparence des visages varie énormément, et ceci peut aussi bien être lié aux conditions dans lesquelles sont acquises les images qu'à la qualité de l'acquisition elle-même. Ainsi, cela suppose bien évidemment l'utilisation de caméras de vidéosurveillance dont les images sont, comme on l'a vu au chapitre 3, sujettes à de nombreux artéfacts principalement liés au flou ou à l'effet de bloc. Mais cela suppose également l'utilisation d'algorithmes capables de tolérer des variations d'apparence du visage liées aussi bien aux conditions d'illumination, qu'aux variations d'expression ou tout simplement à la pose du visage lui-même. De nombreux algorithmes performants ont déjà été proposés pour palier les problèmes d'éclairements [TT07], [VC09b], [ZCPR03] ainsi que ceux liés à l'expression du visage [ZCPR03]. En revanche, les problèmes liés à l'orientation du visage par rapport à la caméra restent le défi majeur des différentes techniques de reconnaissance actuelles [ZCPR03], [ZG09].

En effet, la reconnaissance d'un visage, quelque soit son orientation, est une tâche très difficile à réaliser car son apparence peut varier considérablement. En particulier,

dans un tel cas, l'amplitude des variations inter-sujet (entre deux images de personnes différentes) est souvent moins grande que celle existant entre deux images d'une même personne présentant des poses différentes. Or nous avons présenté brièvement dans le chapitre 1 plusieurs méthodes d'identification spécifiques à la reconnaissance des images de visages vus de face. Ceci explique pourquoi les taux de reconnaissance de ces algorithmes d'identification chutent considérablement lorsque les images d'entrée présentent de fortes variations au niveau de la pose. On peut par exemple citer la méthode proposée par Ahonen dans [AHP04] basée sur le descripteur de texture LBP [OPM02] qui ne tolère qu'un angle de rotation du visage inférieur à 15° [ZG09]. Au-delà, l'erreur d'alignement entre l'image test et celles présentes dans la galerie est trop importante.

Ainsi, depuis quelques années, de nombreux algorithmes spécifiques à la reconnaissance de visage dans un contexte de poses variables ont vu le jour. Et nombre d'entre eux utilisent l'information de pose comme paramètre d'entrée pour déterminer l'identité d'une personne. On définit l'estimation de la pose comme la capacité à inférer l'orientation de la tête d'une personne par rapport à la caméra qui le filme. Dans ce cas la pose doit être estimée avant l'étape de reconnaissance comme c'est le cas dans [PMS94] et [GMB02]. Il existe d'autres méthodes de reconnaissance faisant appel à ce paramètre. Plusieurs d'entre elles l'estiment simultanément à la reconnaissance du visage lui-même [BV03], [LGL00] ou séparément [MCT09]. Ainsi, l'estimation de la pose et la reconnaissance d'un visage dans ce contexte sont intimement liées. Mais l'utilité d'un estimateur de pose est multiple. La connaissance de ce paramètre permet également d'améliorer la détection des visages et de nombreuses études combinent donc détection et estimation [LFG+01]. L'étude de la pose d'une personne est également très utile pour déterminer la direction du regard d'une personne [MCT09].

L'estimation de la pose a donc un intérêt particulier pour de nombreuses applications. Nous nous proposons dans ce chapitre de présenter une nouvelle méthode d'estimation de la pose. Rappelons que notre approche consiste à estimer la dégradation présente dans l'image test puis à adapter la galerie en conséquence. Plusieurs travaux ont déjà montré le bien fondé de l'étape d'adaptation de la galerie pour la reconnaissance de visage lorsque la pose varie. Mais ces méthodes nécessitent généralement de connaître la pose du visage à reconnaître. Nous proposons donc un nouvel estimateur de pose afin de faciliter la reconnaissance sur des visages pris dans des poses variables. Nous introduisons dans un premier temps la problématique en énumérant les qualités que requiert un estimateur de pose dans un contexte de vidéosurveillance puis on présente brièvement l'état de l'art des différentes techniques d'estimation existant à ce jour avant d'introduire l'estimateur de pose que nous proposons et les résultats obtenus.

1. Intérêt d'un estimateur de pose pour la reconnaissance des visages

Nous distinguons de manière générale 3 approches différentes pour faire de la reconnaissance de visage lorsque la pose varie :

1. Restaurer l'image à traiter
2. Estimer une dégradation limite et ignorer les images trop dégradées
3. Traiter toutes les images dégradées ainsi que celles de bonne qualité

La première stratégie, la plus intuitive, consiste à essayer de supprimer les dégradations et donc à essayer de restaurer l'image à traiter. Pour la pose, cela signifie retrouver la vue de face correspondant à l'image de test dont on souhaite connaître l'identité. Une méthode consiste à créer un modèle 3D sur lequel est ajusté l'image de test. Cela permet, une fois l'ajustement réalisé, de disposer d'une vue de face de la personne à reconnaître. Il suffit pour la reconnaissance d'appliquer une méthode classique d'identification. Ceci est réalisé dans [BGPV05]. D'autres auteurs utilisent directement les paramètres de forme et de texture appris lors de l'ajustement pour la reconnaissance comme cela est fait dans [BRV02] et [RBV02]. Ce sont en effet deux façon de procéder similaires [BGPV05]. Néanmoins, dans chacun des cas, l'ajustement nécessite la connaissance a priori ou l'estimation du paramètre de pose de l'image de test.

La **seconde stratégie** pour gérer le problème de la pose consiste à déterminer l'angle critique à partir duquel la dégradation est telle qu'elle ne permet pas une identification correct d'un visage. Comme nous l'avons montré au chapitre 1, de nombreuses méthodes sont robustes à de faibles changements d'orientations du visage. En revanche, dès que l'angle de la pose devient trop important, les performances de ces algorithmes chutent considérablement. Dans le cas où nous disposons d'une vidéo sur laquelle nous souhaitons reconnaître un visage, il y a généralement plusieurs images de la personne à identifier. Lorsque la pose est au dessus d'un seuil critique, nous pouvons décider de l'éliminer et ne garder que les images de visages présentant de faibles poses. Néanmoins, cela nécessite d'une part d'estimer la pose sur chacune des images disponibles et d'autre part, cela ne règle pas le problème de l'identification lorsque tous les visages filmés présentent une large variation de la pose.

Une **troisième stratégie** pour faire de la reconnaissance lorsque la pose varie consiste à pouvoir traiter aussi bien les images de face que celles présentant des variations de la pose importantes.

Une des méthodes les plus simples mais assez intuitive est celle qui consiste à adapter la galerie en fonction de la pose du visage d'entrée. Il a en effet été montré que si l'on dispose de plusieurs images avec des poses différentes dans la galerie, les performances

des algorithmes sont améliorés [ZG09]. Les performances sont même similaires aux performances des algorithmes de reconnaissance utilisés pour la reconnaissance de visages pris de face [ZG09] lorsque l'on compare l'image test aux images de la galerie ayant la même pose que l'image test. Plusieurs auteurs ont utilisé cette méthode avec succès [Bey94]. Néanmoins, cette méthode ne peut être appliquée que si l'on dispose de plusieurs images par individus prises à différentes poses ce qui est rarement le cas. C'est la raison pour laquelle il existe plusieurs méthodes permettant d'interpoler les variations de pose d'un visage à partir d'une seule image par personne. En particulier des techniques 3D comme il est exposé dans [ZG09].

D'autres approches visent à modéliser les changements d'apparence du visage lorsque la pose varie en ne disposant que d'une seule image par sujet [ALC08], [PEWF08], [KY03] et [VC09a]. Le but de ces algorithmes est de pouvoir reconnaître un visage sans avoir à estimer la pose du sujet au préalable. Il s'agit donc de trouver l'information dans l'image qui permette de procéder à la reconnaissance tout en étant autant que possible indépendante de la pose. Les performances de ces méthodes sont présentées sur le **tableau 4.1**.

Table 4.1 – *Comparaison des performances obtenues pour plusieurs algorithmes d'identification de visages présentant des poses variables.*

Comparaison des taux d'identification selon l'angle de la pose			
Méthodes	Bases	Angles (°)	Taux (%)
Kanade et Yamada [KY03]	PIE (34)	45/67.5/90	100/80/40%
Prince et al. [PEWF08]	PIE (100)	16/62	100/91%
Prince et al. [PEWF08]	FERET (100)	22.5/67.5/90	100/99/92%
Vu et Caplier [VC10]	FERET (100)	25/40/60	100/94/83%

Bien que ces méthodes présentent généralement de très bons résultats dans l'état de l'art, elles montrent néanmoins de meilleurs résultats lorsque l'information sur la pose du visage d'entrée est connue [PEWF08], [KY03] et [VC09a].

Ainsi, l'information recueillie par l'ensemble de ces méthode améliore la reconnaissance mais ne peut être totalement indépendante de la pose. Autrement dit, nous pouvons raisonnablement penser qu'en combinant un estimateur de pose avec l'une ou l'autre de ces méthodes, nous obtiendrons de meilleurs résultats. Cela reviendrait à développer une méthode hybride qui permette à la fois d'extraire l'information dans l'image la plus pertinente pour la reconnaissance tout en ciblant le type d'information extrait en fonction de la pose du sujet.

Dans ce chapitre 4, nous proposons un nouvel estimateur de pose qui puisse s'appliquer à un contexte de vidéosurveillance afin d'améliorer les performances de la reconnaissance de visages lorsque la pose de celui-ci varie.

2. Caractéristiques de l'estimateur de pose

De nombreux algorithmes d'estimation de pose ont été proposés depuis plusieurs années. Mais tous ne peuvent s'appliquer à un contexte de vidéosurveillance. Il convient donc de définir dans un premier temps les caractéristiques que nous souhaitons imposer à notre estimateur de pose et les raisons qui justifient ces choix.

1. La résolution des images est très variable en contexte de vidéosurveillance et il peut arriver de devoir identifier des visages sur des images de très petite taille. Cela dépend en effet de la distance entre le sujet à identifier et la caméra. Il est alors difficile de réaliser une détection précise de certains traits caractéristiques du visage comme le réclament de nombreuses méthodes d'estimation de pose et cela est d'autant plus difficile que les images peuvent également être floues. L'algorithme d'estimation de pose doit donc pouvoir traiter des images de résolution différentes.
2. Compte-tenu de l'application et de la résolution des images, il est illusoire de vouloir développer un estimateur de pose au degré près. On traitera donc le problème d'estimation comme un problème de classification (discrétisation de l'ensemble des poses possibles). Cette approche est cohérente avec le choix de l'application de reconnaissance de visages étant donné que plusieurs algorithmes de reconnaissance sont tolérants à une variation de quelques degrés autour d'une valeur moyenne [ZG09].
3. Nous sommes dans un contexte où nous ne disposons que d'une seule image par personne dans la galerie. Autrement dit, toutes les méthodes nécessitant plusieurs images par sujet, au niveau de la galerie, ne peuvent s'appliquer à notre problématique. Nous ne considérerons donc pas les méthodes basées sur la vision stéréo qui permet de voir une personne à travers deux caméras (ou plus) de façon à pouvoir discriminer l'information de profondeur au niveau du visage.
4. Un estimateur de pose se doit d'être le plus général possible et donc le plus indépendant possible de l'identité d'une personne. Il doit pouvoir mettre en évidence les différences qui existent entre deux poses différentes tout en s'affranchissant de l'information identitaire.
5. Enfin, nous gardons en tête la contrainte de traitement temps réel associé à tout système de vidéosurveillance.

3. État de l'art de l'estimation de la pose

Il existe une excellente étude présentée dans [MCT09] qui regroupent l'ensemble des méthodes d'estimation de la pose depuis ces dernières années. L'article propose une catégorisation de toutes les méthodes existantes en 7 classes :

1. Appearance Template Methods

2. Detector Array Methods
3. Nonlinear Regression Methods
4. Flexible Models
5. Geometric Methods
6. Tracking Methods
7. Manifold Embedding Methods

Chacune de ces méthodes présente des avantages et des inconvénients mais toutes ne sont pas adaptées à nos contraintes. Nous détaillerons plus ou moins brièvement chacune de ces méthodes et nous expliquerons en quoi ces méthodes sont intéressantes ou non pour notre problématique.

3.1 Appearance Template Methods

Cette classe regroupe toutes les méthodes d'estimation de pose qui nécessitent plusieurs galeries, chacune associée à une pose donnée du visage. A priori, cette méthode n'est donc absolument pas adaptée à notre projet.

A titre d'exemple, une méthode simple, présentée dans [Bey94] consiste à extraire plusieurs caractéristiques d'un visage, typiquement les yeux, le nez, la bouche, et à s'en servir comme modèle pour une pose donnée. Autrement dit, il s'agit d'associer un jeu de caractéristiques à chacune des poses à estimer. Ces mêmes caractéristiques sont extraites du visage dont on souhaite estimer la pose puis comparées, l'une après l'autre, aux modèles préalablement formés par simple corrélation normalisée. La pose estimée correspond à la pose du modèle ayant obtenue le score le plus élevé. Cette méthode bien que très simple présente deux inconvénients majeurs vis-à-vis de notre problématique. D'une part elle nécessite d'avoir plusieurs images d'une même personne avec des poses différentes, ce qui n'est pas possible dans notre cas, et d'autre part elle est peu efficace car elle est fortement tributaire de l'identité des personnes présentent dans la base d'apprentissage. Or ce problème est un problème récurant dans le cadre de l'estimation de pose comme c'est le cas des méthodes présentées dans [NG02] et [NG99]. C'est pourquoi, il est intéressant de voir quelles solutions ont été proposées à ce problème pour être en mesure de développer un estimateur de pose robuste.

Deux études présentées dans [SGjO99] et [SGO00] proposent une solution qui consiste à utiliser des ondelettes de Gabor pour mettre en valeur la pose de chaque personne plutôt que son identité. Cette étude a ainsi montré que les filtres de Gabor, en fonction de leur orientation, sont capables de discriminer une pose plutôt qu'une autre. Pour cela les auteurs se basent sur le *rapport de similarité de pose* qui se veut représentatif des similarités invariantes à l'identité mais sensibles aux variations de pose.

$$r(\theta, \phi, f(.)) = \frac{d(\theta, \phi, f(.))}{d(\theta \pm \delta\theta, \phi \pm \delta\phi, f(.))}. \tag{4.1}$$

Cette mesure se fait après filtrage et calcul des distances entre images de visages présentant des poses similaires et des identités différentes et après filtrage et calcul des distances entre images présentant des poses et des identités différentes. $d(\theta, \phi, f(.))$ représente donc la distance moyenne entre les images ayant subi une transformation $f(.)$ et présentant des visages appartenant à des personnes différentes mais dont la pose est similaire. $d(\theta \pm \delta\theta, \phi \pm \delta\phi, f(.))$ représente la distance moyenne entre les images ayant subi une transformation f(.) et présentant des visages appartenant à des personnes différentes et des poses différentes également. $f(.)$ est la fonction de transformation (par un filtre d'ondelettes de Gabor par exemple) tandis que θ et ϕ représentent les angles selon lesquels le visage bouge de gauche à droite et de bas en haut respectivement. Ainsi, si les valeurs de ce rapport sont petites (inférieures à 1), cela signifie que $d(\theta \pm \delta\theta, \phi \pm \delta\phi, f(.))$ est grand et donc que les visages pour lesquels la pose est différente sont moins similaires que ceux présentant les mêmes poses. Par conséquent, lorsque le rapport est inférieur à 1, la transformation permet de mettre en évidence les similarités entre visages d'une même pose. Inversement lorsque le rapport est supérieure à 1. Comme on peut le voir sur la **Figure 4.1** et sur la **Figure 4.2**, certaines caractéristiques obtenues pour des orientations spécifiques des filtres de Gabor sont effectivement discriminantes vis-à-vis de la pose. Nous observons en effet que le rapport de similarités varie avec l'orientation des filtres et que la courbe contient des minimas assez bien définis.

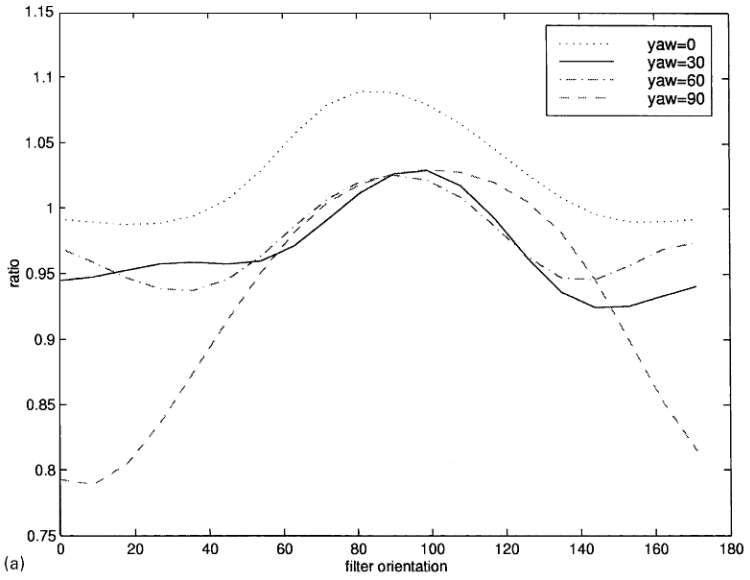

Figure 4.1 – *Valeur du rapport de similarité de pose en fonction de plusieurs orientations du filtre de Gabor appliqué à des images de visage dont la pose est fixe. 4 enregistrements différents ont été faits où seul l'angle de rotation de la tête autour de son axe (Yaw) varie tandis que l'angle représentant un mouvement de la tête de bas en haut (tilt) est fixe. Les courbes représentent le rapport de similarité de pose pour une rotation du visage de gauche à droite d'angle 0°, 30°, 60° ou 90°. [SGjO99].*

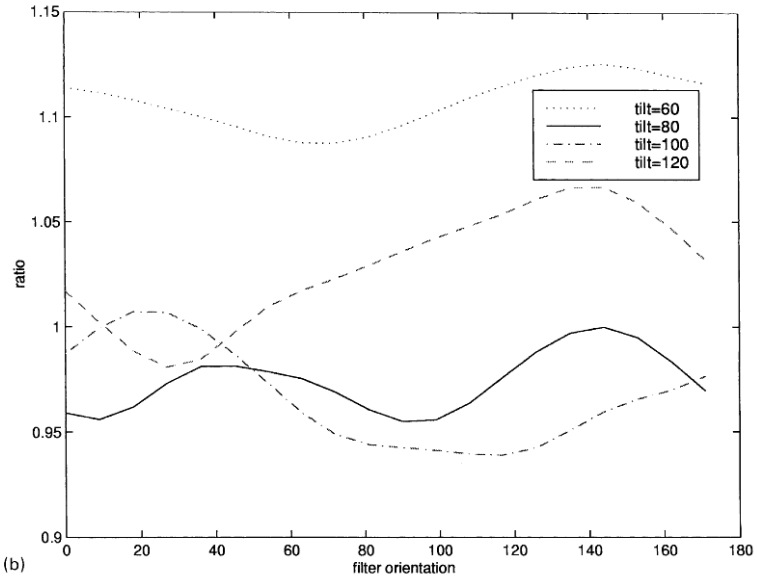

Figure 4.2 – *Valeur du rapport de similarité de pose en fonction de plusieurs orientations du filtre de Gabor appliqué à des images de visage dont la pose est fixe. 4 enregistrements différents ont été faits où seul l'angle de rotation de la tête traduisant un mouvement du bas vers le haut (tilt) varie tandis que l'angle représentant un mouvement de la tête de gauche à droite (yaw) est fixe. Les courbes représentent le rapport de similarité de pose pour une rotation du visage de bas en haut d'angle 60°, 80°, 100° ou 120°.[SGjO99].*

3.2 Detector array Methods

Ce sont toutes les méthodes utilisant une collection de détecteurs afin d'inférer la pose d'un visage d'entrée. Elles correspondent dans l'ensemble à des extensions de la méthode de détection de Rowley [RBK98] ou bien de Viola et Jones [VJ04] bien que certaines méthodes aient utilisé une collection de machines à vecteurs de support [HSW98]. Cela consiste donc à entrainer une cascade de détecteurs pour chacune des poses que l'on souhaite estimer ce qui signifie que tous ces détecteurs doivent être entraînés sur un certain nombre d'exemples négatifs afin de discriminer au mieux une pose d'une autre [ZHLH06]. La sortie de chacun d'entre eux est binaire : soit la pose en entrée est égale à la pose pour laquelle le détecteur est entraîné soit non. Bien que ce type de méthode soit très séduisant, il réclame un temps de calcul relativement long [MCT09] ce qui est rédhibitoire pour notre application.

3.3 Nonlinear Regression Methods

Cette classe regroupe l'ensemble des algorithmes d'estimation de pose utilisant une régression non linéaire pour inférer la pose d'un sujet. Ils ont essentiellement pour but de modéliser la pose sous forme de régression afin d'estimer l'orientation de la tête de façon précise. Or, comme on l'a expliqué au début de ce chapitre, il y a peu d'intérêt à vouloir estimer la pose de façon si précise. C'est pourquoi nous ne nous attarderons pas sur cette

classe.

3.4 Flexible Methods

Cette classe regroupe toutes les méthodes d'estimation de pose qui font appel à des modèles de visage formés à partir de graphe non rigide leur permettant de s'adapter à la géométrie de chaque visage étudié.

Une des méthodes les plus connues appartenant à cette catégorie est la méthode EBGM [WFKvdM99]. On rappelle que cette méthode consiste à faire correspondre un graphe d'une topologie donnée à un visage. Ce graphe est formé de nœuds, placés à des points caractéristiques du visage, et auxquels on associe un ensemble de jets représentant toutes les variabilités possibles de ce point. En prenant en compte les variabilités de chaque nœud et en moyennant sur un ensemble de graphes, les auteurs forment alors une représentation globale du visage très robuste à diverses variations qu'ils appellent Face Bunch Graph (ou FBG). A partir de cette représentation globale, les points caractéristiques d'un visage d'entrée sont automatiquement trouvés, une topologie propre au visage d'entrée est fixée et un jet est associé à chacun de ces nœuds pour former un graphe. La reconnaissance se fait par mesure de similarité entre le graphe de l'image test et l'ensemble des graphes de la galerie. Pour l'estimation de la pose le principe est le même mais légèrement simplifié. Il s'agit en effet de représenter la variabilité du visage liée à son orientation. Les auteurs ont proposé dans [KPM] de représenter un FBG pour chaque pose qu'ils souhaitent reconnaître. Ainsi, un visage d'entrée se voit attribuer d'autant de graphes qu'il y a de FBG possibles. Ces graphes dépendent aussi bien de la taille du visage que de sa pose. La topologie finale obtenue, pour une pose donnée, se fait par optimisation d'une fonction de similarité. Le graphe qui est obtenu avec la similarité la plus élevée détermine la pose du visage d'entrée. On peut voir une illustration de cette méthode sur la **Figure 4.3**.

Il existe d'autres méthodes dans l'état de l'art basées sur le même principe de fonctionnement. Plusieurs auteurs ont en effet développé un estimateur de pose basé sur des modèles actifs d'apparence (Active Appearance Model, AAM) [CET01] et sur des modèles actifs de forme (Active Shape Model ASM) [CTCG95]. Dans [LTC97], les auteurs présentent une méthode à partir de l'ASM. Cette méthode consiste à fabriquer un modèle général représentant les variations de la pose du visage. Dans une phase d'apprentissage, ce modèle est déformé de façon à s'adapter à plusieurs visages présentant des poses différentes. Des paramètres liés à la déformation de ce modèle sont appris en fonction de la pose du visage d'entrée. Lors de la phase d'estimation, le même modèle est déformé pour s'adapter au visage d'entrée et la pose est déterminée en fonction des paramètres de la déformation.

Ainsi, bien que ces méthodes présentent un intérêt particulier pour de nombreuses applications [MCT09], [TChZFZ06], elles ne sont pas pour autant adaptables en pratique à notre problématique. En effet, en fonction du nombre de poses que l'on souhaite déterminer, elles peuvent être très coûteuses en temps de calcul. Mais le problème majeur réside plus dans le principe de la méthode elle-même. En effet, pour fonctionner de façon

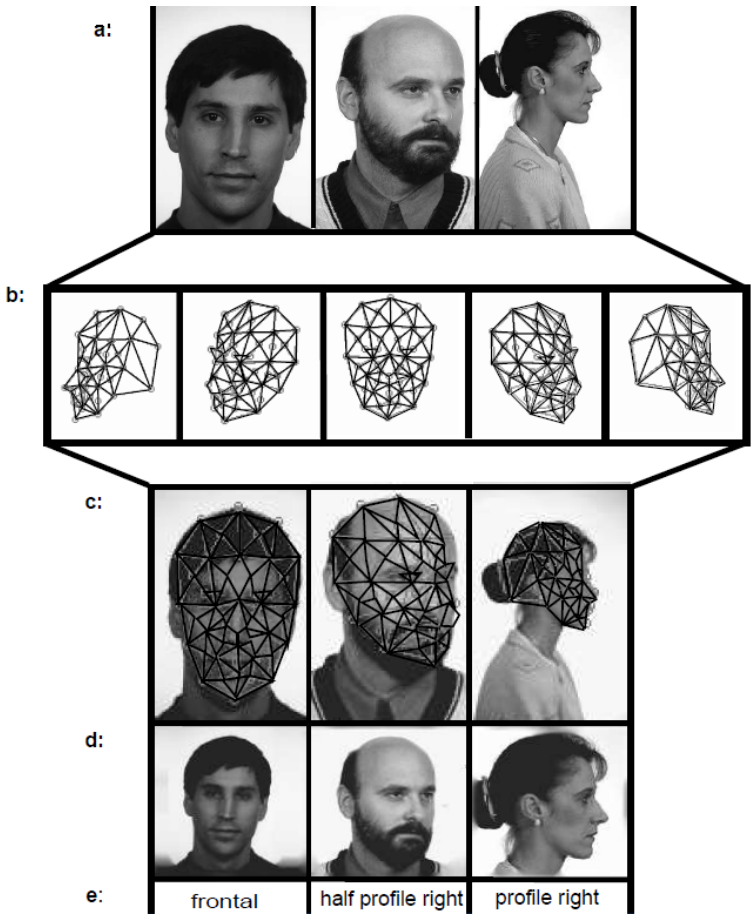

Figure 4.3 – *Ensemble des étape de l'estimateur de pose présenté dans l'article [KPM]. (a) : Images d'entrée du système présentant 5 poses différentes. (b) : Graphes associés aux 5 poses que les auteurs souhaitent pouvoir identifier de façon automatique. (c) : Les images d'entrée avec leur graphe associé. (d)&(e) : la sortie du sytème et la pose estimée des images d'entrée.*

optimale, ces méthodes nécessitent un nombre élevé de points caractéristiques qui représentent les nœuds du réseau. Or, tous ne sont pas forcément visibles, en particulier sur des images issues de caméras vidéosurveillance dont la résolution est souvent très basse. C'est pour cette raison que nous n'avons pas retenu ce type de méthode pour notre estimateur.

3.5 Geometric Methods

Les méthodes géométriques déterminent la position de certains points caractéristiques du visage tels que les yeux, le nez ou la bouche et déterminent la pose d'un individu en mesurant leurs position relatives qui varient lorsque la pose varie.

Il a en effet été démontré dans [WWLC00] que la perception de l'orientation d'un visage dépendait fortement de la déviation du nez par rapport à la verticale par exemple. La difficulté pour l'ensemble des méthodes géométriques repose donc sur la détection des points caractéristiques et de la déformation qu'ils subissent en fonction du mouvement de la tête du sujet [MCT09], [WS07], [HYD96]. Or, comme on l'a expliqué précédemment pour les méthodes de type EBGM, la qualité des images de vidéo-surveillance ne permet pas une extraction précise de ces caractéristiques sur un visage. Autrement dit, nous ne nous attarderons pas sur ce type de méthode dans la mesure où elles ne peuvent être utilisées dans notre contexte.

3.6 Tracking Methods

Les méthodes basées sur le suivi consistent à retrouver la pose de l'individu en se basant sur les changements de position de la tête entre chaque image consécutive d'une séquence vidéo.

Ce type de méthode utilisent donc à la fois l'information spatiale obtenue directement sur les images mais également l'information temporelle apportée par la vidéo. L'attention que peut porter un individu vers un endroit ou une scène en particulier peut servir d'indicateur pour détecter des lieux ou événements ponctuels qui méritent d'être surveillés. Ainsi, l'estimation de la pose peut permettre d'évaluer l'importance qu'a un lieu par rapport à un autre et de guider la caméra en conséquence [BR09]. Grâce à cela, de nombreux travaux récents ont pu être menés afin d'estimer l'orientation de la tête d'un individu de façon plus ou moins précise tout en considérant des images de qualité assez variable. Plusieurs méthodes ont été proposées afin de pouvoir estimer la pose d'un individu dans un environnement non contrôlé. Plusieurs méthodes proposent d'estimer l'orientation du visage sur des images de basse résolution [THT06] ou sur des images dont la distance caméra-sujet est grande [KCG11]. Chacune de ces méthodes utilise l'information temporelle ce qui nécessite d'avoir plusieurs images consécutives d'un même individu. Cependant, ces méthodes nécessitent généralement une initialisation de l'algorithme avec une pose connue [MCT09]. Dans [AKK] par exemple, les auteurs déterminent la pose par rapport aux déformations du visage en estimant les variations de distance avec les contours verticaux qu'ils comparent avec une pose de référence. Or cette référence doit être connue et cela oblige à disposer d'une image avec une pose de face. Ce n'est pourtant pas forcément

le cas, en particulier dans notre contexte de vidéosurveillance où les gens seront filmés furtivement dans un stade.

D'autres méthodes utilisent l'information de texture et de couleur (au niveau de la peau et des cheveux) [RR06] mais cela suppose de pouvoir distinguer les pixels appartenant au visage de ceux qui appartiennent au fond de l'image ce qui n'est pas toujours possible sur des images issues de caméras vidéo-surveillance [OGX09]. Pour s'affranchir de ce problème, certains auteurs recueillent donc l'information des pixels de l'image au niveau des 3 plans RGB et l'utilisent comme un nouvel extracteur de caractéristiques qu'ils combinent avec des machines à vecteurs de support [OGX09]. Les travaux menés avec ce type d'extracteur ont montré de bons résultats. Les tests ont cependant été faits sur seulement 8 classes différentes, prises entre 0° et 360°, chaque classe étant séparée par 45°. Pour pouvoir améliorer les taux d'identification des algorithmes de reconnaissance, un estimateur de pose doit pouvoir estimer la pose d'un visage avec un peu plus de précision. La tolérance des principales méthodes de reconnaissance étant bien inférieure à 45°. Enfin, il existe d'autres méthodes utilisant l'information issue de plusieurs caméras pour inférer la pose [SCK11] mais ce scénario n'existe généralement pas dans un contexte de vidéo-surveillance.

Dans le cadre de notre projet, nous n'avons pas pu disposer d'une vidéo permettant de simuler un scénario de vidéosurveillance. Nous avons donc préféré, dans un premier temps, nous concentrer sur des méthodes ne nécessitant pas l'aspect temporel de la vidéo.

3.7 Manifold Embedding Methods

Cette classe consiste à représenter les données (les images de visages de taille $n \times m$) dans un espace de plus petite dimension dans lequel il est possible de modéliser les variations de mouvements du visage. De cette façon, on peut estimer la pose d'un visage en projetant son image dans cet espace et inférer sa pose par correspondance avec le modèle.

Une des méthodes de réduction de dimension les plus connues est bien entendue l'analyse en composante principale. Dans [GGM+96], les auteurs proposent de discriminer la pose d'un individu en le projetant dans un espace de pose propre (Pose Eigen Space ou PES). Ainsi, une ACP est appliqué sur un ensemble d'images appartenant à une même personne et dont la pose varie. La **Figure 4.4** illustre cette représentation pour un ensemble de 60 images dont le visage présente une rotation allant de -90° à +90°. La **Figure 4.5** permet de comprendre quels types d'informations les premières composantes principales arrivent à mettre en évidence.

On voit en effet très bien sur le schéma que la première composante permet de discriminer les visages de profil dirigés vers la gauche de ceux dirigés vers la droite. Les deuxième et troisième composantes permettent quant à elles de discriminer les visages de profil de ceux pris de face. Néanmoins, cette représentation n'est pas suffisante pour permettre une classification précise des poses que l'on souhaite déterminer. Dans [SB02], les auteurs associent cette représentation à une pose donnée. Ils disposent donc d'autant de représentations que d'espaces à déterminer. Pour chacun de ces espaces, seules les cinq

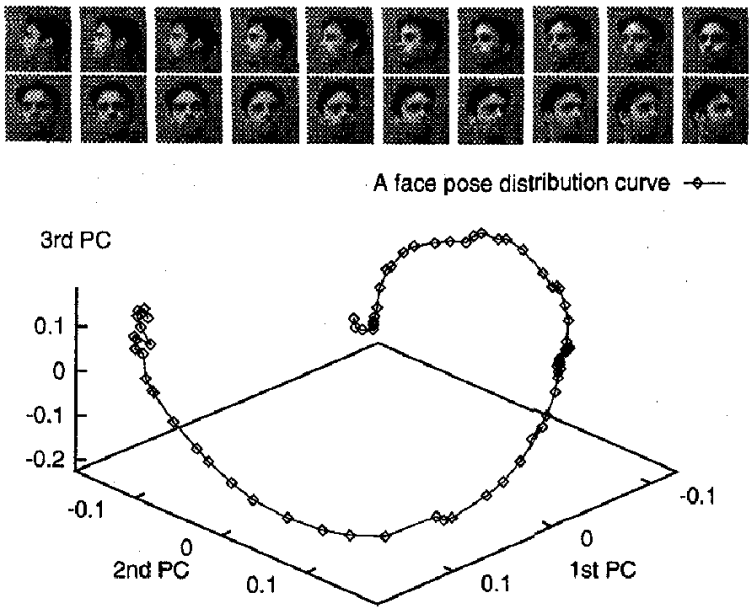

Figure 4.4 – *Représentation de la distribution de la pose du visage dans un espace à trois dimensions formé par les trois premières composantes principales de l'ACP. Pour cette représentation, les auteurs de l'article [GGM+96] ont utilisés 60 images d'une même personne mais présentant 60 poses différentes. Une vingtaine de ces images est représentée.*

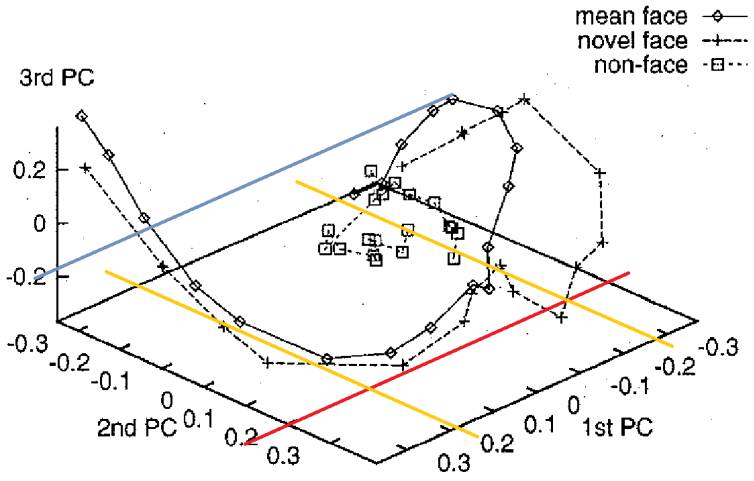

Figure 4.5 – *Représentation de la distribution de la pose du visage (PES), présentée dans [GGM+96] et dénommée par mean face sur le schéma. Comme on peut le voir la projection d'un ensemble d'images de visage dont la pose varie a une forme similaire au modèle PES. L'utilisation des trois premières composantes principales permet de différencier aussi bien les visages tournés vers la droite de ceux tournés vers la gauche (projection sur la première composante en jaune) que les visages de profil des visages de face (projection sur la seconde et la troisième composante en rouge et bleue respectivement).*

premières composantes sont gardées. Ensuite, l'image du visage dont on souhaite inférer la pose est projetée sur chaque espace. Pour chacun de ces espaces, la norme $||W||$ du vecteur des coefficients de projection est calculée. Elle correspond à la quantité d'énergie de l'image d'entrée qui est projetée sur l'espace considéré. Un de ces coefficients w_i est donné par la relation :

$$w_i = x^T.v_i \tag{4.2}$$

où v_i représente le $i^{ème}$ vecteur propre de l'espace considéré. Finalement, la pose de l'espace pour lequel la norme du vecteur $||W||$ est la plus grande est attribuée à l'image d'entrée. Pour affiner cette valeur, une interpolation linéaire peut être modélisée à partir de l'ensemble des courbes d'énergie associées à l'ensemble des poses ou bien uniquement à partir de l'une d'entre elles. Néanmoins, les images étant sujettes à de nombreux artéfacts, cela entraine une distribution des données complexe et surtout non linéaire. Autrement dit, l'ACP ne permet pas de modéliser la variabilité des données correctement.

D'autres méthodes ont donc vu le jour pour permettre une réduction de la dimension des données tout en se plaçant dans un cas non linéaire. Une des méthodes les plus connues consiste à utiliser dans un premier temps des fonctions noyaux (kernel) qui permettent de projeter les données dans un espace de plus grande dimension dans lequel elles seront linéairement séparables. Puis, dans un second temps, à appliquer une méthode de réduction de données comme l'ACP ou bien l'analyse discriminante linéaire (LDA). C'est le cas des méthodes présentées dans les articles [CZH+03] et [LFG+01]. Dans [LFG+01], les auteurs combinent une méthode non linéaire de réduction de dimension des données (KPCA) avec un ensemble de SVMs utilisés pour la classification. Similairement à [SB02], les auteurs souhaitent associer un espace à une pose donnée. Un jeu de données pour chaque pose leur permet d'apprendre les paramètres de ces espaces. Pour l'estimation de la pose, l'image test est projetée sur chacun de ces espaces de façon à obtenir un vecteur de caractéristiques qui lui est associé pour chaque pose. Ce vecteur de caractéristiques est ensuite présenté à l'entrée de chaque SVM dont la tâche est de discriminer une pose donnée de toutes les autres. Enfin, on infère la pose du visage d'entrée en fusionnant la réponse de chaque SVM, pour chaque vecteur de caractéristiques de l'image d'entrée. Dans [CZH+03] les auteurs présentent leurs résultats pour une variation de la pose entre -10° et +10° après avoir réalisé une séparation des données à partir d'une analyse discriminante linaire combinée à une fonction noyau. Mais les meilleurs résultats de l'état de l'art concernant les méthodes qui traitent l'estimation de pose comme un problème de classification ont été obtenus avec la méthode de Ma et al proposée dans [MZS+06] et utilisant l'extracteur de caractéristique LGBP introduit par Zhang et al dans [ZSG+05]. Cet extracteur de caractéristiques consiste à combiner un ensemble d'ondelettes de Gabor avec le descripteur LBP. Dans [MZS+06], cet extracteur de caractéristiques se veut représentatif d'une pose donné. A l'aide d'une réduction de donnée réalisée avec une PCA et d'un ensemble de SVMs, Ma et al ont amélioré de façon très significative les résultats dans ce domaine et présentent jusqu'à maintenant les meilleurs résultats dans l'état de l'art [ZG09].

D'autres techniques permettant une réduction des données de façon non linéaire ont également été appliquées à l'estimation de la pose. Certaines basées sur l'approche de ré-

duction de dimension non linaire de type Locally Linear Embedding (LLE) ou Laplacian Eigenmaps (LE) [RS00] permettent de prendre en compte la distribution locale des données. L'ensemble des voisins de chacune des données est défini. Le but de ces méthodes est alors de réduire la dimension de l'espace dans lequel doivent être exprimées les données tout en préservant les distances qui existaient entre un point et ses voisins dans l'espace d'origine. Les approches basées sur la méthode de réduction des données Isomap [RYS04], [TSL00] et [BYP07] considèrent quant à elles les données de façon plus globale et utilisent une distance géodésique entre voisins et non pas euclidienne. Ces méthodes présentent de très bons résultats dans l'état de l'art, mais ne permettent pas une projection aisée des images test dans l'espace ainsi formé. En effet, contrairement aux méthodes de réduction linéaire des données, elles ne possèdent pas de matrice de projection [BYP07] et nécessite donc une procédure particulière pour intégrer un nouvel échantillon dans l'espace de description des données ainsi formé [MCT09].

3.8 Hybrid Methods

Cette classe regroupe les méthodes combinant plusieurs des méthodes que nous venons de détailler pour s'affranchir des limitations inhérentes à chacune d'entre elles. La plupart de ces méthodes combinent les méthodes de tracking avec l'une ou l'autre des autres méthodes. Étant donné que nous n'avions pas à disposition d'une vidéo de notre projet, nous n'avons pas souhaité nous attarder sur ce type de méthode dans un premier temps.

3.9 Conclusion

Comme on a pu le voir à travers cet état de l'art, il existe une multitude de méthodes plus ou moins efficaces en fonction du contexte pour estimer la pose d'un individu. Chacune présente un intérêt et il convient de choisir celle qui peut satisfaire au mieux les exigences que réclame un estimateur de pose dans un contexte donné. Ainsi, dans le cadre de notre projet, la classe « Manifold Embedding Methods » semble la plus adaptée à nos contraintes. Elle présente de très bons résultats dans l'état de l'art, en particulier avec la méthode d'estimation de pose basée sur le descripteur LGBP par rapport aux méthodes considérant l'estimation de la pose comme un problème de classification, ce à quoi nous nous intéressons. Néanmoins, un des principal problème de cette méthode est lié à son implémentation longue et fastidieuse. Ce n'est pas un algorithme qui peut être appliquée dans un contexte de vidéosurveillance où la rapidité d'exécution est un point très important. C'est pourquoi nous avons porté notre attention sur cette méthode et nous avons cherché à l'améliorer. Dans la suite de ce chapitre, nous présentons l'estimateur de pose de Ma et al en détails et nous comparerons les résultats obtenus avec notre approche avec ceux obtenus avec cette méthode sur la base FERET [PMRR97].

4. Présentation de l'estimateur de pose proposé par Ma et al

Cette méthode d'estimation de pose utilise l'extracteur de caractéristique LGBP introduit par Zhang et al dans [ZSG+05] et présente jusqu'à maintenant les meilleurs résultats dans l'état de l'art par rapport aux méthodes d'estimation utilisant des valeurs discrètes (non continues) de la pose [ZG09]. Nous présentons les détails de cette méthode dans la suite de ce paragraphe ainsi que les avantages et les inconvénients qu'elle présente.

Cette méthode d'estimation de pose consiste d'une part à construire l'extracteur de caractéristiques LGBP qui consiste à combiner deux descripteurs utilisés pour la reconnaissance de visages que sont les ondelettes de Gabor et le descripteur LBP proposé par Ojala et al dans [OPM02]. D'autre part, elle consiste à effectuer une classification de ces vecteurs en fonction de la pose qu'ils représentent à l'aide d'une multitude de SVMs. Ainsi, pour procéder à l'estimation de la pose d'un visage test, le vecteur LGBP est calculé dans un premier temps puis un histogramme est formé à partir de ce vecteur avant d'être soumis à une classification par les SVMs.

4.1 Construction de l'extracteur de caractéristiques LGBP

Transformation de l'image d'entrée avec des ondelettes de Gabor

L'algorithme de Zhang et al extrait dans un premier temps les caractéristiques associées à la pose d'un visage par application de plusieurs filtres de Gabor. Il a été démontré à plusieurs reprises [SGjO99], [SGO00] [MZS+06] que ce type d'ondelettes permet d'extraire des caractéristiques liées à l'orientation de la pose du visage spécifiquement. On rappelle que la représentation en ondelettes de Gabor $G_{\mu,\nu}$ d'un visage pour une orientation μ et une échelle ν est obtenue par convolution de l'image de visage $I(x, y) = I(z)$ avec le filtre de Gabor correspondant telle que :

$$G_{\mu,\nu} = I(z) \star \psi_{\mu,\nu}(z) \tag{4.3}$$

Dans le cadre de cette méthode, l'image est représentée par un jeu de 40 ondelettes ayant 8 orientations $\mu = \{0, 1, ..., 7\}$ et 5 échelles $\nu = \{0, 1, ..., 4\}$ différentes. Soit $GMP_{\mu,\nu}$ le résultat de cette convolution après calcul de la valeur absolue.

Application du descripteur LBP sur chacune des images $GMP_{\mu,\nu}$

Le descripteur de texture LBP est appliqué sur chacune des 40 images $GMP_{\mu,\nu}$ ainsi obtenues. On rappelle que pour un pixel donné p dont le niveau de gris est n_p, l'indice

LBP qui lui est associé est :

$$LBP_{P,R}(n_p) = \sum_{v=1}^{P} S\{n_p - n_v\} 2^{v-1} \tag{4.4}$$

où n_v représente le niveaux de gris du pixels voisins v pris parmi les P voisins considérés dans un voisinage circulaire de rayon R autour du pixel p et où :

$$S\{n_p - n_v\} = \left\{ \begin{array}{ll} 1 & \text{Si } n_p \geq n_v \\ 0 & \text{Si } n_p < n_v \end{array} \right\} . \tag{4.5}$$

Une fois l'indice LBP calculé pour l'ensemble des pixels de l'image $GMP_{\mu,\nu}$, nous obtenons l'image labellisée $LG_{\mu,\nu}$.

Formation de l'histogramme LGBP

Chacune des images $LG_{\mu,\nu}$ est découpée en r régions. En prenant en compte l'ensemble des orientations et des échelles, nous obtenons le vecteur LGR formé des $40 \times r$ imagettes tel que : $LGR = \{LG_{0,0,0}, ..., LG_{\nu,\mu,i}, ..., LG_{4,7,r}\}$. Un histogramme $LGH_{\mu,\nu,i}$ est ensuite formé pour chacune de ces imagettes puis l'ensemble est ensuite concaténé en un vecteur global $LGBPH$ que l'on écrira par la suite LH pour simplifier l'écriture. Un schéma récapitulatif des 3 étapes principales que l'on vient de détailler est présenté sur la **Figure 4.6**.

4.2 Estimation de la pose d'un visage

Le vecteur LH ainsi obtenu constitue le descripteur associé à une image de visage. Celui-ci est calculé pour plusieurs visages et plusieurs poses différentes. L'ensemble de ces vecteurs permet, lors d'une phase d'apprentissage, d'apprendre les paramètres nécessaires à la mise en place d'une méthode de classification supervisée qui doit permettre une séparation des classes une fois les paramètres appris. Pour cela, les auteurs utilisent une classification à base de plusieurs SVMs en utilisant la méthode *une classe contre une autre*. Ainsi, une image d'entrée I_θ de pose inconnue θ dont le vecteur de caractéristique est LH_{I_θ} est classée selon la fonction de décision suivante :

$$y(I_\theta) = signe\{ \sum_{LH_{I_i} \in \mathcal{S}} \alpha_i y_i K(LH_{I_\theta}, LH_{I_i}) + w_0 \} \tag{4.6}$$

où $\mathcal{S}$ représente l'ensemble des vecteurs de support, K est une fonction noyaux choisie parmi celles que nous présenterons dans le paragraphe suivant consacré aux machines à vecteur de support, α_i est le $i^{ème}$ multiplicateur de Lagrange, $y_i = \pm 1$ et w_0 est un paramètre de l'hyperplan.

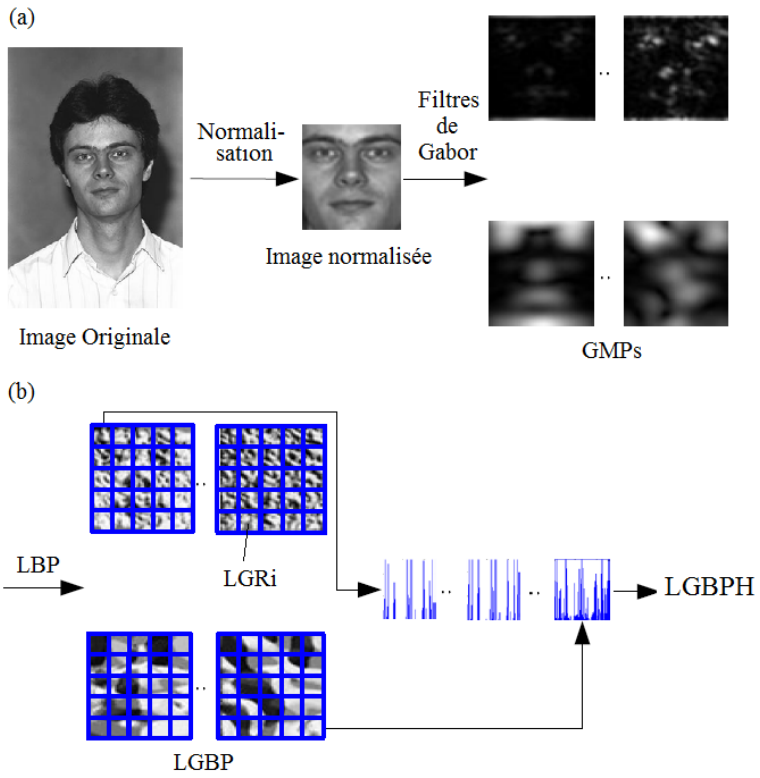

Figure 4.6 – *Descriptif des trois étapes principales de l'opérateur LGBP. (a) : Étapes de normalisation et de construction des images GMP obtenues après convolution des 40 filtres de Gabor avec l'image d'origine. (b) : Étapes de construction des imagettes après découpage en région des $GMPs$ et construction des histogramme à partir de l'extracteur de caractéristique LBP. [ZSG+05]*

5. Introduction aux Machines à Vecteurs de Support (SVMs)

Les machines à vecteurs de support sont des méthodes de classification par apprentissage supervisé. C'est-à-dire que le système apprend à classer des exemples connus dans des classes prédéfinies. Pour la pose, il s'agit de classer des images de visages selon leur orientation. La classification à partir d'un seul SVM est dite binaire car elle permet seulement de prédire l'appartenance ou non d'un exemple de données à une classe. Par exemple dans le cas de la pose, elle permet de dire si oui ou non un visage est d'une pose « p » donnée mais elle ne permet pas de définir cette pose.

Nous présentons la méthode dans le cas où les données sont linéairement séparables dans un premier temps puis dans le cas où la classification est non linéaire dans un second temps.

5.1 Classification linéaire par les SVMs

On se place dans la phase d'apprentissage du cas le plus simple qui soit, à savoir la séparation de données appartenant à deux classes différentes.

Définition des paramètres qui permettent de définir l'hyperplan séparateur

Nous cherchons à savoir si les données d'entrées X (dans notre cas des images de visages avec deux poses différentes) peuvent se séparer de façon optimale. Pour cela, on cherche une fonction f formée des échantillons d'entrée x et qui produit la sortie y telle que $y = f(x)$. Cette fonction représente l'hyperplan permettant de séparer de façon linéaire les deux classes. Cette fonction discriminante est alors de la forme :

$$y(\mathbf{x}) = \vec{w}^T \vec{x} + w_0 \tag{4.7}$$

et la classe est donnée par le signe de $y = f(x)$. Un vecteur d'entrée est donc assigné à la classe $+1$ si le signe de $f(x)$ est positive et à la classe -1 si le signe est négatif. Ainsi, pour séparer des données représentées dans un espace à D dimensions, l'hyperplan séparateur sera à $D - 1$ dimensions.

La marge γ qui représente la distance de l'hyperplan à l'observation la plus proche s'exprime comme suit :

$$\gamma = \frac{2}{||\vec{w}||} \tag{4.8}$$

Et les données satisfont l'équation $y_i(\vec{w}^T \vec{x} + w_0) \geq 1$.

Un schéma récapitulatif est donné sur **Figure 4.7**. Pour plus de précisions sur l'obtention de ces équations se référer à l'annexe D

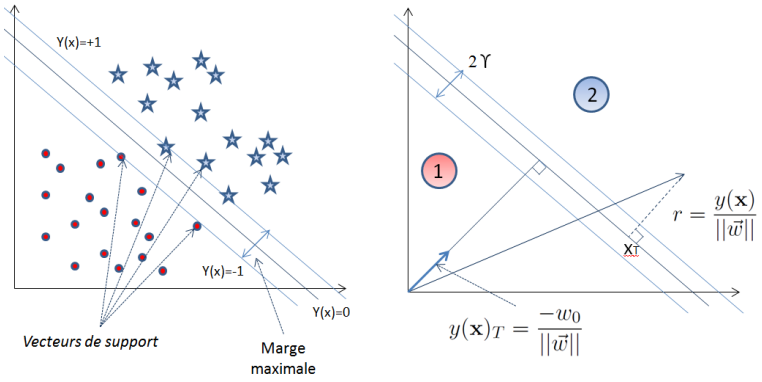

Figure 4.7 – *Représentation de l'hyperplan optimal et des variables qui lui sont associées.*

Maximisation de la marge γ

On souhaite maximiser la marge γ donnée par $\frac{2}{||\vec{w}||}$ ce qui revient à minimiser le terme $\frac{1}{2}||\vec{w}||^2$. Il s'agit donc de trouver les paramètres $\vec{w}$ et w_0 qui permettent de vérifier les conditions suivantes :

$$\left\{ \begin{array}{l} \text{Minimiser } \frac{1}{2}||\vec{w}||^2 \\ \text{tel que } y_i(\vec{w}^T\vec{x} + w_0) \geq 1 \ , \ \forall\{i = 1, ..., n\} \end{array} \right\} . \tag{4.9}$$

Or ce problème d'optimisation fait apparaître $p + 1$ paramètres ($\vec{w}$ et w_0) où p est la dimension des x_i qui sont les n variables d'entrée. Cette formulation du problème est appelée formulation primale. Pour simplifier le problème, on fait intervenir une fonction appelé Lagrangien qui permet d'énoncer la formulation duale du problème, après simplification, comme suit :

$$\left\{ \begin{array}{l} \max\limits_{\alpha}\{\sum\limits_{i=1}^{n} \alpha_i - \frac{1}{2}\sum\limits_{i,j=1}^{n} \alpha_i\alpha_j y_i y_j \vec{x_i}^T\vec{x_j}\} \\ \alpha_i \geq 0 \ , \ \forall\{i = 1, ..., n\} \\ \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \end{array} \right\} \tag{4.10}$$

où les α_i sont les multiplicateurs de Lagrange tous définis positifs. Leur intérêt principal est qu'ils permettent de ne considérer que les observations $\vec{x}_i$ appartenant aux deux hyperplans définis par $\vec{w}^T\vec{x} + w_0 = \pm 1$. En effet, tous les multiplicateurs de Lagrange correspondant aux données $\vec{x}_i$ qui n'appartiennent pas à ces hyperplans sont nuls. Les observations pour lesquelles ces multiplicateurs sont non nuls sont les vecteurs de support. Autrement dit, l'hyperplan séparateur optimum peut être déterminé grâce aux vecteurs de support seuls. Pour plus de précision se référer à l'annexe D.

Après simplification, nous arrivons à la forme duale du problème qui s'exprime comme suit :

et dont la solution est l'hyperplan de marge maximale $\frac{2}{||\vec{w}||}$ est :

$$\boxed{y(\vec{x}) = \sum_{\vec{x}_i \in \mathcal{S}} \alpha_i y_i \vec{x_i}^T\vec{x} + w_0} \tag{4.11}$$

$\mathcal{S}$ est l'ensemble des vecteurs de support.

5.2 Classification non linéaire par les SVMs

Il existe plusieurs méthodes permettant de s'affranchir du problème de non linéarité. Les variables dites, ressort, permettent de garder les équations permettant la résolution du problème telles qu'elles ont été formulées au paragraphe précédent tout en admettant une tolérance au niveau de la marge ainsi définie. Une seconde solution consiste à exprimer

les données dans un espace de plus grande dimension dans lequel les données peuvent être séparées de façon linéaire.

Utilisation des variables ressort

Les variables ressort ξ permettent de relâcher les contraintes au niveau de l'expression de la marge de façon à admettre des erreurs mais en les minimisant. Ainsi, le problème primal revient maintenant à minimiser l'équation :

$$\frac{1}{2}||\vec{w}||^2 + C\sum_{i=1}^{n}\zeta_i \tag{4.12}$$

où

$$y_i(\vec{w}^T\vec{x} + w_0) \geq 1 - \zeta_i \tag{4.13}$$

$C > 0$ est un coefficient permettant de réguler l'erreur admise et de rendre le système plus ou moins contraignant en fonction de la marge ainsi définie.

De la même manière que précédemment, on utilise la formulation duale pour résoudre le problème.

Transformation des données à l'aide de fonctions noyaux

Le principe de la méthode de classification non linéaire est présentée sur la **Figure 4.8**. Les variables qui ne sont pas linéairement séparables peuvent être exprimées à l'aide d'une fonction Φ qui permet de ré-exprimer ces variables dans un espace de plus grande dimension dans lequel ces mêmes variables seront séparables.

Nous cherchons donc, dans un premier temps, une fonction Φ qui à $\vec{x}$ associe $\Phi(\vec{x})$ telle que :

$$\mathbb{R}^p \rightarrow \mathbb{R}^q \text{ où p<q.} \tag{4.14}$$

$$\Phi : \vec{x} \rightarrow \Phi(\vec{x}) \tag{4.15}$$

Dès lors l'équation de la frontière de décision s'exprime comme suit :

$$y(\mathbf{x}) = \vec{w}^T\Phi(\vec{x}) + w_0 \tag{4.16}$$

et la résolution du problème dual est alors la suivante :

$$\boxed{y(\vec{x}) = \sum_{\vec{x_i}\in\mathcal{S}} \alpha_i y_i K(\vec{x_i}, \vec{x}) + w_0} \tag{4.17}$$

où $K(.,.)$ est une fonction noyaux telle que $K(\vec{x_i}, \vec{x_j}) = \Phi(\vec{x_i})^T\Phi(\vec{x_j})$. Il existe plusieurs fonctions noyaux régulièrement utilisées comme les fonctions polynomiale et les fonctions gaussienne.

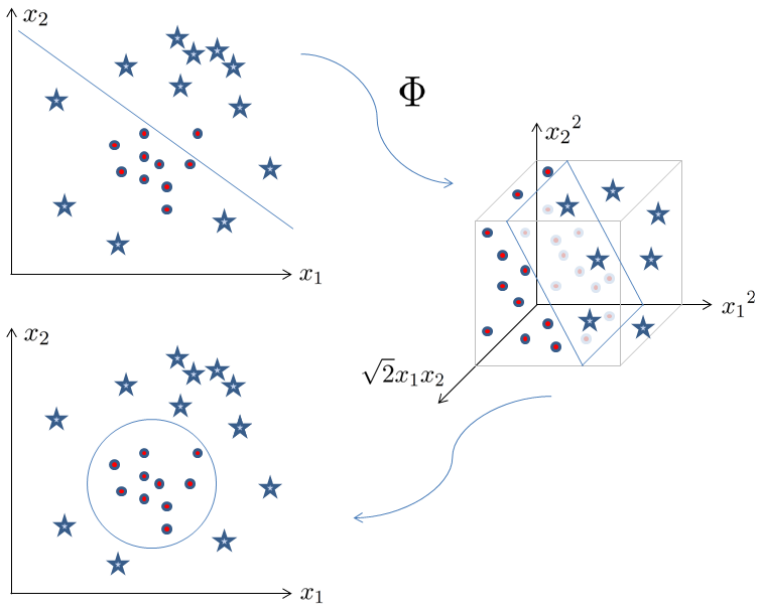

Figure 4.8 – *Principe de la classification non linéaire. Les données sont exprimés dans un nouvel espace à partir d'une fonction noyau ϕ (ici polynomiale de degré 2) pour permettre une séparation non linéaire des classes dans l'espace initial de description des données.*

Pour plus de précision sur la classification non linéaire à partir des SVMs, se référer à l'annexe D.

5.3 Application des SVMs aux cas de plus de deux classes

Il existe plusieurs type d'extensions mais nous nous contenterons d'en présenter seulement deux qui sont assez souvent utilisées. Dans la suite de ce paragraphe nous considérerons un problème à K classes avec $K > 2$.

Une classe contre toutes les autres (one versus the rest) Celle-ci consiste à considérer K hyperplans définis par : $y_k(\mathbf{x}) = \vec{w_k}^T\vec{x} + w_0$ où l'indice k fait référence au $k^{ème}$ hyperplan tel que si, $\forall k \in \{1, ..., K-1\}$, $y_k(\mathbf{x})$ est positif alors $\vec{x}$ appartient à la classe C_k et si $y_k(\mathbf{x})$ est négatif, $\vec{x}$ appartient à toutes les classes sauf C_k.

Une classe contre une autre (one versus one) C'est généralement cette dernière approche qui est la plus souvent utilisée dans les méthodes d'estimation de pose [MZS+06], [OGX09]. Pour ce type d'extension on considère K(K-1)/2 hyperplans où $y_{i,j}$ représente l'hyperplan permettant de séparer la classe C_i de la classe C_j. Par conséquent, $y_{i,j} = y_{j,i}$. Pour chacun des hyperplans, la règle de décision est la même : soit $\vec{x}$ appartient à la classe C_i, soit il appartient à la classe C_j. Lorsque tous les hyperplans ont été considérés, une classe peut avoir été représentée plusieurs fois. Ainsi, la décision finale consiste à compter le nombre

d'occurrence pour chacune des classes et à choisir celle qui a été la plus représentée. Néanmoins, il peut y avoir quelques ambiguïtés sur le résultat avec ce type de classification dans la mesure où deux classes peuvent avoir été représentées le même nombre de fois. La difficulté dans ce cas consiste à trouver une façon d'attribuer l'une ou l'autre de ces classes à la variable $\vec{x}$ en considérant, par exemple, les classes qui ont été un peu moins représentées.

6. Développement d'un estimateur de pose adapté à un contexte de vidéo-surveillance

Le travail que nous avons réalisé ici a pour but final de permettre l'intégration d'un estimateur de pose à un système de reconnaissance de visages dans un contexte de vidéo-surveillance.

Dans un premier temps, nous proposons un estimateur de pose qui se veut facilement applicable à un environnement non contrôlé. Pour cela, nous proposons une méthode simple à mettre en œuvre, robuste aux variations d'illumination et surtout rapide. Dans un second temps, nous proposons une extension de notre estimateur de pose, plus efficace en terme d'estimation mais beaucoup plus lourde en temps de calcul ce qui la rend peu attractive pour une utilisation dans un contexte de vidéo-surveillance.

Les deux estimateurs de pose développés abordent le problème d'estimation de la pose comme un problème de classification. Nous utilisons pour cela un ensemble de SVMs dont le nombre varie en fonction du nombre de classes que l'on souhaite reconnaître. Le premier estimateur extrait dans un premier temps les caractéristiques d'un visage en utilisant le descripteur POEM. Puis, il utilise ce vecteur comme vecteur d'entrée d'un ensemble de SVMs pour procéder à la classification et déterminer la pose du visage d'entrée. Le second estimateur de pose propose de développer un nouvel extracteur de caractéristiques basé sur la combinaison du descripteur POEM avec des ondelettes de Gabor. Le vecteur ainsi obtenu est ensuite utilisé comme vecteur d'entrée d'un ensemble de SVMs pour procéder à la classification.

Nous justifions le choix du descripteur POEM dans un premier temps avant de présenter les deux estimateurs de pose dans un second temps.

6.1 Choix du descripteur de caractéristiques POEM

Vu et al. [VC10] ont récemment développé une nouvelle méthode de reconnaissance en proposant un nouvel extracteur de caractéristiques qui combine l'information locale en recueillant l'orientation de contours au niveau de petites cellules avec l'information prise à plus grande échelle au niveau de régions formées au voisinage de ces cellules.

C'est la méthode que nous avons présentée en détails au chapitre 1. Ainsi, à l'inverse de l'extracteur de caractéristiques LGBP qui utilisent les ondelettes sur l'ensemble de l'image, le descripteur POEM calcule une orientation de gradient au niveau local, au sein de petite cellule de taille pré-définie. Puis, il recueille l'information à plus grande échelle en appliquant l'opérateur LBP au voisinage de ces cellules au niveau de larges blocs tandis que le descripteur LGBP applique quant à lui l'opérateur au niveau de petites régions de l'image de Gabor obtenue. Ainsi, les deux descripteurs sont capables d'extraire des caractéristiques sensibles à l'orientation à différentes échelles.

Le descripteur POEM ne nécessite pas la détection de points particuliers dans l'image et par conséquent, il peut être adapté à des images de différente résolution. Par ailleurs, il permet d'extraire l'information de contour au niveau des images par calcul de gradient. Par conséquent, il ne considère pas directement la valeur du niveau de gris des pixels de l'image ce qui le rend d'autant plus robuste aux variations d'illumination.

Et enfin et surtout, il permet d'extraire des caractéristiques sensibles à différente orientations de façon beaucoup moins complexe que ne le fait le descripteur LGBP. L'application de 40 ondelettes de Gabor sur l'image est en effet très lourde en temps de calcul. L'utilisation de gradients permet au descripteur POEM d'être à la fois robuste aux variations d'illumination [VC10] tout en étant d'une complexité faible. L'extraction de ce descripteur au niveau d'une image nécessite en effet 0.077 secondes seulement contre 0.616 secondes pour calculer les 40 convolutions nécessaires à l'implantation du descripteur LGBP.

6.2 Estimateur de pose proposé : version de base

Nous présentons dans un premier temps le principe de l'estimateur de pose proposé puis, dans un second temps, nous présentons les expérimentations qui ont été faites et enfin les résultats que nous avons obtenus.

6.2.1 Principe de l'estimateur de pose proposé

A l'image de la méthode proposée par Ma et al, le principe de notre méthode se divise en trois étapes : extraction des caractéristiques de l'image, réduction de la dimension des données et classification de ces vecteurs afin d'inférer la pose du visage.

Extraction des caractéristiques à l'aide du descripteur POEM

Comme on l'a vu au chapitre 2, le descripteur POEM est défini par :

$$POEM^{\theta_i}_{L,w,n}(p) = \sum_{j=1}^{n} Q\{S(\mathcal{G}^{\theta_i}_{p}, \mathcal{G}^{\theta_i}_{c_j})\}2^j \tag{4.18}$$

où $\mathcal{G}_p$ et $\mathcal{G}_{c_j}$ représentent respectivement l'amplitude des gradients du pixel central p et de ses voisins c_j dans le bloc pour une orientation θ_i donnée. $S(.,.)$ représente la fonction de similarité qui est par exemple, à l'instar de l'extracteur LBP, la différence entre deux

amplitudes de gradient. L et w sont les tailles des blocs et des cellules respectivement. n est le nombre de voisins du pixel p et enfin, Q est défini par :

$$Q\{x\} = \left\{ \begin{array}{ll} 1 & \text{Si } x \geq \tau \\ 0 & \text{Si } x < \tau \end{array} \right\} . \tag{4.19}$$

Et le descripteur $POEM$ final est obtenu après concaténation de tous les descripteurs $POEM^{\theta_i}$ calculés pour une orientation donnée :

$$POEM_{L,w,n}(p) = \{POEM^{\theta_1}, POEM^{\theta_2}, ..., POEM^{\theta_m}\} \tag{4.20}$$

où m représente le nombre d'orientations discrètes choisies.

Diminution de la taille des données avec une analyse en composantes principales

Comme on l'a vu dans l'état de l'art, un bon estimateur doit pouvoir mettre en évidence la pose d'une personne tout en ignorant l'information liée à son identité. Et l'analyse en composante principal permet justement de mettre en évidence ce type de caractéristique tout en diminuant la taille des données. Selon les paramètres du descripteur choisis, il convient de diminuer la taille de l'histogramme obtenu. C'est pourquoi avant d'appliquer une classification à base de machines à vecteur de support sur les données, chacune d'entre elles est projetée dans un espace à d dimensions. Les vecteurs propres de cet espace ont été construits au préalable lors d'une phase d'apprentissage à partir de plusieurs images présentant plusieurs poses différentes. L'ACP a donc été faite globalement pour l'ensemble des classes.

Étape de classification à l'aide des SVMs

Nous considérons maintenant que le descripteur *desc*, après application d'une ACP, constitue une représentation pertinente de l'information liée à la pose d'un visage. Mais ces vecteurs contiennent également de l'information liée à l'identité du visage pour lequel a été extrait le descripteur *desc*. Autrement dit, il s'agit maintenant de mettre en évidence l'information liée à la classe à laquelle appartient le visage (sa pose). Pour cela, il faut apprendre à séparer chacune de ces classes pour former un classificateur robuste capable d'inférer la pose d'un visage d'entrée. Nous utilisons pour cela un ensemble de SVMs avec la méthode « une classe contre une autre (one versus one) ». Ainsi, une image d'entrée I_θ de pose inconnue θ et de vecteur de caractéristiques $desc_{I_\theta}$, est classée selon la fonction de décision suivante :

$$y(I_\theta) = signe\{\sum_{desc_{I_i} \in \mathcal{S}} \alpha_i y_i K(desc_{I_\theta}, desc_{I_i}) + w_0\} \tag{4.21}$$

où $\mathcal{S}$ représente l'ensemble des vecteurs de support, K est une fonction noyaux, α_i est le $i^{ème}$ multiplicateur de Lagrange, $y_i = \pm 1$ et w_0 est un paramètre de l'hyperplan.

6.2.2 Expérimentation

Nous présentons dans ce paragraphe la base d'images choisie ainsi que la façon dont l'apprentissage des vecteurs propres de l'ACP et des vecteurs de support a été réalisé.

La base d'apprentissage

Les images de test et d'apprentissage sont issues de la base Feret. Celle-ci est formée de 9 poses différentes : [-60°,-40°, -25°,-15°,0°,+15°,+25°,+40°,+60°]. Pour chacune d'entres elles, nous disposons de 196 images. Soient 1764 images au total. Nous avons divisé chacune des bases en deux afin d'avoir suffisamment d'images pour l'apprentissage et pour le test. Nous avons donc 100 images par pose pour l'apprentissage et 96 images par pose pour le test. Aucune des images utilisées pour l'apprentissage n'a été utilisée pour le test.

Pour l'alignement des images, nous avons détecté manuellement les yeux et la bouche de chacun des visages. Les images ont été ensuite découpées de façon à ce que les yeux de l'ensemble des images soient alignés. Nous avons fixé la taille de chaque image à 64×64 pixels. Nous présentons un exemple d'images obtenues après alignement sur la **Figure 4.9**.

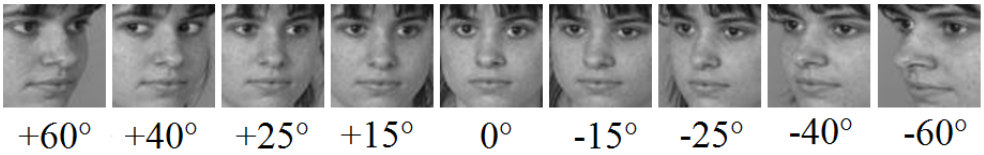

Figure 4.9 – *Présentation des images obtenues après alignement pour chacune des poses de la base FERET.*

Phase d'apprentissage pour l'ACP

Pour réaliser une ACP, nous avons besoin d'apprendre dans un premier temps les vecteurs propres qui constituent la nouvelle base dans laquelle on souhaite projeter les données. Pour cela nous avons utilisé l'ensemble des images d'apprentissage appartenant à k classes différentes. Comme nous disposons de 9 classes, $k = 9$ dans notre cas et donc $100 \times 9 = 900$ images sont utilisées pour l'apprentissage. Les 864 images test sont ensuite projetées sur l'espace adéquat quelque soit leur pose.

Phase d'apprentissage pour les SVMs

La méthode une classe contre une autre nécessite l'utilisation de $k(k-1)/2$ SVMs. Autrement dit, pour une classification à 9 classes il faut apprendre les paramètres de 36 SVMs notés SVM_{p_i,p_j} avec $p_j \neq p_i$ et $p_i, p_j \in \{1, 2, ..., k\}$. Il n'existe pas vraiment de méthode permettant un apprentissage rigoureux de ces paramètres. Néanmoins, plusieurs méthodes permettent de s'en approcher. L'optimisation consiste à faire varier les 3 principaux paramètres que sont :

1. les variables ressort ζ_i qui permettent de construire un hyperplan en admettant plus ou moins d'erreur.
2. le paramètre de régularisation C qui permet un compromis entre la marge et le nombre d'erreurs admissible. Une faible valeur de C entraine une faible tolérance.
3. les paramètres liés à la fonction noyaux choisie.

Chacun de ces paramètres doit être appris pour l'ensemble des SVMs, SVM_{p_i,p_j}, nécessaires à la résolution de notre problème à k classes. SVM_{p_i,p_j}, est utilisé pour résoudre un problème à deux classes seulement. Pour l'apprentissage des paramètres de SVM_{p_i,p_j}, nous avons besoin d'images des poses p_i et p_j données. Dans notre cas, nous disposons de 100 images d'apprentissage pour la pose p_i et de 100 images d'apprentissage pour la pose p_j. $\{p_i, p_j\} \in \{1, 2, ..., 9\}$.

Dans le cadre de cet apprentissage, nous avons choisi d'utiliser la méthode de validation croisée de type n-fold (ie. « n-fold cross validation »). Cette méthode consiste dans un premier temps à diviser le jeu d'images d'apprentissage défini précédemment de taille m en n jeux d'images disjoints S_i de taille m/n puis, à tester la classification sur chacun de ces jeux S_i. Dans notre cas, $m = 200$ puisque nous disposons de 100 images par pose. Si un jeu S_i est sélectionné pour le test, alors tous les autres jeux S_j, $j \neq i$ servent pour l'apprentissage. On effectue ce test pour tous les jeux S_i. On mesure l'erreur pour chacun et on fait une moyenne de l'ensemble de ces erreurs pour évaluer la classification avec un choix de paramètre donné. On recommence cette procédure pour des paramètres différents et on choisit les paramètres qui permettent de minimiser l'erreur moyenne. Il est bien évident que les images prises pour l'apprentissage doivent être différentes des images prises pour le test final. Pour notre méthode, aucune des images utilisées pour le test n'ont servi pour l'apprentissage. Étant donné le nombre d'images dont nous disposons pour l'apprentissage, nous avons décidé de prendre $n = 3$ pour éviter le phénomène de sur-apprentissage.

Nous avons optimisé l'apprentissage de chacun des SVMs en utilisant une fonction polynomiale. Pour chacun de ces SVMs, nous avons fait varier le degré de cette fonction polynomiale entre 1 et 3 et le paramètre de régularisation C entre 0.01 et 1 en fixant la valeur des variables ressort. Nous avons testé cette dernière valeur pour $\zeta_i = 10^-10$ et 10. Il est intéressant de noter que dans la majorité des cas, après optimisation des paramètres, le degré du polynôme retenu est de un. Autrement dit, l'ACP appliquée pour la réduction des données tend bien à les linéariser.

Choix des paramètres pour le descripteur POEM

Pour optimiser ce descripteur, nous devons dans un premier temps fixer les trois paramètres que sont :

1. le nombre d'orientations : en combien d'orientations est-il préférable de discrétiser l'orientation des gradients ?

2. la taille des cellules.
3. la taille des blocs

Nous avons choisi de fixer le nombre d'orientation à 3 comme il est préconisé dans [VC10]. Nous avons en revanche déterminé la taille des cellules et des blocs expérimentalement.

Pour cela, nous avons calculé un vecteur de caractéristiques basé sur le descripteur *POEM* pour chaque image d'apprentissage. Nous avons fait cela pour l'ensemble des poses représentées dans la base d'apprentissage. Puis, nous avons appliqué le descripteur POEM sur l'ensemble des images de la base test. Pour la classification, chaque vecteur de caractéristiques obtenu a été comparé aux vecteurs de la galerie toutes poses confondues. Nous avons donc mesuré les similarités entre le vecteur de caractéristiques associé à l'image test et les 900 vecteurs appris lors de l'apprentissage à l'aide d'un calcul de distance $\chi 2$. La pose associée au vecteur de caractéristiques pour laquelle la distance obtenue est la plus faible correspond à la pose de l'image test. Une fois la pose de chaque image test estimée, nous obtenons un taux d'estimation de la pose pour une taille de cellule et une taille de bloc donné. Les taux d'estimation les plus élevés ont été obtenus pour une taille de bloc de 3×3 et une taille de cellule de 7×7.

Finalement, nous avons fixé le nombre d'orientation à 3, la taille des cellules à 7×7 et la taille des blocs à 3×3.

6.2.3 Résultats

Nous présentons dans ce paragraphe l'ensemble des résultats obtenus à partir de l'estimateur de pose que nous proposons.

Nous souhaitons présenter dans cette section les résultats obtenus avec le descripteur POEM (la méthode que nous proposons) et le descripteur LGBP. Dans un premier temps, nous proposons de valider l'intérêt des SVMs pour la classification en comparant les résultats obtenus avec la méthode de base que nous proposons à ceux obtenus en utilisant une distance $\chi 2$. La méthode de classification utilisant une mesure de distance de type $\chi 2$ a été présenté lors du choix des paramètres d'entrée du descripteur POEM. Dans un deuxième temps, nous montrons l'intérêt d'utiliser POEM par rapport au descripteur LGBP.

Il ne serait en effet pas juste de dire que nous comparons nos résultats par rapport à la méthode proposée dans Ma et al. [MZS+06]. En effet, la méthode qui y est présentée a été testé sur une base de visages que nous ne disposons pas (la base CMU). Par conséquent, nous avons décidé de l'implanter nous même. Or les détails donnés par les auteurs ne sont pas suffisants pour implanter cet algorithme de la même manière. En particulier, il n'a pas été précisé la valeur des paramètres d'optimisation des SVMs ni la façon dont l'ACP a été réalisé. Il est en effet précisé dans la publication qu'elle est de la même taille que l'histogramme obtenu sur une région de l'image après application du descripteur LBP,

soit 256. Par conséquent, nous avons appliqué une ACP pour chaque région de l'image et concaténer l'ensemble des vecteurs obtenus pour former le descripteur final. Ceci n'est pas absurde dans la mesure où l'histogramme obtenu pour une région est indépendant des régions voisines. Dans toute la suite de cette section, les résultats présentés ont été testé sur le même ensemble de SVMs. Par conséquent, seul le descripteur est modifié. Nous faisons référence au descripteur proposée dans [MZS+06] par le terme « LGBP ».

Intérêt des SVMs pour la classification

Nous avons reporté les résultats obtenus avec les deux méthodes de classification (mesure de distance $\chi 2$ et SVMs) sur le **tableau 4.2**. Les taux d'estimation reportés dans ce tableau correspondent au pourcentage d'images ayant été correctement classées parmi l'ensemble des images de la base de test (à savoir 864 images au total).

Table 4.2 – *Comparaison des résultats obtenus pour les deux modes de classification : mesure de distance $\chi 2$ et SVM.*

Taux d'estimation de la pose obtenu sur la base FERET (864 images).	
Méthode	Taux d'estimation
$\chi 2$	70.95%
SVM	83.33%

Comme on pouvait s'y attendre, les performances de l'estimateur proposé à partir des SVMs sont nettement meilleures. Autrement dit, l'utilisation de la méthode de classification basée sur les SVMs permet de mieux distinguer les caractéristiques d'une classe par rapport à une autre. Mieux que ne le fait une simple mesure de distance $\chi 2$. Nous obtenons en effet un taux d'estimation de 84.45% avec les SVMs contre 70.95% avec une mesure $\chi 2$. Dans les **Tableaux 4.3** et **4.4**, nous avons reporté le taux d'estimation obtenu par classe, c'est à dire le pourcentage d'images, par classe, dont la pose a été correctement estimée avec une mesure de distance $\chi 2$ et des SVMs respectivement.

Table 4.3 – *Taux de bonne classification avec l'estimateur de pose basé sur l'extracteur de caractéristiques POEM avec une classification de type $\chi 2$.*

Taux d'estimation obtenu pour une pose donnée (%) $\chi 2$									
-60	-40	-25	-15	0	15	25	40	60	Taux
89.58	61.46	67.71	62.5	64.58	59.38	65.63	79.17	88.54	70.95

Table 4.4 – *Taux de bonne classification avec l'estimateur de pose basé sur l'extracteur de caractéristiques POEM avec une classification basée sur les SVMs.*

Taux d'estimation obtenu pour une pose donnée (%) $SVMs$									
-60	-40	-25	-15	0	15	25	40	60	Taux
90.63	78.13	84.38	78.13	84.38	83.33	76.04	83.33	91.67	83.33

Comme nous pouvons le voir sur ces deux tableaux, l'utilisation des SVMs permet d'améliorer très nettement la classification, en particulier au niveau des poses concentrées

autour de zéro souvent difficile à distinguer, même à l'œil nu. En effet, les taux obtenus pour les poses de $\pm 15°$ et $\pm 25°$ avec une mesure de distance $\chi 2$ sont beaucoup plus bas que ceux obtenus en utilisant les SVMs. Nous obtenons par exemple un taux de bonne classification, pour un angle de 15°, de 59.38% avec une mesure de distance $\chi 2$ contre 83.33% avec les SVMs. Nous avons reporté la répartition des angles estimés pour une pose réelle de l'image test, par l'algorithme utilisant une mesure de distance $\chi 2$ sur la figures **Figure 4.10** et par celui utilisant les SVMs sur **Figure 4.11**.

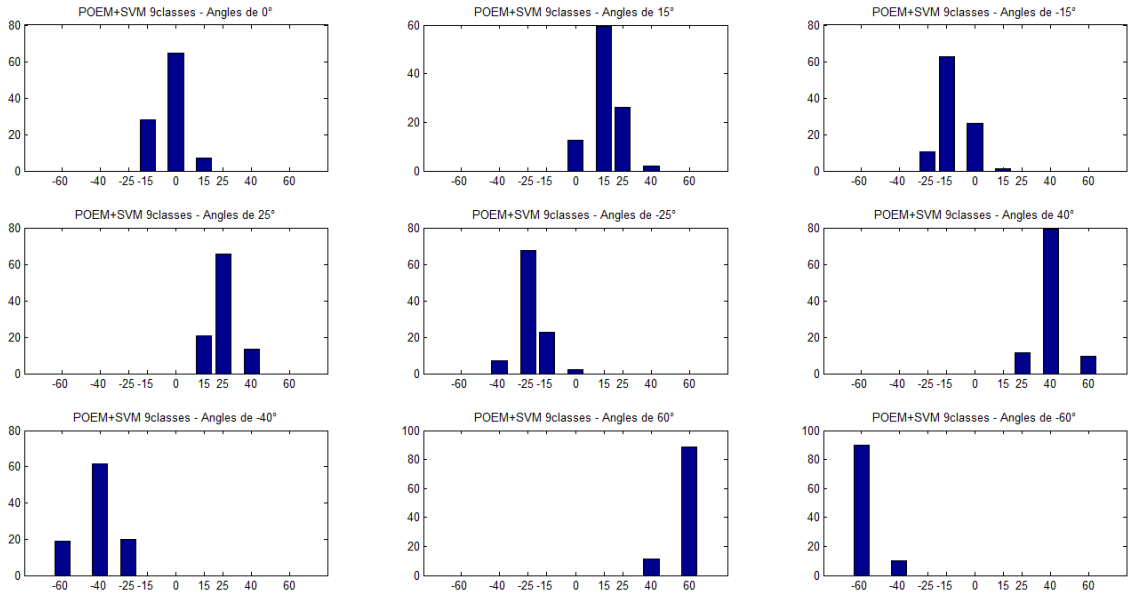

Figure 4.10 – *Répartition des poses estimées dans le cas où nous utilisons une mesure de distance $\chi 2$. 9 classes ont été considérées. L'angle réel de la pose du visage test d'entré est donné au dessus de chacun des 9 graphes représentés.*

Comme nous pouvons le constater, la répartition des erreurs est meilleure dans le cas où nous utilisons les SVMs mais dans les deux cas, elle se fait essentiellement autour de la pose réelle du visage que nous souhaitons estimer. Nous pouvons voir que la répartition est la plus importante pour les angles de $\pm 15°$. Pour une pose réelle de 15°, les erreurs sont concentrées sur les poses de 0°, 25° et 40°. L'écart étant seulement de 10° entre les angles de 15° et 25° et de 15° entre les angles de 0° et 15°, nous obtenons une répartition d'erreur non homogène autour de la pose réelle de 15°. Nous pouvons faire le même constat pour la répartition des erreurs autour d'un angle de 25°. Néanmoins, cette répartition d'erreur est moindre comme nous l'avons vu sur les **Tableaux 4.3** et **4.4** pour l'algorithme utilisant les SVMs.

Nous venons de montrer l'intérêt des SVMs pour la classification. Nous reportons maintenant sur le **Tableaux 4.5** le pourcentage d'images ayant été correctement classées parmi l'ensemble des images de la base de test (à savoir 864 images au total) pour la méthode de base que nous proposons en utilisant le descripteur POEM et celle où nous

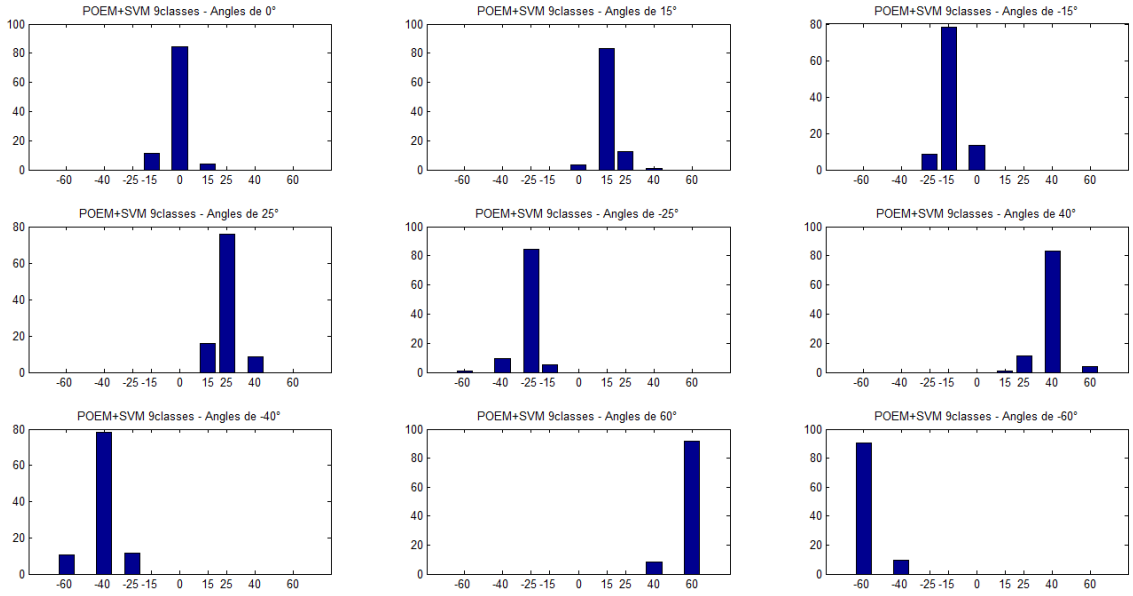

Figure 4.11 – *Répartition des poses estimées dans le cas où nous utilisons un ensemble de SVMs pour la classification. 9 classes ont été considérées. L'angle réel de la pose du visage test d'entré est donné au dessus de chacun des 9 graphes représentés.*

utilisons le descripteur LGBP.

Table 4.5 – *Comparaison des résultats obtenus en utilisant le descripteur POEM ou le descripteur LGBP. L'ensemble de ces résultats a été obtenu sur 864 images de la base FERET.*

Taux moyen d'estimation de la pose obtenu sur la base FERET		
Méthode	Taux d'estimation moyen	Temps
POEM+SVM	83.33%	77ms
LGBP+SVM	84.15%	616ms

Nous pouvons constater sur ce tableau que le résultat obtenu avec la méthode qui combine le descripteur POEM avec un ensemble de SVMs atteint un résultat similaire à l'estimateur utilisé avec le descripteur LGBP. Nous obtenons en effet un taux d'estimation de 83.33% contre 84.15% ce qui est sensiblement équivalent. Ceci mérite d'être souligné dans la mesure où notre estimateur de pose nécessite très peu de temps de calcul comparé à celle basée sur le descripteur LGBP. Nous avons en effet estimé à 77ms le temps nécessaire au calcul du vecteur de caractéristiques d'une image donnée par le descripteur POEM contre 616ms pour le calcul du vecteur de caractéristiques d'une image donnée obtenu avec le descripteur LGBP. Nous avons donc réussi à obtenir un estimateur de pose qui réclame peu de temps de calcul et présente un taux correct de bonne classification. Le taux d'estimation obtenu par classe, c'est à dire le pourcentage d'images, par classe, dont la pose a été correctement estimée avec le descripteur LGBP est donné sur le **Tableau 4.6**.

Table 4.6 – *Taux de bonne classification obtenu pour une pose donnée avec l'estimateur de pose utilisant le descripteur LGBP. 9 classes ont été considérées.*

Répartition de l'estimation pour LGBP+SVM (%)									
-60	-40	-25	-15	0	15	25	40	60	Taux
87.5	81.25	80.21	84.38	86.46	84.38	78.13	82.29	92.71	84.15

Comparé aux résultats présentés dans **Tableaux 4.11**, nous pouvons constater que les taux obtenus pour chacune des poses sont similaires. Mais nous pouvons noter une meilleure estimation pour les angles autour de 0° avec l'algorithme utilisant le descripteur LGBP et une meilleure estimation pour les angles extrêmes avec la méthode utilisant le descripteur POEM. La **Figure 4.12** présente la répartition des angles estimés pour une pose réelle de l'image test lorsque nous utilisons le descripteur LGBP.

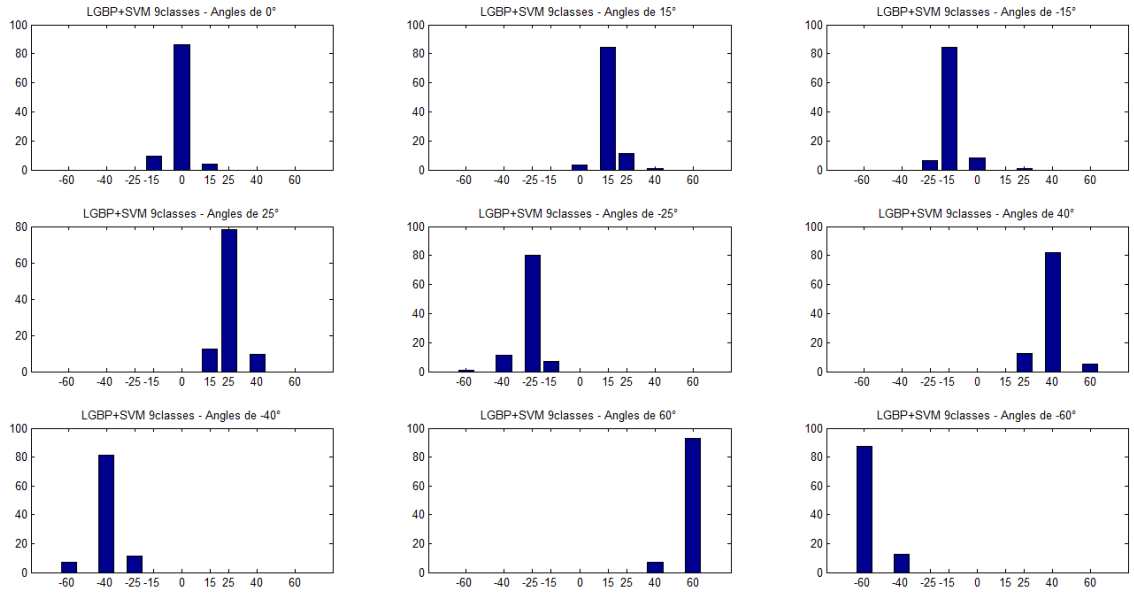

Figure 4.12 – *Répartition des poses donnée par l'estimateur utilisant le descripteur LGBP [MZS+06] pour une image d'entrée avec une pose donnée. L'angle réel de la pose du visage test d'entré est donné au dessus de chacun des 9 graphes représentés.*

Comme nous pouvons le constater, la répartition des erreurs se fait également autour des angles réels à estimer mais il n'y a pas un écart important. La répartition se fait sur les angles directement voisins.

Ainsi, l'utilisation du descripteur POEM pour l'estimation de la pose permet grâce à sa robustesse et son faible coût d'implantation de pouvoir s'appliquer à un contexte de vidéosurveillance. Néanmoins, les taux d'estimation ne sont pas encore suffisants et nous pouvons améliorer ce résultat en proposant une extension de notre estimateur comme nous le présentons dans la section suivante.

6.3 Extension de la méthode d'estimation de pose proposée

6.3.1 Principe de l'extension proposée

Pour améliorer la classification, nous avons décidé d'accentuer l'extraction des caractéristiques en combinant 40 filtres de Gabor avec le descripteur POEM. Il a en effet été démontré [SGjO99] et [SGO00] que les ondelettes de Gabor permettent d'extraire de l'information spécifique à une orientation donnée et par conséquent permettent la construction de descripteurs utiles pour l'estimation de la pose comme c'est le cas pour le descripteur LGBP [MZS+06]. Ainsi, nous avons appliqué dans un premier temps une quarantaine de filtres de Gabor à l'image d'entrée puis, sur chacune de ces images, nous avons calculé le descripteur POEM afin de combiner les deux types d'extracteur. Enfin, nous avons appliqué une ACP sur l'ensemble des données avant de procéder à la classification à partir des SVMs.

Extraction des caractéristiques à partir du descripteur POEM et d'un ensemble de filtres de Gabor

Une fois la représentation en ondelettes de Gabor $G_{\mu,\nu}$ d'un visage obtenue pour une orientation μ et une échelle ν , nous appliquons sur chacune d'entre elle le descripteur POEM. Autrement dit, nous obtenons pour une image $Gm_{\mu,\nu}$ donnée :

$$Poem_{L,w,n}(Gm_{\mu,\nu}) = \{Poem^{\theta_1}, ..., Poem^{\theta_2}, ..., Poem^{\theta_m}\} \tag{4.22}$$

où m représente le nombre d'orientations discrètes choisi (3 dans notre cas).

Ainsi, pour une image donnée, nous obtenons un vecteur *desc* correspondant à la concaténation des 40 vecteurs $Poem_{L,w,n}(Gm_{\mu,\nu})$ obtenus tels que :

$$desc_{L,w,n} = \{Poem_{L,w,n}(Gm_{0,0}), Poem_{L,w,n}(Gm_{\mu,\nu}), ..., Poem_{L,w,n}(Gm_{7,4})\} \tag{4.23}$$

Diminution de la taille des données avec une analyse en composantes principales

Étant donnée la taille exponentielle d'un descripteur, il n'est pas possible d'appliquer une ACP sur le vecteur lui même. Néanmoins, une diminution de la taille des données est indispensable pour effectuer une classification correcte. Nous avons supposé que les descripteurs associés à une orientation μ donnée sont indépendants les uns des autres. Cela n'est pas totalement vrai dans la mesure où il y a toujours un recouvrement des filtres. Nous avons donc effectué une ACP pour chacune des orientations μ. Le descripteur final est obtenu après projection de chacune de ses composantes (associées à une orientation donnée) sur l'espace correspondant et concaténation de l'ensemble après projection.

Étape de classification à l'aide des SVMs

De la même manière que précédemment, nous effectuons une séparation des classes à l'aide de la méthode « une classe contre une autre ». Nous avons utilisé les mêmes paramètres que précédemment pour optimiser l'apprentissage des SVMs.

6.3.2 Résultats

Pour valider notre nouvelle approche, nous comparons les résultats obtenus avec l'estimateur de pose initialement proposé et celui utilisant le descripteur LGBP. Les résultats sont reportés dans le **tableau 4.7**. Nous y présentons les taux d'estimation correspondant au pourcentage d'images ayant été correctement classées parmi l'ensemble des images de la base de test (864 images au total).

Table 4.7 – *Résultats obtenus avec l'estimateur de pose que nous proposons en utilisant le descripteur POEM puis le descripteur POEM combiné à un ensemble d'ondelettes de Gabor et comparaison des résultats en utilisant le descripteur LGBP. L'ensemble de ces résultats a été obtenu sur des images de la base FERET.*

Taux moyen d'estimation de la pose obtenu sur la base FERET	
Méthode	Taux d'estimation moyen
POEM+SVM	83.33%
LGBP+SVM	84.15%
POEM+Gabor+ SVM	88.19%

Comme nous pouvons le constater, la combinaison du descripteur POEM avec des ondelettes de Gabor permet d'améliorer le résultat de la classification. Nous obtenons en effet un taux moyen de 88.19% contre 83.33% initialement. Par ailleurs, le taux obtenu surpasse également celui de l'estimateur de pose utilisant LGBP qui est de 84.15%. Les taux d'estimation obtenus par classe, c'est à dire le pourcentage d'images, par classe, dont la pose a été correctement estimée avec l'extension de la méthode que nous proposons sont présentés sur le **tableau 4.8**.

Table 4.8 – *Taux de bonne classification obtenu pour une pose donnée avec l'extension de notre estimateur de pose basé sur l'extracteur de caractéristiques POEM et sur l'utilisation d'ondelettes de Gabor.*

Répartition de l'estimation pour POEM + Gabor + SVM (%)									
-60	-40	-25	-15	0	15	25	40	60	Taux
92.71	88.54	83.33	85.42	93.75	88.54	80.21	86.46	94.79	88.19

Comme nous pouvons le constater sur ce tableau, la combinaison des ondelettes de Gabor avec le descripteur POEM permet d'améliorer les résultats pour les angles proches de zéro. Nous obtenons en effet un taux d'estimation pour les angles -15°, 0° et 15° de 85.42%, 93.75% et 88.54% contre 78.13%, 84.38% et 83.33% respectivement lorsque nous utilisons uniquement le descripteur POEM. Les résultats obtenus pour les angles extrêmes (±40° et ±60°) ont été sensiblement améliorés également. La répartition des erreurs de classification pour une pose réelle donnée est présentée sur la **Figure 4.13**.

La répartition des erreurs se fait toujours sur les angles les plus proches de l'angle réel. Nous pouvons néanmoins nous étonner que la répartition autour de l'angle zéro ne soit pas homogène alors que l'écart entre les angles 0° et -15° est le même que celui entre les angles 0° et +15°. Il en est de même pour les autres méthodes comme nous pouvons le voir sur les

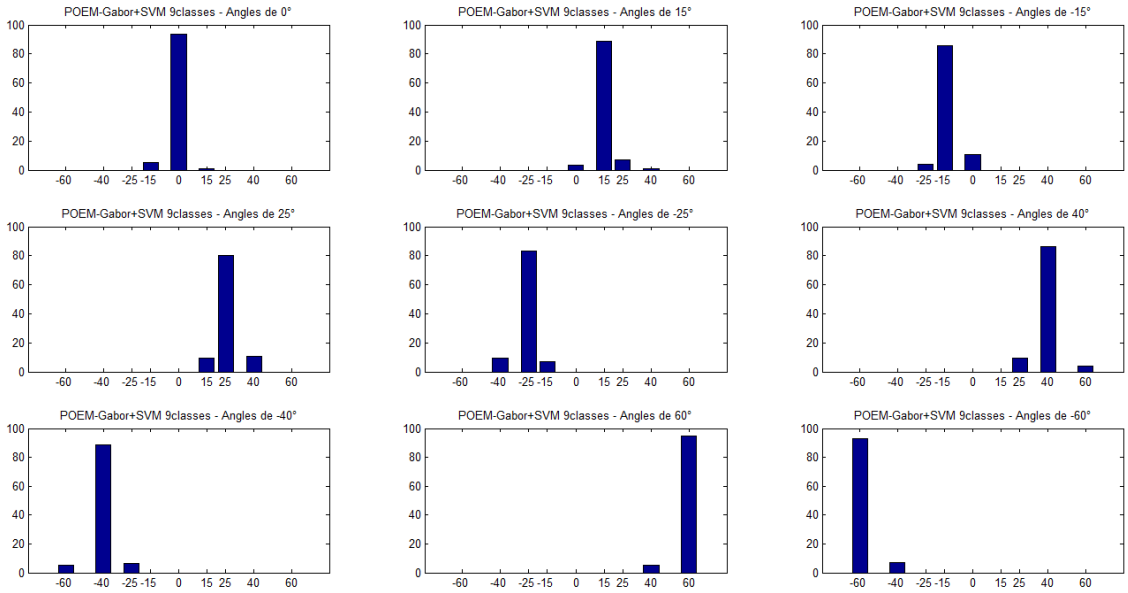

Figure 4.13 – *Répartition des poses données par l'extension de notre estimateur pour une image d'entrée avec une pose donnée. 9 classes ont été considérées. L'angle réel de la pose du visage test d'entré est inscrit au dessus de chacun des 9 graphes représentés.*

Figures : 4.10, 4.11 et 4.12. Cela est peut être due à la façon dont ont été acquises les images de la base FERET. En effet, il est simplement demandé aux candidats de tourner la tête et le corps vers un point donné mais cela n'est pas très précis, il peut y avoir plusieurs degrés d'écarts entre chaque candidat. Cela peut du moins expliqué pourquoi il y a encore un nombre d'erreur encore relativement important lors de l'estimation même si celle-ci est limitée. De plus, nous avons retaillé nos images de façon à ne voir principalement que les yeux, le nez et la bouche de chaque individu. Il est donc assez difficile, même pour quelqu'un d'averti, d'estimer visuellement la pose des visages. Ceci est donc une seconde raison qui pourrait expliquer les erreurs observées lors de la classification quelque soit la méthode utilisée.

Ainsi, nous venons de présenter les résultats obtenus avec notre méthode et son extension. Nous avons montré que la combinaison du descripteur POEM avec les SVMs permet de discriminer de façon efficace les différentes poses de la base de test tout en étant rapide en terme de temps de calcul. L'extension de cet estimateur de pose à l'utilisation des filtres de Gabor a permis d'améliorer de façon significative le taux d'estimation de la pose par rapport à l'utilisation du descripteur LGBP. Par conséquent la combinaison des deux descripteurs POEM et ondelettes de Gabor est plus discriminante que ne l'est la combinaison de LBP avec les ondelettes de Gabor. Néanmoins, le temps de calcul qu'elle

nécessite ne permet pas une application temps réel de l'estimateur.

Conclusion

Nous avons vu que les performances des algorithmes de reconnaissance sont fortement dégradées à partir du moment où la pose du visage à reconnaître présente une variation de la pose importante. En particulier, à partir d'un angle de $\pm 25°$, le taux d'identification des algorithmes de reconnaissance chute considérablement.

Une méthode permettant de s'affranchir de se problème est d'adapter la galerie en fonction de la pose du visage à reconnaître. Il s'agit de disposer d'une galerie par pose et d'estimer la pose du visage d'entrée pour ensuite le comparer aux images de la galerie dont la pose se rapproche le plus. D'autres méthodes permettent de trouver les caractéristiques du visage invariantes à la pose à l'aide de descripteurs spécifiques. Néanmoins, les performances de ces algorithmes sont meilleures lorsque la pose du visage d'entrée est connue.

C'est la raison pour laquelle il existe dans l'état de l'art de nombreuses méthode d'estimation de la pose. Certaines permettent d'estimer la pose à l'angle près tandis que d'autres l'estiment de façon discrète. Dans le cadre de la reconnaissance de visage, les algorithmes sont robustes à de faibles variations de la pose, il n'est donc pas utile de l'estimer finement. C'est la raison pour laquelle nous avons choisi de développer un estimateur de pose en traitant ce problème comme un problème de classification et non de régression.

Nous avons dans un premier temps proposé un estimateur permettant d'estimer la pose d'un visage d'entrée de façon très rapide pour permettre une application à un contexte de vidéosurveillance. Pour cela, nous avons combiné le descripteur POEM à une méthode de classification basée sur les SVMs. Nous avons obtenu des performances similaires à celles obtenues en utilisant le descripteur LGBP qui est utilisé par Ma et al. L'utilisation de POEM permet ainsi d'obtenir de bons résultats tout en permettant une implantation rapide de l'algorithme.

Dans un second temps, nous avons proposé une extension de la méthode proposée afin d'améliorer ses performances. Pour cela, nous avons combiné le descripteur POEM à des ondelettes de Gabor avant de procéder à l'étape de classification par les SVMs. Les résultats obtenus sont meilleurs que ceux obtenus avec le descripteur POEM seul. En revanche, le temps de calcul a été considérablement augmenté ne permettant pas une implantation de l'algorithme en temps réel.

Conclusion générale

Nous avons proposé dans ce manuscrit de thèse une approche permettant d'améliorer la reconnaissance de visages sur des images dégradées dans un contexte de vidéosurveillance. Cette méthode consiste à adapter les images de la galerie en fonction de la dégradation de l'image d'entrée. L'approche présentée se divise en deux étapes. Une étape *d'estimation* qui permet d'évaluer l'ampleur de la dégradation présente dans une image et une étape de *dégradation* qui permet de modéliser la dégradation et de dégrader la galerie de façon adaptée. Dans cette thèse, nous proposons d'améliorer l'une ou l'autre des deux étapes en fonction de la dégradation à laquelle nous nous intéressons, afin de faciliter la mise en œuvre de l'approche dans un contexte de vidéosurveillance. Il est important de souligner que cette méthode se situe en amont du système de reconnaissance ce qui permet de l'utiliser avec n'importe quel algorithme de reconnaissance faciale.

En particulier, nous nous sommes focalisés sur trois types de perturbations couramment rencontrées dans un tel contexte :

1. Le flou
2. Les effets de bloc
3. La pose

En ce qui concerne le flou et l'effet de bloc, nous avons montré le bien fondé de l'approche d'adaptation de la galerie en fonction du niveau des artéfacts présents dans l'image en testant cette méthode sur plusieurs algorithmes de reconnaissance faciale couramment utilisés dans l'état de l'art. En particulier, nous avons positionné notre approche en amont du système de reconnaissance basé sur le descripteur LBP d'une part et du descripteur LPQ d'autre part. Nous avons dégradé les images de la galerie artificiellement en fonction de l'artéfact considéré tandis que l'estimation de la dégradation a été réalisée grâce à des métriques de qualité adaptées (BluM pour estimer le niveau de flou et BLE pour estimer

les artéfacts de bloc). La stratégie de dégradation, combinée à la stratégie d'estimation, a permis d'améliorer les performances des algorithmes de reconnaissance dans les cas où la dégradation est forte, voire très forte.

- Les visages dont les images sont fortement dégradées par l'artéfact de flou sont difficiles à reconnaître par un algorithme de reconnaissance classique. Le BluM permet d'estimer le niveau de flou de l'image et de choisir la galerie dont la force de dégradation est la plus proche de celle de l'image test. Nous avons amélioré le taux de reconnaissance de façon significative pour des images présentant un flou de mise au point tout comme celles présentant un flou de bougé.
- Les algorithmes de reconnaissance que nous avons testés sont relativement robustes aux artéfacts de bloc. Les performances de ces méthodes commencent à se détériorer lorsque la qualité de compression est proche de vingt. Pour estimer la dégradation introduite par la compression, nous avons utilisé la métrique de bloc BLE qui permet de quantifier le nombre de blocs de l'image. La difficulté du travail dans ce cas, réside dans l'établissement d'une table de correspondance entre le facteur de qualité auquel nous n'avons pas accès et le niveau d'effet de bloc estimé par le BLE. Nous avons dégradé la galerie en fonction de cette table de correspondance. Les résultats obtenus ont permis d'améliorer la reconnaissance des images fortement dégradées et de mettre en évidence des propriétés spécifiques aux deux méthodes de reconnaissance de visages testées.

En ce qui concerne la pose, le bien fondé de la stratégie d'adaptation de la galerie pour la reconnaissance de visages dans le cas où la pose varie a déjà été démontré dans l'état de l'art. En revanche, il existe peu d'algorithmes permettant d'estimer la pose d'un individu tout en répondant aux critères que nécessite la reconnaissance de visage dans un contexte non contrôlé. C'est la raison pour laquelle, concernant l'artéfact de pose, nous nous sommes placés au niveau de l'étape d'estimation. Pour cela, nous avons proposé un estimateur basé sur la combinaison du descripteur *POEM* qui permet d'extraire de l'information liée à l'orientation des contours avec une méthode de classification utilisant un ensemble de SVMs. En effet, dans le cadre de la reconnaissance de visages, il n'est pas utile de connaître l'angle de la pose au degré près étant donné que la majorité des algorithmes de reconnaissance sont robustes à une certaine variation de l'angle de la pose. L'algorithme proposé en utilisant le descripteur POEM a permis d'atteindre les mêmes performances que celles obtenues avec le descripteur LGBP tout en étant 8 fois plus rapide. Néanmoins, pour améliorer les performances de cet algorithme, nous avons proposé une extension de l'estimateur de pose proposé en combinant le descripteur *POEM* avec un ensemble d'ondelettes de Gabor. Les résultats obtenus montrent effectivement une amélioration des performances de l'estimateur de base mais au détriment de la rapidité des calculs rendant l'utilisation de la méthode impossible dans un contexte de vidéosurveillance.

Nous avons donc proposé une approche globale permettant de s'affranchir de trois des types de dégradation souvent rencontrées dans le cadre d'un contexte non contrôlé. Chacune de ces trois contributions a permis d'améliorer la reconnaissance d'un visage en permettant une implantation rapide de l'algorithme proposé en amont de la méthode de reconnaissance choisie et en permettant de traiter l'identification d'un visage en n'ayant

à notre disposition qu'une seule image par personne dans la galerie.

Perspectives

Les extensions que nous pouvons apporter à notre travail sont multiples. N'oublions pas que dans le cadre du projet Biorafale, nous travaillons sur des images issues de vidéosurveillance. Or, nous n'avons jamais considéré l'aspect temporel de la vidéo dans le cadre de cette thèse. Nous avons en effet, pour les trois contributions apportées, considéré une seule image test par personne. Or, dans ce contexte, il est a priori facile de disposer de plusieurs images d'une même personne. Combiner l'information apportée par cet ensemble d'images devrait permettre une amélioration notable des performances, par exemple, de l'estimateur de pose proposé.

D'autre part, nous avons testé les performances de nos algorithmes sur des bases d'images qui ne sont pas issues d'un réel contexte de vidéosurveillance. Ceci était nécessaire pour pouvoir maîtriser les dégradations introduites dans les images et pour permettre la comparaison de nos résultats à ceux des autres méthodes de reconnaissance déjà existantes. Maintenant que le bien fondé de l'approche a été montré, il serait intéressant de tester la méthode sur des images prises dans des conditions réelles d'utilisation.

En ce qui concerne l'effet de bloc, il serait intéressant d'améliorer la table de correspondance afin d'estimer le taux de compression de façon plus précise.

Étant donné que dans un contexte non contrôlé, une image peut présenter une combinaison d'artéfacts, il faudrait pouvoir fusionner l'approche afin de proposer une méthode permettant de traiter les artéfacts de flou, d'effets de bloc et de pose dans le même temps.

Annexes

Annexe A

Analyse en Composante Principales (ACP)

Pour rédiger cette annexe, nous nous somme inspiré du très bon didacticiel de J.Shlens sur l'ACP présenté dans [Shl05].

Diagonalisation de la matrice de covariance XX^T

Soit la matrice X de dimension $N \times M$ telle que : $X = [\phi_1, \phi_2, ..., \phi_M]$. On cherche une transformation linéaire qui à X associe Y telle que $Y = PX$. Les colonnes de P forment un jeu de vecteurs de base sur lesquels ont été projetées chaque colonne de X. La question est de savoir comment exprimer de manière judicieuse la base exprimée par la matrice P.

La matrice de covariance C_x de taille $N \times N$ est définie comme suit :

$$C_x = \sum_{i=1}^{M} \phi_i \phi_i^t = XX^t. \tag{A.1}$$

L'ACP suppose que tous les vecteurs de la base sont orthogonaux ce qui revient à dire que :

$$P_i^T P_j = \delta_{i,j} = \begin{cases} 1 \text{ si i=j} \\ 0 \text{ si i} \neq \text{j} \end{cases} \tag{A.2}$$

Autrement dit, nous cherchons une matrice P telle que $Y = PX$ et telle que $C_y = YY^T$ soit diagonale. Rappelons que le principe de l'ACP repose sur une diminution de la redondance d'information, laquelle se traduit en terme mathématiques par la matrice de covariance. Si nous arrivons à diagonaliser cette matrice, alors les covariances (termes non diagonaux dans la matrice) sont toutes nulles.

Exprimons la matrice de covariance C_y en fonction de P :

$$C_y = YY^T \tag{A.3}$$
$$= (PX)(PX)^T \tag{A.4}$$
$$= PXX^TP^T \tag{A.5}$$
$$C_y = P(XX^T)P^T \tag{A.6}$$

Étant donné que XX^T est symétrique et par conséquent diagonalisable, nous pouvons écrire que $XX^T = EDE^T$ où $E = [e_1, e_2, ..., e_r]$ est la matrice des vecteurs propres de XX^T et D sa matrice diagonale associée qui s'écrit de façon plus générale :

$$XX^TE = ED \Leftrightarrow XX^T = EDE^{-1} \tag{A.7}$$

E contient un jeu de vecteurs propres de dimension $N \times 1$. Pour diagonaliser C_y, il ne reste plus qu'à créer une matrice P dont chacune des colonnes est un vecteur propre de XX^T. Dans ce cas, nous avons $P \equiv E^T$ et donc $XX^T = P^TDP$. En effet, C_y peut alors s'écrire comme suit :

$$C_y = PXX^TP^T \tag{A.8}$$
$$= PP^TDPP^T \tag{A.9}$$
$$= PP^{-1}DPP^{-1} \tag{A.10}$$
$$= D \tag{A.11}$$

Autrement dit, nous avons réussi à diagonaliser la matrice de covariance C_x dont la matrice diagonale est $C_y = YY^T$.

Annexe B

Les ondelettes de Gabor

Cette annexe est une adaptation du tutoriel sur les filtres de Gabor proposé par Javier R.Movellan dans [Mov02]

On définit la porteuse $P(z)$ au point de coordonnées $z = (x, y)$ comme suit :

$$P(x, y) = exp(j(2.\pi(u_o.x + v_o.y) + P)) \tag{B.1}$$

où le point de coordonnées (u_o, v_o) représente la fréquence spatiale de la sinusoïde (fréquence centrale du filtre passe-bande de l'ondelette de Gabor) et P représente la phase de la sinusoïde. Comme on le verra par la suite, ce paramètre est généralement mis à zéro dans la plupart des articles sur la reconnaissance de visages utilisant les filtres de Gabor. On peut réécrire l'expression de la porteuse en coordonnées polaire en définissant les points F_o et θ_o tels que :

$$F_o = \sqrt{u_o^2 + v_o^2} \tag{B.2}$$

et (B.3)

$$\theta_o = tan^{-1}\frac{v_o}{u_o} \tag{B.4}$$

On obtient alors :

$$P(x, y) = exp(j(2.\pi.F_o(x.cos(\theta_o) + y.sin(\theta_o)) + P)) \tag{B.5}$$

La fonction enveloppe, quant à elle, est définie au point de coordonnées (x, y) comme suit :

$$E(x, y) = K.exp(-\pi(a^2(x - x_o)_r^2 + b^2(y - y_o)_r^2)) \tag{B.6}$$

Le point de coordonnées (x_o, y_o) est le pic de la fonction, les paramètres a et b définissent

la forme de l'enveloppe gaussienne tandis que l'indice r indique une opération de rotation d'angle θ_g telle que :

$$(x - x_o)_r = (x - x_o).\cos(\theta_g) + (y - y_o).\sin(\theta_g) \tag{B.7}$$
$$(y - y_o)_r = -(x - x_o).\sin(\theta_g) + (y - y_o).\cos(\theta_g) \tag{B.8}$$
$$\tag{B.9}$$

Finalement, on définit une ondelette de Gabor $G(x, y)$ comme le produit de la porteuse $P(x, y)$ par la fonction enveloppe $E(x, y)$:

$$G(x, y) = P(x, y).E(x, y) \tag{B.10}$$

Cependant, ainsi définie, l'ondelette de Gabor ne peut être utilisée dans le domaine de la reconnaissance de visage. La réponse d'un filtre doit en effet être indépendante de la valeur moyenne des niveaux de gris d'une image, ce qui n'est pas le cas de $G(x, y)$. Pour s'affranchir de ce problème, il suffit de définir un nouveau filtre, $H(x, y)$, en soustrayant un filtre passe-bas, $F(x, y)$, au filtre original $G(x, y)$ selon :

$$\boxed{H(x, y) = G(x, y) - C.F(x, y)} \tag{B.11}$$

$F(x, y)$ est un filtre passe-bas de forme gaussienne centré en zéro, (x_0,y_0)=(0,0), qui a la même forme que l'enveloppe $E(x, y)$, soit :

$$F(x, y) = K.exp(-\pi(a^2(x)_r^2 + b^2(y)_r^2)) \tag{B.12}$$
$$\equiv \mathrm{E(x,y)} \tag{B.13}$$

C est une constante que l'on détermine en calculant la réponse continue du filtre, $\hat{H}(0)$, selon :

$$\hat{H}(0) = \hat{G}(0) - C.\hat{F}(0) \tag{B.14}$$

Nous proposons dans les paragraphes suivants de déterminer l'expression des deux transformées de Fourier de $\hat{G}(z)$ et $\hat{F}(z)$ avant d'en déduire celle de $\hat{H}(z)$ qui est le filtre de Gabor régulièrement utilisé dans le domaine de la reconnaissance de visages.

1. Calcul de la transformée de Fourier de $G(x, y)$

On cherche à calculer la transformée de Fourier $\hat{G}(w)$ au point $w = (u, v)$ d'une ondelette de Gabor $G(z) = G(x, y)$.

D'après la définition de la transformée de Fourier :

$$\hat{G}(w) = \int_{-\infty}^{+\infty} g(z).exp(-2\pi j z^T)dz \tag{B.15}$$

$$= \int_{-\infty}^{+\infty} g(x,y).exp(-2\pi j(ux+vy)\ dx.dy \tag{B.16}$$

En fixant le pic de la fonction au point de coordonnées $(x_0, y_0) = (0,0)$, on obtient :

$$\hat{G}(u,v) = \int_{-\infty}^{+\infty} .exp\{-\pi[a^2(xcos(\theta_g) + ysin(\theta_g)^2) + b^2(-xsin(\theta_g) + ycos(\theta_g)^2]\} \times K.exp\{jP\}.exp\{j2\pi((u_0-u)x + (v_0-v)y)\}\ dx.dy \tag{B.17}$$

Nous simplifions l'expression en égalant les écarts type a et b ($a = b = \sigma$) :

$$\hat{G}(u,v) = \int_{-\infty}^{+\infty} .exp\{-\pi\sigma^2(x^2cos^2(\theta_g) + y^2sin^2(\theta_g) + x^2sin^2(\theta_g) + y^2cos^2(\theta_g))\} \times K.exp\{jP\}.exp\{j2\pi((u_0-u)x + (v_0-v)y)\}\ dx.dy \tag{B.18}$$

$$\begin{aligned}\hat{G}(u,v) &= \int_{-\infty}^{+\infty} exp\{-\pi\sigma^2(x^2cos^2(\theta_g) + y^2sin^2(\theta_g) + x^2sin^2(\theta_g) + y^2cos^2(\theta_g))\} \\ &\quad \times K.exp\{jP\}.exp\{j2\pi((u_0-u)x + (v_0-v)y)\}\ dx.dy \\ &= \int_{-\infty}^{+\infty} .exp\{-\pi\sigma^2(x^2+y^2)\} \\ &\quad \times K.exp\{jP\}.exp\{j2\pi((u_0-u)x + (v_0-v)y)\}\ dx.dy \\ &= K.exp\{jP\} \int_{-\infty}^{+\infty} exp\{-\pi\sigma^2x^2\}.exp\{j2\pi(u_0-u)x)\} \\ &\quad \times \int_{-\infty}^{+\infty} exp\{-\pi\sigma^2y^2\}.exp\{j2\pi(v_0-v)y)\}\ dx.dy\end{aligned} \tag{B.19, B.20}$$

$$\text{Or } TF[exp\{-m^2x^2\}] = \frac{\sqrt{(}\pi)}{|m|} exp\left\{\frac{-\pi^2u^2}{m^2}\right\} \text{ d'où :} \tag{B.21}$$

$$\hat{G}(u,v) = \frac{K}{\sigma^2}.exp\{jP\}.exp\left\{\frac{-\pi(u-u_0)^2}{\sigma^2}\right\}.exp\left\{\frac{-\pi(v-v_0)^2}{\sigma^2}\right\} \tag{B.22}$$

(B.23)

$$\text{Soit : } \boxed{\hat{G}(u,v) = \frac{K}{\sigma^2}.exp\left\{jP\}.exp\{\frac{-\pi}{\sigma^2}\left((u-u_0)^2 + (v-v_0)^2\right)\right\}} \tag{B.24}$$

2. Calcul de la transformée de Fourier de $F(x, y)$

$$\hat{F}(w) = \int_{-\infty}^{+\infty} f(z).exp(-2\pi j z^T)dz \tag{B.25}$$

$$= \int_{-\infty}^{+\infty} f(x,y).exp(-2\pi j(ux+vy)\ dx.dy \tag{B.26}$$

$$= \int_{-\infty}^{+\infty} K.exp\{-\pi[a^2(xcos(\theta_g)+ysin(\theta_g)^2) + b^2(-xsin(\theta_g)+ycos(\theta_g)^2]\} \times exp\{j(2\pi(ux+vy)+P)\}\ dx.dy \tag{B.27}$$

Nous simplifions l'expression en égalant les écarts type a et b ($a = b = \sigma$) :

$$\text{Soit : } \boxed{\hat{F}(u,v) = \frac{K}{\sigma^2}.exp\{jP\}.exp\left\{\frac{-\pi}{\sigma^2}(u^2+v^2)\right\}} \tag{B.28}$$

3. Calcul de la transformée de Fourier de $H(x, y)$

La transformée de Fourier $\hat{H}(w)$ de $H(x, y)$ est donnée par $\hat{H}(w) = \hat{G}(w) - C.\hat{F}(w)$. Pour déterminer son expression exacte, il faut dans un premier temps déterminer la constante C. Pour cela, nous devons, comme indiqué plus haut, calculer la réponse continue du filtre.

Soit :

$$\hat{H}(0) = \hat{G}(0) - C.\hat{F}(0) \tag{B.29}$$

$$= 0 \text{ (par définition)} \tag{B.30}$$

$$\tag{B.31}$$

En prenant une phase nulle ($P = 0$) :

$$\hat{H}(0) = \frac{K}{\sigma^2}.\left(exp\left\{\frac{-\pi}{\sigma^2}(u_0^2+v_0^2)\right\} - C\right) \tag{B.32}$$

$$\text{d'où : } C = exp\left\{\frac{-\pi}{\sigma^2}(u_0^2+v_0^2)\right\} \tag{B.33}$$

On obtient donc l'expression de $H(x, y)$ suivante :

$$\boxed{H(x,y) = K.exp\{-\pi\sigma^2(x^2+y^2) \times \left(exp\{j2\pi(u_0x+v_0y)\} - exp\left\{\frac{-\pi}{\sigma^2}(u_0^2+v_0^2)\right\}\right)} \tag{B.34}$$

Cette expression de $H(x, y)$ correspond au filtre de Gabor dont la réponse à l'intensité d'une image est nulle.

Rappelons les trois hypothèses utilisées pour l'obtenir sous cette forme :

1. La phase P est supposée nulle : $P = 0$

2. Le pic de la fonction Gaussienne à deux dimensions est centrée en zéro : $(x_0, y_0) = (0, 0)$

3. Les écarts type a et b de cette fonction Gaussienne sont égaux : $a = b = \sigma$

   Pour aboutir à l'expression des filtres de Gabor régulièrement utilisés dans les articles relatifs à la reconnaissance de visages, une hypothèse supplémentaires est nécessaires, soit 4 hypothèses au total. Les quatrième est la suivante :

4. On pose :

$$u_0 = F_0 cos\theta_0 = \frac{\sigma^2}{\sqrt{2\pi}} cos\theta_0 \tag{B.35}$$

$$v_0 = F_0 sin\theta_0 = \frac{\sigma^2}{\sqrt{2\pi}} sin\theta_0 \tag{B.36}$$

D'où :

$$\boxed{\begin{aligned} H(x,y) = K.exp\{-\pi\sigma^2(x^2+y^2) \\ \times \left(exp\{j\sqrt{2\pi}\sigma^2(cos\theta_0 x + sin\theta_0 y)\} - exp\left\{\frac{-\sigma^2}{2}\right\}\right) \end{aligned}} \tag{B.37}$$

Soit le vecteur d'onde $\vec{k}$, on peut exprimer l'équation B.37 comme suit :

$$\boxed{H(x,y) = \frac{||\vec{k}||^2}{\sigma^2}.exp\{-\frac{||\vec{k}||^2.||z^2||}{2\sigma^2} \times \left(exp\{j\vec{k}.\vec{z}\} - exp\left\{\frac{-\sigma^2}{2}\right\}\right)} \tag{B.38}$$

$$\text{avec : } K = \frac{||\vec{k}||^2}{\sigma^2} \tag{B.39}$$

$$\text{et : } ||\vec{k}|| = \sqrt{2\pi}\sigma^2 \tag{B.40}$$

Annexe C

Généralités sur la compression spatiale et temporelle

Les signaux numériques requièrent une bande passante plus importante que les signaux analogiques. Par conséquent, avec le nombre croissant d'outils numériques sont arrivées de nouvelles techniques permettant de réduire les coûts de stockage et le temps de transmission. Cependant, ces méthodes ont un coût, car elles induisent une perte définitive d'une partie de l'information contenue dans l'image.

La compression peut être de deux types : spatiale ou temporelle selon qu'elle étudie les redondances d'informations dans l'image ou bien les similarités entre images successives (dans le cas de la vidéo uniquement).

Comprendre comment s'obtient la compression d'une image va nous permettre de comprendre l'origine des artéfacts que nous voyons apparaître sur une image compressée et en quoi ces artéfacts peuvent affecter, de façon significative ou non, les algorithmes de reconnaissance de visage.

1. La compression spatiale

La compression spatiale d'une image fixe se fait uniquement par l'étude des redondances ce qui va se traduire, d'un point de vue mathématiques, par une étude de l'image dans le domaine fréquentiel. Cependant, pour éviter de manipuler trop de données à la fois, ce qui induirait là encore des calculs trop coûteux en temps et en espace de stockage, l'image est au préalable découpée en blocs de 8*8 pixels.

Une fois l'image divisée, on applique une DCT (transformée en Cosinus Discrète) sur chacun des blocs, ce qui va permettre de condenser l'information initiale en la décrivant en

fréquence et en amplitude plutôt qu'en pixel (valeurs de la chrominance et de la luminance en un point de l'image). Pour un bloc de taille $N \times N$, la DCT s'exprime comme suit :

$$DCT(i,j) = \frac{2}{N} C(i)C(j) \sum_{x=0}^{N-1} \sum_{x=0}^{N-1} pixel(i,j) cos\{\frac{(2x+1)i\pi}{2N}\} cos\{\frac{(2y+1)i\pi}{2} \} \quad \text{(C.1)}$$

où on définit « C » par :

$$C = \left\{ \begin{array}{ll} \frac{1}{\sqrt{2}} & \text{pour } x = 0 \\ 1 & \text{pour} x > 0 \end{array} \right\} . \quad \text{(C.2)}$$

Ainsi, on obtient pour chaque bloc de l'image une matrice de 8*8 coefficients dont les valeurs permettent de pondérer chaque fréquence spatiale représentative du bloc étudié. La Figure C.1 ci-dessous illustre les motifs (composantes en fréquence) que l'on peut obtenir avec une matrice DCT de taille 8*8. Comme on peut le voir les fréquences deviennent d'autant plus grandes (et donc les détails de l'image d'autant plus fins) que l'on s'approche du coin inférieur droit de la matrice tandis que la première case en haut à gauche correspond à la fréquence nulle qui représente la composante continue de l'image. Autrement dit, le maximum d'information contenu dans l'image est concentré dans le coin supérieur gauche de la matrice.

A ce stade de la compression il n'y a aucune perte d'information sur l'image qu'on peut retrouver par une simple DCT inverse.

A l'étape suivante intervient la quantification qui correspond, quant à elle, à la phase non conservative. Cette étape permet de réduire le nombre de bits de l'image en divisant chaque élément de la matrice DCT en fonction de son importance dans la représentation de l'image. Les coefficients de la matrice de quantification sont ainsi choisis en fonction de la sensibilité du SVH (Système Visuel Humain) aux fréquences puisqu'il est admis que celui-ci agit comme un filtre passe bas. Leurs valeurs dépendent également de plusieurs autres paramètres tels que le taux de compression ou bien le contenu de l'image elle-même.

La dernière étape de compression exploite la redondance dite statistique de l'information en réarrangeant les bits permettant de coder l'image de la façon la plus compacte possible.

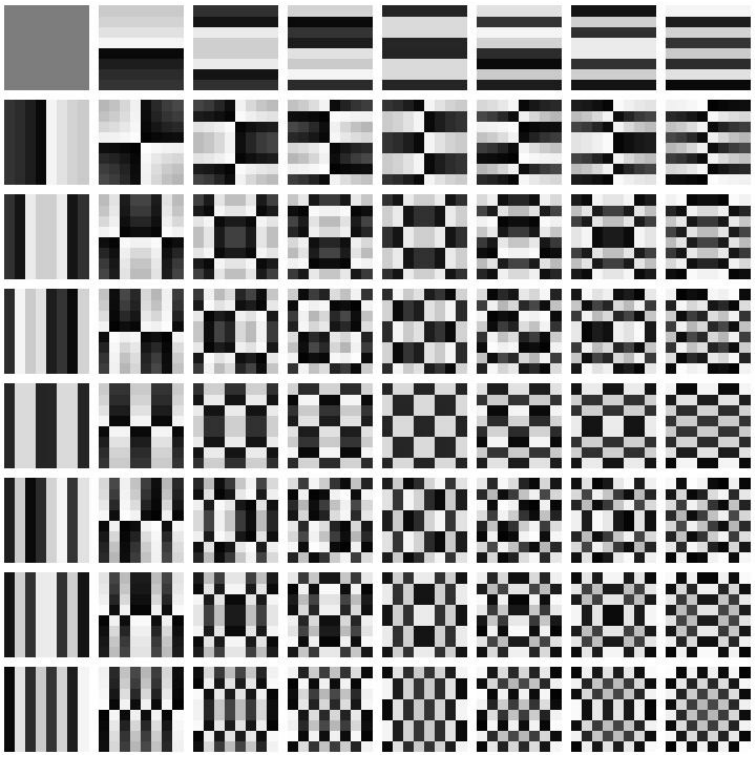

Figure C.1 – *Motifs élémentaires associées à la DCT pour un bloc de* 8×8 *pixels.*

2. La compression temporelle

Celle-ci est basée sur la redondance des informations existant entre plusieurs images successives d'une séquence vidéo.

Une séquence vidéo est composée de 3 types d'images : les images Intra (I), les images Prédites (P) et les images Bidirectionnelles (B).

Une image I est codée sans prédiction par un algorithme de compression. Elle est donc codée selon une compression spatiale et ne fait référence à aucune autre image de la séquence. Autrement dit cette image est décrite de la même façon qu'une image fixe classique ce qui justifie l'intérêt de notre étude faite à partir de visages pris dans la base d'images FERET et non issus de séquences vidéo.

Une image P est prédite d'une image I ou P précédente en utilisant une compensation de mouvement. Par conséquent, elle est codée à l'aide de vecteurs de mouvement indiquant les déplacements de ses éléments par rapport à l'image de référence.

Une image B est décrite avec deux compensations de mouvement. La première est réalisée à partir d'une image I ou P précédente et la seconde à partir d'une image I ou P future. Les images P et B sont donc obtenues grâce à la technique de compensation de mouvement. Celle-ci est basée sur la recherche de macroblocs semblables entre deux images successives et le codage des vecteurs de mouvement qui caractérisent le déplacement de ces blocs entre les deux instants.

Annexe D

Introduction aux machines à vecteurs de support (SVMs)

Cette annexe sur les SVMs se veut suffisamment complet pour permettre une bonne compréhension du sujet sans toutefois mettre tous les détails des calculs pour donner une idée générale et simple de ce que sont les SVMs. Pour plus de détails concernant l'obtention de certaines formules ou pour obtenir plus d'information sur l'apprentissage artificiel en général, le lecteur pourra se référer à l'excellent manuscrit de thèse de Mathieu Feuilloy [FEU09] dont cet annexe s'est également inspiré.

Les $SVMs$ sont basés sur l'utilisation de fonctions qui permettent de séparer de façon optimale les données. Lorsque l'on considère le cas de deux classes dont les données sont linéairement séparables, il existe une infinité d'hyperplans permettant la séparation des observations (cf. **Fig. D.1(a)**).

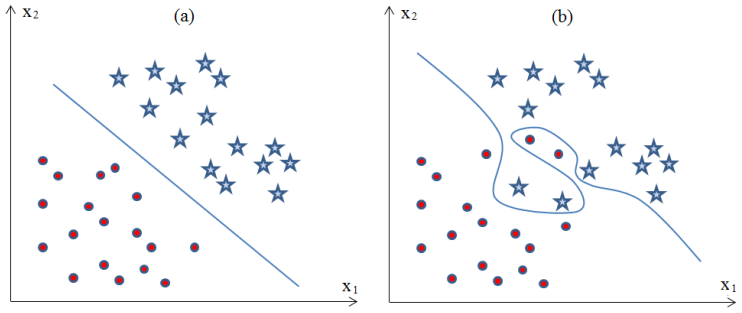

Figure D.1 – *(a) : cas linéairement séparable par une droite de deux échantillons de données représentés dans un plan. (b) : cas non linéairement séparable dans le plan de deux échantillons de données*

Mais un seul permet de maximiser la distance entre cet hyperplan et les observations les plus proches. On appelle cette distance la marge tandis que les observations les plus

proches de cet hyperplan sont les vecteurs de support. Cependant, toutes les données ne sont pas linéairement séparables (cf. **Fig. D.1(b)**). On leur applique alors dans un premier temps une transformation permettant de les représenter dans un espace de plus grande dimension où elles sont linéairement séparables comme on peut le voir sur la **Fig. D.2**. Puis, dans un second temps, on procède à la classification.

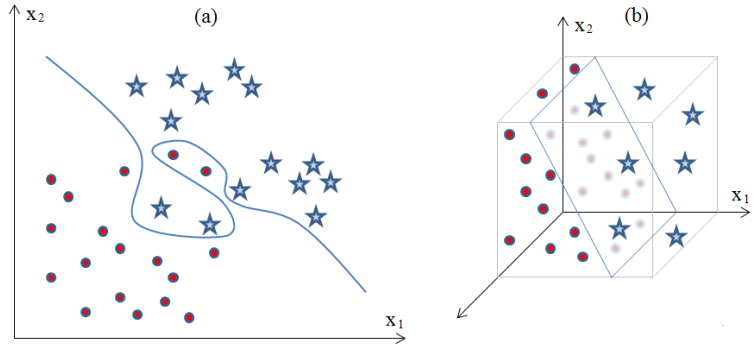

Figure D.2 – *(a) : cas non linéairement séparable dans le plan de deux échantillons de données. (b) : Séparation par un plan des mêmes échantillons de données présentés en (a) après une transformation Φ de celles-ci permettant leur représentation dans un espace à trois dimensions.*

1. Classification linéaire par les SVMs

On se place dans la phase d'apprentissage du cas le plus simple qui soit, à savoir la séparation de données appartenant à deux classes différentes.

1.1 Définition des paramètres qui permettent de définir l'hyperplan séparateur

Nous cherchons à savoir si les données d'entrées X (dans notre cas des images de visages avec deux poses différentes) peuvent se séparer de façon optimale. Pour cela, on cherche une fonction f qui, appliquée aux échantillons d'entrée x produit la sortie y telle que $y = f(x)$. Cette fonction représente l'hyperplan permettant de séparer de façon linéaire les deux classes. Elle a la forme :

$$y(\mathbf{x}) = \vec{w}^T \vec{x} + w_0 \tag{D.1}$$

Ceci n'est rien d'autre qu'une combinaison linéaire des vecteurs d'entrées pondérés par les poids $\vec{w}$ et le biais w_0 qui sont les paramètres à déterminer. Pour cela, on souhaite trouver une fonction h qui à X associe Y tel que la probabilité $P(h(X) = Y)$ soit minimale. On connaît l'échantillon d'apprentissage X ainsi que les sorties Y qui représentent simplement

la classe des données contenues dans l'échantillon X. Autrement dit, on peut dire que $Y = \{-1, +1\}$ mais on considérera plutôt une fonction f à valeurs dans $\mathbb{R}$ telle que :

$$f : X \to \mathbb{R} \tag{D.2}$$

Dans ce cas, la classe est donnée par le signe de $y = f(x)$. Un vecteur d'entrée est donc assigné à la classe $(+1)$ si le signe de $f(x)$ est positif et à la classe (-1) si le signe est négatif. Ainsi, pour séparer des données représentées dans un espace à D dimensions, l'hyperplan séparateur sera à $(D-1)$ dimensions.

On souhaite maximiser la marge γ qui représente la distance de l'hyperplan à l'observation x_a :

$$\gamma = \frac{y(\vec{x_a})}{||\vec{w}||} \tag{D.3}$$

Calcul de la marge γ :
Un vecteur d'entrée $\vec{x}$ appartient à la classe (+1) si $y(\vec{x}) > 0$ et à la classe (-1) si $y(\vec{x}) < 0$. La frontière de décision est donc donnée par $y(\vec{x}) = 0$ qui est l'équation d'un hyperplan. Le projeté orthogonal x_T de x sur la frontière de décision vérifie :

$$y(x_T) = \vec{w}^T \vec{x}_T + w_0 = 0 \tag{D.4}$$

Si deux points A et B appartiennent à la frontière de décision : $y(x_A) = y(x_B)$ et donc $\vec{w}^T(\vec{x}_A - \vec{x}_B) = 0$. Par conséquent $\vec{w}$ est perpendiculaire à la frontière de décision car il est perpendiculaire à tous les vecteurs appartenant à celle-ci. Autrement dit : $\vec{w}$ détermine l'orientation de la frontière de décision.
Soit la distance γ, la distance la plus courte entre un point X et l'hyperplan. Les coordonnées $\vec{x}$ de ce point peuvent donc se décomposer de la façon suivante : $\vec{x} = \vec{x}_T + \gamma \frac{\vec{w}}{||\vec{w}||}$ où $\vec{x}_T$ est la projection orthogonale de X sur la surface de décision. Sachant que $\vec{w}^T.\vec{x}_T + w_0 = 0$, on peut donc écrire :

$$y(x) = \vec{w}^T.\vec{x}_T + w_0 + \vec{w}^T.\gamma||\vec{w}|| \tag{D.5}$$

$$et : \gamma = \frac{y(\vec{x})}{||\vec{w}||} \tag{D.6}$$

Autrement dit, les observations d'apprentissage sont à une distance au moins égale ou supérieure à γ et satisfont l'expression $y_i(\vec{w}^T\vec{x} + w_0) \geq \gamma$ où $y_i = \pm 1$. Pour simplifier, nous pouvons normaliser $||\vec{w}||$ et w_0 de telle manière que la valeur de $y(\vec{x})$ aux points les plus proches des classes soit égale à 1 si $\vec{x}$ appartient à la classe 1 et à -1 si $\vec{x}$ appartient à la classe 2. La marge s'exprime alors comme suit :

$$\gamma = \frac{2}{||\vec{w}||} \tag{D.7}$$

et les données satisfont l'équation $y_i(\vec{w}^T\vec{x} + w_0) \geq 1$. Un schéma récapitulatif est donné sur **Fig. D.3**.

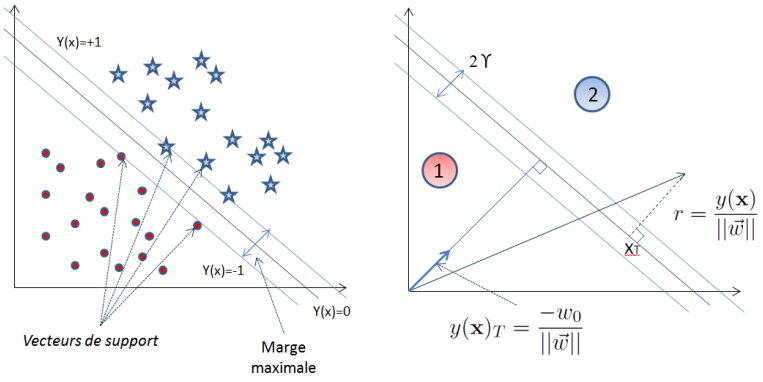

Figure D.3 – *Représentation de l'hyperplan optimal et des variables qui lui sont associées.*

1.2 Maximisation de la marge γ

On souhaite maximiser la marge γ donnée, selon D.4, par $\frac{2}{||\vec{w}||}$ ce qui revient à minimiser le terme $\frac{1}{2}||\vec{w}||^2$. Il s'agit donc de trouver les paramètres $\vec{w}$ et w_0 qui permettent de vérifier les conditions suivantes :

$$\left\{ \begin{array}{l} \text{Minimiser } \frac{1}{2}||\vec{w}||^2 \\ \text{tel que } y_i(\vec{w}^T\vec{x} + w_0) \geq 1 \ , \ \forall\{i = 1, ..., n\} \end{array} \right\} . \tag{D.8}$$

Cette optimisation concerne $p+1$ paramètres ($\vec{w}$ et w_0) où p est la dimension des x_i qui sont les n variables d'entrée. Cette approche du problème est appelée formulation primale. Pour simplifier, on fait intervenir une fonction appelé Lagrangien qui permet d'énoncer la formulation duale du problème comme suit :

$$\zeta(\vec{w}, w_0, \alpha) = \frac{1}{2}||\vec{w}||^2 - \sum_{i=1}^{n} \alpha_i(y_i(\vec{w}^T\vec{x} + w_0) - 1) \tag{D.9}$$

Les α_i sont les multiplicateurs de Lagrange tous définis positifs.

L'optimisation de ce problème dual est obtenue en déterminant le point pour lequel les dérivées du Lagrangien par rapport aux paramètres s'annulent :

$$\left\{ \begin{array}{l} \frac{\partial}{\partial w}\zeta(\vec{w}, w_0, \alpha) = 0 \\ \\ \frac{\partial}{\partial w_0}\zeta(\vec{w}, w_0, \alpha) = 0 \end{array} \right\} . \tag{D.10}$$

La résolution mène à :

$$\vec{w} = \sum_{i=1}^{n} \alpha_i y_i \vec{x}_i \tag{D.11}$$

&

$$\sum_{i=1}^{n} \alpha_i y_i = 0 \tag{D.12}$$

$$\tag{D.13}$$

Selon les conditions de Karush-Kuhn Tucker, au point où les dérivés de la fonction s'annulent, les multiplicateurs de Lagrange vérifient l'égalité suivante :

$$\alpha_i(y_i(\vec{w}^T\vec{x} + w_0) - 1) = 0 \ , \forall\{i = 1, ...n\} \tag{D.14}$$

Autrement dit, seules sont considérées les observation $\vec{x}_i$ appartenant aux deux hyperplans définis par $\vec{w}^T\vec{x} + w_0 = \pm 1$. En effet, tous les multiplicateurs de Lagrange correspondant aux données $\vec{x}_i$ qui n'appartiennent pas à ces hyperplans sont nuls. Les observations pour lesquelles ces multiplicateurs sont non nuls sont les vecteurs de support. Après simplification, nous arrivons à la forme duale du problème qui s'exprime comme suit :

$$\left\{ \begin{array}{l} \max\limits_{\alpha}\{\sum\limits_{i=1}^{n} \alpha_i - \dfrac{1}{2}\sum\limits_{i,j=1}^{n} \alpha_i\alpha_j y_i y_j \vec{x_i}^T\vec{x_j}\} \\ \alpha_i \geq 0 \ , \forall\{i = 1, ..., n\} \\ \\ \sum_{i=1}^{n} \alpha_i y_i = 0 \end{array} \right\} . \tag{D.15}$$

et dont la solution est l'hyperplan de marge maximale $\frac{2}{||\vec{w}||}$ est :

$$y(\vec{x}) = \sum_{\vec{x}_i \in \mathcal{S}} \alpha_i y_i \vec{x_i}^T\vec{x} + w_0 \tag{D.16}$$

$\mathcal{S}$ est l'ensemble des vecteurs de support.

2. Classification non linéaire par les SVMs

Le cas présenté dans le paragraphe précédent est un cas idéal que l'on rencontre peu. Cependant plusieurs méthodes permettent de s'affranchir du problème de non linéarité. Une des techniques consiste à utiliser des variables, dites ressort, permettant de conserver les équations telles qu'elles ont été formulées au paragraphe précédent tout en admettant une tolérance au niveau de la marge ainsi définie. Une seconde solution consiste à exprimer les données dans un espace de dimension supérieure dans lequel elles peuvent être séparées de façon linéaire.

2.1 Utilisation des variables ressort

Les variables ressort, notées ξ, permettent de relâcher les contraintes au niveau de l'expression de la marge de façon à admettre des erreurs mais en les minimisant. Ainsi, le problème primal revient à minimiser l'équation :

$$\frac{1}{2}||\vec{w}||^2 + C\sum_{i=1}^{n}\zeta_i \tag{D.17}$$

avec :

$$y_i(\vec{w}^T\vec{x} + w_0) \geq 1 - \zeta_i \tag{D.18}$$

C est un coefficient > 0 permettant de réguler l'erreur admise et de rendre le système plus ou moins contraignant en fonction de la marge ainsi définie.

De la même manière que précédemment, on utilise la formulation duale pour résoudre le problème. Les multiplicateurs de Lagrange présentent cette fois un maximum mais de la même manière que précédemment, on ne s'occupe que des variables correspondant aux vecteurs de support pour lesquels les α_i ne sont pas nuls.

$$\left\{ \begin{array}{l} \max\limits_{\alpha}\{\sum\limits_{i=1}^{n}\alpha_i - \frac{1}{2}\sum\limits_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j \vec{x_i}^T\vec{x_j}\} \\ 0 \leq \alpha_i \leq C \ , \ \forall\{i = 1, ..., n\} \\ \\ \sum_{i=1}^{n}\alpha_i y_i = 0 \end{array} \right\} \tag{D.19}$$

2.2 Transformation des données à l'aide de fonctions noyaux

Le but est d'appliquer aux variables qui ne sont pas linéairement séparables une transformation Φ qui permet de les reformuler dans un espace de plus grande dimension où elles seront séparables. On cherche donc, dans un premier temps, une fonction Φ qui à $\vec{x}$ associe $\Phi(\vec{x})$ telle que :

$$\mathbb{R}^p \rightarrow \mathbb{R}^q \text{ où p<q.} \tag{D.20}$$

$$\Phi : \vec{x} \rightarrow \Phi(\vec{x}) \tag{D.21}$$

Dès lors l'équation de la frontière de décision s'exprime comme suit :

$$y(\mathbf{x}) = \vec{w}^T\Phi(\vec{x}) + w_0 \tag{D.22}$$

et le problème dual est alors donné par les contraintes :

$$\left\{ \begin{array}{l} \max\limits_{\alpha}\{\sum\limits_{i=1}^{n}\alpha_i - \frac{1}{2}\sum\limits_{i,j=1}^{n}\alpha_i\alpha_j y_i y_j \Phi(\vec{x_i})^T\Phi(\vec{x_j})\} \\ 0 \leq \alpha_i \leq C \ , \ \forall\{i = 1, ..., n\} \\ \\ \sum_{i=1}^{n}\alpha_i y_i = 0 \end{array} \right\} \tag{D.23}$$

La difficulté réside ici dans le calcul du produit scalaire $\Phi(\vec{x_i})^T\Phi(\vec{x_j})$ qu'il n'est pas toujours évident de déterminer étant donnée la dimension des vecteurs $\Phi(\vec{x})$ qui peut s'avérer extrêmement grande voire infinie. C'est pourquoi il est d'usage, pour faciliter les calculs, de faire appel à des fonctions noyaux notées $K(.,.)$ pour lesquelles il n'est pas nécessaire de calculer les vecteurs $\Phi(\vec{x_i})$ et $\Phi(\vec{x_j})$ afin d'en déterminer le produit scalaire $\Phi(\vec{x_i})^T\Phi(\vec{x_j})$. Celui-ci peut en effet être calculé directement à partir des vecteurs d'observations d'entrée telles que $K(\vec{x_i},\vec{x_j}) = \Phi(\vec{x_i})^T\Phi(\vec{x_j})$. Il existe plusieurs fonctions noyaux régulièrement utilisées comme :

Les fonctions polynomiales de degré d, c étant une constante :

$$K(\vec{x_i},\vec{x_j}) = (\vec{x_i}^T\vec{x_j} + c)^d \tag{D.24}$$

Les fonctions gaussiennes d'écart type σ :

$$K(\vec{x_i},\vec{x_j}) = e^{-\frac{||(\vec{x_i}-\vec{x_j})||^2}{2\sigma}} \tag{D.25}$$

Une fois cette transformation réalisée, le principe de la méthode ne change pas. La seule différence réside dans le calcul de ce produit scalaire. La forme de la solution du système ne change pas. Un récapitulatif de la méthode de classification non linéaire est présentée sur la **Fig. D.4**.

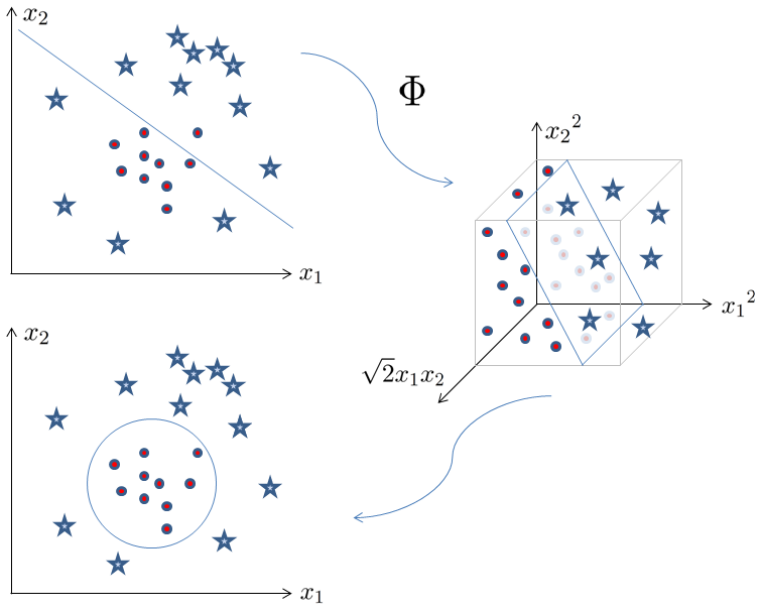

Figure D.4 – *Principe de la classification non linéaire. Les données sont exprimés dans un nouvel espace à partir d'une fonction noyau ϕ (ici polynomiale de degré 2) pour permettre une séparation non linéaire des classes dans l'espace initial de description des données.*

Bibliographie

[AHP04] Timo Ahonen, Abdenour Hadid et Matti Pietikainen: *Face Recognition with Local Binary Patterns.*
Dans *Lecture Notes in Computer Science*, tome 3021/2004, pages 469–481, Berlin, Heidelberg, 2004. Springer-Verlag.

[AKK] Stylianos Asteriadis, Kostas Karpouzis et Stefanos Kollias: *Head Pose Estimation with One Camera, in Uncalibrated Environments.*

[ALC08] Ahmed Bilal Ashraf, Simon Lucey et Tsuhan Chen: *Learning patch correspondences for improved viewpoint invariant face recognition.*
Dans *CVPR'08*, pages –1–1, 2008.

[AROH08] Timo Ahonen, Esa Rahtu, Ville Ojansivu et Janne Heikkilä: *Recognition of blurred faces using Local Phase Quantization.*
Dans *ICPR*, pages 1–4, 2008.

[BCT09] Jean Yves Baudouin, Valérian Chambon et Guy Tiberghien: *Expert en visages ? Pourquoi sommes-nous tous des experts en reconnaissance des visages.*
Lèvolution Psychiatrique, 74(1) :3–25, 2009.
`http://linkinghub.elsevier.com/retrieve/pii/S0014385508001539`.

[Bey94] David J Beymer: *Face recognition under varying pose*, pages 756–761.
Numéro 1461. IEEE Comput. Soc. Press, 1994.
`http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=323893`.

[BGPV05] Volker Blanz, Patrick Grother, Jonathon P. Phillips et Thomas Vetter: *Face Recognition Based on Frontal Views Generated from Non-Frontal Images.*
Computer Vision and Pattern Recognition, IEEE Computer Society Conference on, 2 :454–461, 2005, ISSN 1063-6919.
`http://dx.doi.org/10.1109/CVPR.2005.150`.

[BHK97] P. N. Belhumeur, J. P. Hespanha et D. J. Kriegman: *Eigenfaces vs. Fisherfaces : recognition using class specific linear projection.*
IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(7) :711–720, juillet 1997, ISSN 01628828.
http://dx.doi.org/10.1109/34.598228.

[BK97] M.R. Banham et A.K. Katsaggelos: *Digital image restoration.*
Signal Processing Magazine, IEEE, 14(2) :24–41, 1997.

[BMS02] M. S. Bartlett, J. R. Movellan et T. J. Sejnowski: *Face recognition by independent component analysis.*
IEEE Transactions on Neural Networks, 13(6) :1450–1464, novembre 2002, ISSN 1045-9227.
http://dx.doi.org/10.1109/TNN.2002.804287.

[BP93] Roberto Brunelli et Tomaso Poggio: *Face recognition : features versus templates.*
IEEE Transactions on Pattern Analysis and Machine Intelligence, 15(10) :1042–1052, 1993.

[BR09] Ben Benfold et Ian Reid: *Guiding Visual Surveillance by Tracking Human Attention.*
Dans *BMVC'09*, pages –1–1, 2009.

[BRV02] V. Blanz, S. Romdhani et T. Vetter: *Face identification across different poses and illuminations with a 3D morphable model.*
pages 202–207, 2002.

[BS95] A. J. Bell et T. J. Sejnowski: *An information-maximization approach to blind separation and blind deconvolution.*
Neural Comput, 7(6) :1129–59, novembre 1995.

[BV03] Volker Blanz et Thomas Vetter: *Face recognition based on fitting a 3D morphable model.*
IEEE Transactions on Pattern Analysis and Machine Intelligence, 25 :2003, 2003.

[BYP07] Vineeth Nallure Balasubramanian, Jieping Ye et Sethuraman Panchanathan: *Biased Manifold Embedding : A Framework for Person-Independent Head Pose Estimation.*
Dans *CVPR'07*, pages –1–1, 2007.

[CB10] Chunhua Chen et Jeffrey A. Bloom: *A Blind Reference-Free Blockiness Measure.*
Dans *PCM (1)'10*, pages 112–123, 2010.

[CBB+08] Shaokang Chen, Erik Berglund, Abbas Bigdeli, Conrad Sanderson et Brian C. Lovell: *Experimental Analysis of Face Recognition on Still and CCTV Images.*
Dans *Proceedings of the 2008 IEEE Fifth International Conference on Advanced Video and Signal Based Surveillance*, pages 317–324, Washington, DC, USA, 2008. IEEE Computer Society, ISBN 978-0-7695-3341-4.

[CET01] Timothy F. Cootes, Gareth J. Edwards et Christopher J. Taylor: *Active Appearance Models.*
IEEE Transactions on Pattern Analysis and Machine Intelligence, 23 :681–685, 2001, ISSN 0162-8828.

[CG02] Jorge E. Caviedes et Sabri Gurbuz: *No-reference sharpness metric based on local edge kurtosis.*
Dans *ICIP (3)'02*, pages 53–56, 2002.

[CKP+09] CH. Chan, J. Kittler, N. Poh, T. Ahonen et M. Pietikainen: *(Multiscale) local phase quantization histogram discriminant analysis with score normalisation for robust face recognition.*
Dans *Proc. 1st IEEE Workshop on Video-Oriented Object and Event Classification, Kyoto, Japan, in press*, 2009.

[CMH+11] Shaokang Chen, Sandra Mau, Mehrtash T. Harandi, Conrad Sanderson, Abbas Bigdeli et Brian C. Lovell: *Face recognition from still images to video sequences : a local-feature-based framework.*
J. Image Video Process., 2011 :1–11, January 2011, ISSN 1687-5176.

[CRDLN07] Frédérique Crété-Roffet, Thierry Dolmiere, Patricia Ladret et Marina Nicolas: *The Blur Effect : Perception and Estimation with a New No-Reference Perceptual Blur Metric.*
Dans *SPIE proceedings SPIE Electronic Imaging Symposium Conf Human Vision and Electronic Imaging*, tome XII, pages EI 6492–16, San Jose États-Unis d'Amérique, January 2007.

[CTCG95] T. F. Cootes, C. J. Taylor, D. H. Cooper et J. Graham: *Active shape models - their training and application.*
Comput. Vis. Image Underst., 61 :38–59, January 1995, ISSN 1077-3142.
`http://dl.acm.org/citation.cfm?id=206543.206547`.

[CTL94] T.F. Cootes, C.J. Taylor et A. Lanitis: *Automatic Face Identification System Using Flexible Appearance Models.*
pages xx–yy, 1994.

[CZH+03] L. Chen, L. Zhang, Y. Hu, M. Li et H. Zhang: *Head pose estimation using Fisher Manifold learning.*
Dans *IEEE International Workshop on Analysis and Modeling of Faces and Gesture*, pages 203–207, octobre 2003.
`http://dx.doi.org/10.1109/AMFG.2003.1240844`.

[Dau85] J. G. Daugman: *Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters.*
Journal of the Optical Society of America A : Optics, Image Science, and Vision, 2(7) :1160–1169, 1985.

[DBBB03] Bruce A. Draper, Kyungim Baek, Marian Stewart Bartlett et J. Ross Beveridge: *Recognizing faces with PCA and ICA.*
Comput. Vis. Image Underst., 91 :115–137, July 2003, ISSN 1077-3142.

http://dx.doi.org/10.1016/S1077-3142(03)00077-8.

[DC86] R. Diamond et S. Carey: *Why faces are and are not special : an effect of expertise.*
Journal of experimental psychology. General, 115(2) :107–117, juin 1986, ISSN 0096-3445.
http://view.ncbi.nlm.nih.gov/pubmed/2940312.

[DV09] Josh P. Davis et Tim Valentine: *CCTV on trial : Matching video images with the defendant in the dock.*
Applied Cognitive Psychology, 23(4) :482–505, 2009, ISSN 1099-0720.
http://dx.doi.org/10.1002/acp.1490.

[FDW+02] Pina Marziliano Frederic, Frederic Dufaux, Stefan Winkler, Touradj Ebrahimi et Genimedia Sa: *A No-Reference Perceptual Blur Metric.*
Dans *IEEE 2002 International Conference on Image Processing*, pages 57–60, 2002.

[FEU09] Mathieu FEUILLOY: *Étude d'algorithmes d'apprentissage artificiel pour la prédiction de la syncope chez l'homme.*
Thèse de doctorat, Université d'Angers, 2009.

[FK07] Rony Ferzli et Lina J. Karam: *A No-Reference Objective Image Sharpness Metric Based on Just-Noticeable Blur and Probability Summation.*
Dans *ICIP (3)'07*, pages 445–448, 2007.

[FR07] F.Crété-Roffet: *Estimer, mesurer et corriger les artefacts de compression pour la télévision numérique.*
Thèse de doctorat, GIPSA-Lab - Université Joseph Fourier, 2007.

[FS98] Jan Flusser et Tomas Suk: *Degraded Image Analysis : An Invariant Approach.*
IEEE Trans. Pattern Analysis and Machine Intelligence, 20 :590–603, 1998.

[GF09] Valérie Gouaillier et Aude Emmanuelle Fleurant: *La vidéosurveillance intelligente : promesses et défis.*
rapport technique, CRIM et Technopôle Défense et Sécurité, 2009.
http://www.technopoleds.org/upload/technopoleds/editor/asset/La%20vid%C3%A9osurveillance%20intelligente%20promesses%20et%20d%C3%A9fis.pdf.

[GGM+96] Shaogang Gong, Shaogang Gong, Stephen Mckenna, Stephen Mckenna, John J. Collins et John J. Collins: *An Investigation into Face Pose Distributions.*
Dans *In Proc. IEEE International Conference on Face and Gesture Recognition*, pages 265–270, 1996.

[GLC00] Guodong Guo, Stan Z. Li et Kapluk Chan: *Face Recognition by Support Vector Machines.*
Dans *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition 2000*, FG '00, pages 196–, Washington, DC, USA, 2000. IEEE Computer Society, ISBN 0-7695-0580-5.

http://portal.acm.org/citation.cfm?id=795661.796198.

[GMB02] Ralph Gross, Iain Matthews et Simon Baker: *Appearance-Based Face Recognition and Light-Fields.*
IEEE Transactions on Pattern Analysis and Machine Intelligence, 26 :449–465, 2002.

[GQ05] Yongsheng Gao et Yutao Qi: *Robust visual similarity retrieval in single model face databases.*
Pattern Recogn., 38 :1009–1020, July 2005, ISSN 0031-3203.
http://dx.doi.org/10.1016/j.patcog.2004.12.006.

[HBK08] Pablo H. Hennings-Yeomans, Simon Baker et B.V.K. Vijaya Kumar: *Simultaneous super-resolution and feature extraction for recognition of low-resolution faces.*
Dans *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, June 2008.

[HH06] H. Hu et G. De Haan: *Low cost robust blur estimator*, 2006.

[HHP+01] Bernd Heisele, Purdy Ho, Tomaso Poggio, Ý Purdy Ho et Þ Tomaso Poggio: *Face Recognition with Support Vector Machines : Global versus Component-based Approach.*
Dans *In Proc. 8th International Conference on Computer Vision*, pages 688–694, 2001.

[HNS10] Abdenour Hadid, Masashi Nishiyama et Yoichi Sato: *Recognition of Blurred Faces via Facial Deblurring Combined with Blur-Tolerant Descriptors.*
Dans *ICPR'10*, pages 1160–1163, 2010.

[HSW98] Jeffrey Huang, Xuhui Shao et Harry Wechsler: *Face Pose Discrimination Using Support Vector Machines (SVM)*, 1998.

[HWS10] Rania Hassen, Zhou Wang et Magdy Salama: *No-reference image sharpness assessment based on local phase coherence measurement.*
Dans *ICASSP'10*, pages 2434–2437, 2010.

[HYD96] T. Horprasert, Y. Yacoob et L. S. Davis: *Computing 3-D head orientation from a monocular image sequence.*
Dans *Proceedings of the 2nd International Conference on Automatic Face and Gesture Recognition (FG '96)*, FG '96, pages 242–, Washington, DC, USA, 1996. IEEE Computer Society, ISBN 0-8186-7713-9.
http://dl.acm.org/citation.cfm?id=524467.796020.

[HYX08] Zhang Hua, Zhou Yiran et Tian Xiang: *A Weighted Sobel Operator-Based No-Reference Blockiness Metric.*
Dans *PACIIA (1)'08*, pages 1002–1006, 2008.

[JH91] C. Jutten et J. Herault: *Blind separation of sources, Part I : An adaptive algorithm based on neuromimetic architecture.*
Signal Process., 24(1) :1–10, 1991.

[KCG11] Nils Krahnstoever, Ming Ching Chang et Weina Ge: *Gaze and Body Pose Estimation from a Distance.*
Dans *2011 8th IEEE International Conference on Advanced Video and Signal-Based Surveillancei (AVSS)*, page 6, Aug. 2011.

[KO05] Pavel Korshunov et Wei Tsang Ooi: *Critical video quality for distributed automated video surveillance.*
Dans *Proceedings of the 13th annual ACM international conference on Multimedia*, MULTIMEDIA '05, pages 151–160, New York, NY, USA, 2005. ACM, ISBN 1-59593-044-2.

[KPM] N. Krüger, M. Pötzsch et C. V. D. Malsburg: *Determination of Face Position and Pose With a Learned Representation Based on Labelled Graphs.*

[KS08] H. Keval et M. A. Sasse: *Can we ID from CCTV ? Image quality in digital CCTV and face identification performance.*
Dans *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, tome 6982 de *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, mai 2008.

[KY03] Takeo Kanade et Akihiko Yamada: *Multi-subregion based probabilistic approach toward pose-invariant face recognition.*
Dans *In IEEE International Symposium on Computational Intelligence in Robotics and Automation*, tome 2, pages 954–959, 2003.
`http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.69.611`.

[LFG+01] Stan Z. Li, QingDong Fu, Lie Gu, Bernhard Schölkopf, Yimin Cheng et HongJiag Zhang: *Kernel Machine Based Learning For Multi-View Face Detection and Pose Estimation*, 2001.

[LGL00] Yongmin Li, Shaogang Gong et H. Liddell: *Support vector regression and classification based multi-view face detection and recognition.*
Dans *Proceedings of Fourth IEEE International Conference on Automatic Face and Gesture Recognition*, pages 300–305, mars 2000.
`http://dx.doi.org/10.1109/AFGR.2000.840650`.

[LTC97] Andreas Lanitis, Chris J. Taylor et Timothy F. Cootes: *Automatic interpretation and coding of face images using flexible models.*
IEEE Transactions on Pattern Analysis and Machine Intelligence, 19 :743–756, 1997.

[LVB+93] Martin Lades, Jan C. Vorbrüggen, Joachim Buhmann, Jörg Lange, Christoph V. D. Malsburg, Rolf P. Würtz et Wolfgang Konen: *Distortion Invariant Object Recognition in the Dynamic Link Architecture.*
IEEE Trans. Computers, 42 :300–311, 1993.

[LW00] Chengjun Liu et Harry Wechsler: *Evolutionary Pursuit and Its Application to Face Recognition.*
IEEE Trans. Pattern Anal. Mach. Intell., 22 :570–582, June 2000, ISSN 0162-8828.

http://portal.acm.org/citation.cfm?id=355091.355093.

[LW02] Chengjun Liu et H. Wechsler: *Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition.* Image Processing, IEEE Transactions on, 11(4) :467–476, April 2002.

[LYBW11] C. Li, W. Yuan, A.C. Bovik et X. Wu: *No-reference blur index using blur comparisons.* Electronics Letters, 47(17) :962 –963, 2011, ISSN 0013-5194.

[Man92] *A feature based approach to face recognition*, juin 1992. http://dx.doi.org/10.1109/CVPR.1992.223162.

[Mar02] Aleix M. Martinez: *Recognizing Imprecisely Localized, Partially Occluded and Expression Variant Faces from a Single Sample per Class*, 2002.

[MCT09] Erik Murphy-Chutorian et Mohan Manubhai Trivedi: *Head Pose Estimation in Computer Vision : A Survey.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 31 :607–626, 2009, ISSN 0162-8828.

[MDWE04] P. Marziliano, F. Dufaux, S. Winkler et T. Ebrahimi: *Perceptual Blur and Ringing Metrics : Application to JPEG2000.* Signal Processing : Image Communication, 19(2) :163–172, 2004.

[Mel09] Anouar Mellakh: *Reconnaissance des visages en conditions dégradées.* Thèse de doctorat, Département Electronique et Physique de l'Institut national des télécommunications, 2009.

[MK01] Aleix M. Martinez et Avinash C. Kak: *PCA versus LDA.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 23(2) :228–233, 2001. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.28.9072.

[Mog00] B. Moghaddam: *Bayesian face recognition.* Pattern Recognition, 33(11) :1771–1782, novembre 2000, ISSN 00313203. http://dx.doi.org/10.1016/S0031-3203(99)00179-X.

[Mov02] Javier R Movellan: *Tutorial on Gabor Filters.* Response, pages 1–23, 2002. http://mplab.ucsd.edu/wordpress/tutorials/gabor.pdf.

[MP97] Baback Moghaddam et Alex Pentland: *Probabilistic Visual Learning for Object Representation.* IEEE Trans. Pattern Anal. Mach. Intell., 19(7) :696–710, août 1997, ISSN 0162-8828. http://dx.doi.org/10.1109/34.598227.

[MZS+06] Bingpeng Ma, Wenchao Zhang, Shiguang Shan, Xilin Chen et Wen Gao: *Robust Head Pose Estimation Using LGBP.*

Dans *Proceedings of the 18th International Conference on Pattern Recognition - Volume 02*, ICPR '06, pages 512–515, Washington, DC, USA, 2006. IEEE Computer Society, ISBN 0-7695-2521-0.
http://dx.doi.org/10.1109/ICPR.2006.1006.

[NG99] Jeffrey Ng et Shaogang Gong: *Multi-View Face Detection and Pose Estimation Using A Composite Support Vector Machine across the View Sphere.*
Dans *In Proc. IEEE International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, pages 14–21, 1999.

[NG02] Jeffrey Ng et Shaogang Gong: *Composite support vector machines for detection of faces across views and pose estimation.*
Image and Vision Computing, 20(5-6) :359–368, avril 2002.
http://dx.doi.org/10.1016/S0262-8856(02)00008-2.

[NHT+11] Masashi Nishiyama, Abdenour Hadid, Hidenori Takeshima, Jamie Shotton, Tatsuo Kozakaya et Osamu Yamaguchi: *Facial deblur inference using subspace analysis for recognition of blurred faces.*
IEEE Transactions on Pattern Analysis and Machine Intelligence, 33(4) :838–845, 2011.
http://www.ncbi.nlm.nih.gov/pubmed/21079280.

[NTS+09] Masashi Nishiyama, Hidenori Takeshima, Jamie Shotton, Tatsuo Kozakaya et Osamu Yamaguchi: *Facial deblur inference to improve recognition of blurred faces.*
IEEE Conference on Computer Vision and Pattern Recognition (2009), pages 1115–1122, 2009.
http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5206750.

[OGX09] Javier Orozco, Shaogang Gong et Tao Xiang: *Head Pose Classification in Crowded Scenes.*
Dans *BMVC'09*, pages –1–1, 2009.

[OH08] Ville Ojansivu et Janne Heikkilä: *Blur Insensitive Texture Classification Using Local Phase Quantization.*
Dans *ICISP '08 : Proceedings of the 3rd international conference on Image and Signal Processing*, pages 236–243, Berlin, Heidelberg, 2008. Springer-Verlag.

[Ong03] Eeping Ong: *A no-reference quality metric for measuring image blur*, tome 1, pages 469–472.
Ieee, 2003.
http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1224741.

[OPH96] T. Ojala, M. Pietikainen et D. Harwood: *A Comparative Study of Texture Measures with Classification Based on Feature Distributions.*

PR, 29(1) :51–59, January 1996.

[OPM02] T. Ojala, M. Pietikainen et T. Maenpaa: *Multiresolution gray-scale and rotation invariant texture classification with local binary patterns.* Pattern Analysis and Machine Intelligence, IEEE Transactions on, 24(7) :971–987, July 2002.

[PEWF08] Simon J.D. Prince, James H. Elder, Jonathan Warrell et Fatima M. Felisberti: *Tied Factor Analysis for Face Recognition across Large Pose Differences.* IEEE Transactions on Pattern Analysis and Machine Intelligence, 30 :970–984, 2008, ISSN 0162-8828.

[PH11] Matti Pietikäinen et Janne Heikkilä: *Image and Video Description with Local Binary Pattern, Variants*, June 2011. `http://www.ee.oulu.fi/research/imag/mvg/files/pdf/CVPR-tutorial-final.pdf`.

[PLR+04] Feng Pan, Xiao Lin, Susanto Rahardja, Weisi Lin, Ee Ping Ong, Susu Yao, Zhongkang Lu et Xiaokang Yang: *A locally-adaptive algorithm for measuring blocking artifacts in images and videos.* Dans *ISCAS (3)*, pages 925–928, 2004.

[PMG05] Cristian Perra, Francesco Massidda et Daniele D. Giusto: *Image blockiness evaluation based on Sobel operator.* Dans *ICIP (1)*, pages 389–392, 2005. `http://dblp.uni-trier.de/db/conf/icip/icip2005-1.html#PerraMG05`.

[PMRR97] P.J. Phillips, Hyeonjoon Moon, P. Rauss et S.A. Rizvi: *The FERET evaluation methodology for face-recognition algorithms.* Dans *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, pages 137–143, June 1997.

[PMS94] Alex Pentland, Baback Moghaddam et Thad Starner: *View-Based and Modular Eigenspaces for Face Recognition.* Dans *IEEE INTERNATIONAL CONFERENCE ON COMPUTER VISION & PATTERN RECOGNITION*, 1994.

[RB08] B. Rossion et A. Boremanse: *Nonlinear relationship between holistic processing of individual faces and picture-plane rotation : Evidence from the face composite illusion.* J Vis, 8(4) :3.1–13, 2008.

[RBK98] Henry A. Rowley, Shumeet Baluja et Takeo Kanade: *Neural Network-Based Face Detection.* IEEE Transactions On Pattern Analysis and Machine intelligence, 20 :23–38, 1998.

[RBV02] Sami Romdhani, Volker Blanz et Thomas Vetter: *Face identification by fitting a 3D morphable model using linear shape and texture error functions.*

Dans *in European Conference on Computer Vision*, pages 3–19, 2002.

[RR06] N. M. Robertson et I. D. Reid: *Estimating Gaze Direction from Low-Resolution Faces in Video.*
Dans *Proceedings of the 2006 European Conference on Computer Vision*, 2006.

[RS00] Sam T. Roweis et Lawrence K. Saul: *Nonlinear dimensionality reduction by locally linear embedding.*
SCIENCE, 290 :2323–2326, 2000.

[RYS04] Bisser Raytchev, Ikushi Yoda et Katsuhiko Sakaue: *Head Pose Estimation by Nonlinear Manifold Learning.*
Dans *Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 4 - Volume 04*, ICPR '04, pages 462–466, Washington, DC, USA, 2004. IEEE Computer Society, ISBN 0-7695-2128-2.
`http://dx.doi.org/10.1109/ICPR.2004.432`.

[SB02] Sujith Srinivasan et Kim L. Boyer: *Head Pose Estimation Using View Based Eigenspaces.*
Pattern Recognition, International Conference on, 4 :40302, 2002, ISSN 1051-4651.

[SB06] Linlin Shen et Li Bai: *A review on Gabor wavelets for face recognition.*
Pattern Anal. Appl., 9(2) :273–292, 2006.

[SCK11] Karthik Sankaranarayanan, Ming Ching Chang et Nils Krahnstoever: *Tracking gaze direction from far-field surveillance cameras.*
Dans *Proceedings of the 2011 IEEE Workshop on Applications of Computer Vision (WACV)*, WACV '11, pages 519–526, Washington, DC, USA, 2011. IEEE Computer Society, ISBN 978-1-4244-9496-5.
`http://dx.doi.org/10.1109/WACV.2011.5711548`.

[SGjO99] Jamie Sherrah Shaogang, Shaogang Gong et Eng jon Ong: *Understanding Pose Discrimination in Similarity Space.*
Dans *10 th British Machine Vison Conference*, pages 523–532. BMVA Press, 1999.

[SGO00] J. Sherrah, S. Gong et E. J. Ong: *Face distributions in similarity space under varying head pose*, 2000.

[Shl05] Jonathon Shlens: *A Tutorial on Principal Component Analysis*, décembre 2005.
`http://www.snl.salk.edu/~shlens/pub/notes/pca.pdf`.

[SI00] I. Stainvas et N. Intrator: *Blurred face recognition via a hybrid network architecture.*
Dans *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, tome 2, pages 805–808, Barcelona, Spain, 2000.

[SMI00] Inna Stainvas, Amiram Moshaiov et Nathan Intrator: *Improving Classification via Reconstruction*, 2000.

[SSM98] Bernhard Scholkopf, Alexander Smola et Klaus Robert Muller: *Nonlinear Component Analysis as a Kernel Eigenvalue Problem.*
Neural Comp., 10(5) :1299–1319, juillet 1998.
http://sml.nicta.com.au/~smola/papers/SchSmoMul98.pdf.

[TChZFZ06] Xiaoyang Tan, Songcan Chen et Zhi hua Zhou Fuyan Zhang: *Face recognition from a single image per person : A survey.*
Pattern Recognition, 39 :1725–1745, 2006.

[Tho80] P. Thompson: *Margaret Thatcher : a new illusion.*
Perception, 9(4) :483–484, 1980, ISSN 0301-0066.
http://view.ncbi.nlm.nih.gov/pubmed/6999452.

[THT06] Jilin Tu, Thomas Huang et Hai Tao: *Accurate Head Pose Tracking in Low Resolution Video.*
Dans *Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*, FGR '06, pages 573–578, Washington, DC, USA, 2006. IEEE Computer Society, ISBN 0-7695-2503-2.
http://dx.doi.org/10.1109/FGR.2006.19.

[TP91] M. A. Turk et A. P. Pentland: *Face recognition using eigenfaces.*
Dans *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 586–591. IEEE Comput. Sco. Press, 1991, ISBN 0-8186-2148-6.
http://dx.doi.org/10.1109/CVPR.1991.139758.

[TSL00] Joshua B. Tenenbaum, Vin Silva et John C. Langford: *A Global Geometric Framework for Nonlinear Dimensionality Reduction.*
Science, 290(5500) :2319–2323, décembre 2000, ISSN 00368075.
http://dx.doi.org/10.1126/science.290.5500.2319.

[TT07] Xiaoyang Tan et Bill Triggs: *Enhanced Local Texture Feature Sets for Face Recognition under Difficult Lighting Conditions.*
Dans *Analysis and Modelling of Faces and Gestures*, tome 4778 de *LNCS*, pages 168–182. Springer, oct 2007.
http://lear.inrialpes.fr/pubs/2007/TT07.

[VC09a] Ngoc Son Vu et Alice Caplier: *Efficient statistical face recognition across pose using local binary patterns and Gabor wavelets.*
Dans *Proceedings of the 3rd IEEE international conference on Biometrics : Theory, applications and systems*, BTAS'09, pages 44–48, Piscataway, NJ, USA, 2009. IEEE Press, ISBN 978-1-4244-5019-0.
http://dl.acm.org/citation.cfm?id=1736406.1736413.

[VC09b] Ngoc Son Vu et Alice Caplier: *Illumination-robust face recognition using retina modeling.*
Dans *2009 16th IEEE International Conference on Image Processing (ICIP)*, pages 3289–3292. IEEE, novembre 2009, ISBN 978-1-4244-5653-6.
http://dx.doi.org/10.1109/ICIP.2009.5413963.

[VC10] Ngoc Son Vu et Alice Caplier: *Face Recognition with Patterns of Oriented Edge Magnitudes.*
Dans Kostas Daniilidis, Petros Maragos et Nikos Paragios (rédacteurs) : *Computer Vision ECCV 2010*, tome 6311 de *Lecture Notes in Computer Science*, pages 313–326. Springer Berlin / Heidelberg, 2010, ISBN 978-3-642-15548-2.
http://dx.doi.org/10.1007/978-3-642-15549-9_23.

[VJ04] Paul Viola et Michael J. Jones: *Robust Real-Time Face Detection.*
Int. J. Comput. Vision, 57 :137–154, May 2004, ISSN 0920-5691.
http://portal.acm.org/citation.cfm?id=966432.966458.

[Vu10] Ngoc Son Vu: *Contributions à la reconnaissance de visages à partir d'une seule image et dans un contexte non-contrôlé.*
Thèse de doctorat, Institut polytechnique de Grenoble, 2010.

[WBE00] Zhou Wang, Alan C Bovik et Brian L Evans: *Blind measurement of blocking artifacts in images*, tome 3, pages 981–984.
Ieee, 2000.
http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=899622.

[WFKvdM99] Laurenz Wiskott, Jean Marc Fellous, Norbert Krüger et Christoph von der Malsburg: *Face recognition by elastic bunch graph matching*, pages 355–398.
CRC Press, Inc., Boca Raton, FL, USA, 1999, ISBN 0-8493-2055-0.
http://portal.acm.org/citation.cfm?id=320684.320713.

[WPV05] Jie Wang, K. N. Plataniotis et A. N. Venetsanopoulos: *Selecting discriminant eigenfaces for face recognition.*
Pattern Recogn. Lett., 26 :1470–1482, July 2005, ISSN 0167-8655.
http://dx.doi.org/10.1016/j.patrec.2004.11.029.

[WS07] Jian Gang Wang et Eric Sung: *EM enhancement of 3D head pose estimated by point at infinity.*
Image Vision Comput., 25 :1864–1874, December 2007, ISSN 0262-8856.
http://dl.acm.org/citation.cfm?id=1287837.1287945.

[WSB02] Zhou Wang, Hamid R. Sheikh et Alan C. Bovik: *No-Reference Perceptual Quality Assessment of JPEG Compressed Images.*
Dans *Proceedings of IEEE 2002 International Conferencing on Image Processing*, pages 477–480, 2002.

[WWLC00] H R Wilson, F Wilkinson, L M Lin et M Castillo: *Perception of head orientation.*
Vision Research, 40(5) :459–472, 2000.
http://www.ncbi.nlm.nih.gov/pubmed/17853198.

[WZ02] Jianxin Wu et Zhi Hua Zhou: *Face recognition with one training image per person.*
Pattern Recognition Letters, 23(14) :1711–1719, 2002.

[Yan02] Ming H. Yang: *Kernel Eigenfaces vs. Kernel Fisherfaces : Face Recognition Using Kernel Methods.*
Dans *FGR '02 : Proceedings of the Fifth IEEE International Conference on Automatic Face and Gesture Recognition*, pages 215+, Washington, DC, USA, 2002. IEEE Computer Society, ISBN 0-7695-1602-5.
http://portal.acm.org/citation.cfm?id=875432.

[YHH87] A. W. Young, D. Hellawell et D. C. Hay: *Configurational information in face perception.*
Perception, 16(6) :747–759, 1987.
http://dx.doi.org/10.1068/p160747.

[ZCPR03] W. Zhao, R. Chellappa, P. J. Phillips et A. Rosenfeld: *Face recognition : A literature survey.*
ACM Comput. Surv., 35(4) :399–458, 2003.

[ZG09] Xiaozheng Zhang et Yongsheng Gao: *Face recognition across pose : A review.*
Pattern Recogn., 42 :2876–2896, November 2009, ISSN 0031-3203.
http://dl.acm.org/citation.cfm?id=1563046.1563061.

[Zha02] *Discriminant analysis of principal components for face recognition*, août 2002.
http://dx.doi.org/10.1109/AFGR.1998.670971.

[ZHLH06] Zhenqiu Zhang, Yuxiao Hu, Ming Liu et Thomas Huang: *T. : Head Pose Estimation in Seminar Rooms Using Multi View Face Detectors.*
Dans *Proc. CLEAR Workshop, LNCS*, pages 299–304, 2006.

[ZSCG07] Baochang Zhang, Shiguang Shan, Xilin Chen et Wen Gao: *Histogram of Gabor phase patterns (HGPP) : a novel object representation approach for face recognition.*
IEEE Transactions on Image Processing, 16(1) :57–68, 2007.
http://www.ncbi.nlm.nih.gov/pubmed/17283765.

[ZSG+05] Wenchao Zhang, Shiguang Shan, Wen Gao, Xilin Chen et Hongming Zhang: *Local Gabor Binary Pattern Histogram Sequence (LGBPHS) : A Novel Non-Statistical Model for Face Representation and Recognition.*
Dans *Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1 - Volume 01*, pages 786–791, Washington, DC, USA, 2005. IEEE Computer Society, ISBN 0-7695-2334-X-01.
http://dl.acm.org/citation.cfm?id=1097114.1097724.