Machine Observation of the Direction of Human Visual Focus of Attention

Nicolas Gourier

▶ To cite this version:

HAL Id: tel-00150756
https://theses.hal.science/tel-00150756

Submitted on 31 May 2007

INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

<u>THÈSE</u>

pour obtenir le grade de

DOCTEUR DE L'INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE

Spécialité : Imagerie, Vision et Robotique
Ecole Doctorale : Mathématiques, Sciences et Technologie de l'Information

présentée et soutenue publiquement
par

Nicolas GOURIER

le 19 octobre 2006

MACHINE OBSERVATION OF THE DIRECTION OF HUMAN VISUAL FOCUS OF ATTENTION

Directeur de thèse : M. James L. CROWLEY

JURY

Mme Catherine GARBAY, Présidente
M. Roberto CIPOLLA, Rapporteur
Mme Monique THONNAT, Rapporteuse
M. James L. CROWLEY, Directeur
Mlle Daniela HALL, Codirectrice
M. Josep R. CASAS, Examinateur

Thèse préparée dans le laboratoire GRAVIR – IMAG au sein du projet PRIMA
INRIA Rhône-Alpes, 655 av. de l'Europe, 38334 Saint-Ismier, France.

Abstract

People often look at objects and people with which they are likely to interact. The first step for computer systems to adapt to the user and to improve interaction and with people is to locate where they are, and especially the location of their faces on the image. The next step is to track their focus of attention. For this reason, we are interested in techniques for estimating and tracking gaze of people, and in particular the head pose.

This thesis proposes a fully automatic approach for head pose estimation independant of the person identity using low resolution images acquired in unconstrained imaging conditions. The developed method is demonstrated and evaluated using a densly sampled face image database. We propose a new coarse-to-fine approach that uses both global and local appearance to estimate head orientation. This method is fast, easy to implement, robust to partial occlusion, uses no heuristiques and can be adapted to other deformable objects. Face region images are normalized in size and slant by a robust face tracker. The resulting normalized imagettes are projected onto a linear auto-associative memory learned using the Widrow-Hoff rule. Linear auto-associative memories require very few parameters and offer the advantage that no cells in hidden layers have to be defined and class prototypes can be saved and recovered for all kinds of applications. A coarse estimation of the head orientation on known and unknown subjects is obtained by searching the best prototype which matches the current image.

We search for salient facial features relevant for each head pose. Feature points are locally described by Gaussian receptive fields normalized at intrinsic scale. These descriptors have interesting properties and are less expensive than Gabor wavelets. Salient facial regions found by Gaussian receptive fields motivate the construction of a model graph for each pose. Each node of the graph can be displaced localy according to its saliency in the image. Linear auto-associative memories deliver a coarse estimation of the pose. We search among the coarse pose neighbors the model graph which obtains the best match. The pose associated with its salient grid graph is selected as the head pose of the person on the image. This method does not use any heuristics, manual annotation or prior knowledge on the face and can be adapted to estimate the pose of configuration of other deformable objects.

Keywords: Head pose estimation, focus of attention, real-time face tracking, linear auto-associative memory, Gaussian derivative receptive fields, feature saliency, grid graphs.

Résumé

Les personnes dirigent souvent leur attention vers les objets avec lesquels ils interagissent. Une première étape que doivent franchir les systèmes informatiques pour s'adapter aux utilisateurs et améliorer leurs interactions avec eux est de localiser leur emplacement, et en particulier la position de leur tête dans l'image. L'étape suivante est de suivre leur foyer d'attention. C'est pourquoi nous nous intéressons aux techniques permettant d'estimer et de suivre le regard des utilisateurs, et en particulier l'orientation de leur tête.

Cette thèse présente une approche complètement automatique et indépendante de l'identité de la personne pour estimer la pose d'un visage à partir d'images basse résolution sous conditions non contraintes. La méthode developpée ici est évaluée et validée avec une base de données d'images échantillonnée. Nous proposons une nouvelle approche à 2 niveaux qui utilise les apparences globales et locales pour estimer l'orientation de la tête. Cette méthode est simple, facile à implémenter et robuste à l'occlusion partielle. Les images de visage sont normalisées en taille dans des images de faible résolution à l'aide d'un algorithme de suivi de visage. Ces imagettes sont ensuite projetées dans des mémoires autoassociatives et entraînées par la règle d'apprentissage de Widrow-Hoff. Les mémoires autoassociatives ne nécessitent que peu de paramètres et évitent l'usage de couches cachées, ce qui permet la sauvegarde et le chargement de prototypes de poses du visage humain. Nous obtenons une première estimation de l'orientation de la tête sur des sujets connus et inconnus.

Nous cherchons ensuite dans l'image les traits faciaux saillants du visage pertinents pour chaque pose. Ces traits sont décrits par des champs réceptifs gaussiens normalisés à l'échelle intrinsèque. Ces descripteurs ont des propriétés intéressantes et sont moins coûteux que les ondelettes de Gabor. Les traits saillants du visage détectés par les champs réceptifs gaussiens motivent la construction d'un modèle de graphe pour chaque pose. Chaque nœud du graphe peut être déplacé localement en fonction de la saillance du point facial qu'il représente. Nous recherchons parmi les poses voisines de celle trouvée par les mémoires autoassociatives le graphe qui correspond le mieux à l'image de test. La pose correspondante est sélectionnée comme la pose du visage de la personne sur l'image. Cette méthode n'utilise pas d'heuristique, d'annotation manuelle ou de connaissances préalables sur le visage et peut être adaptée pour estimer la pose d'autres objets déformables.

Mots clés : estimation de l'orientation de la tête, foyer d'attention, suivi du visage en temps réel, mémoires linéaires autoassociatives, champs réceptifs de dérivées gaussiennes, régions saillantes, graphes.

Acknowledgements

I would like to thank all people who contributed in all manners to the achievement of my thesis work.

First of all, my thanks go to my supervisor Prof. James L. Crowley for his well advised discussions and his motivation. I also would like to thank Dr. Daniela Hall for her ideas and her patience. I am grateful to Roberto Cipolla, Monique Thonnat, Catherine Garbay and Josep R. Casas for their interest in my work and for being members of my jury.

I would like to thank Jérôme Maisonnasse for his criticism and for being a fun officemate, Olivier Riff who introduced me to the PRIMA group, Alban Caparossi for his collaboration, Augustin Lux for his help and open discussions, and Matthieu and Marina for sharing the office. Thanks to all PRIMA members, Patrick, Dominique, Alba, Stan, Matthieu, Julien, Rémi, Hai, Suphot, Olivier, Sonia, Oliver and Caroline for the cool atmosphere in the group. It was a pleasure to work at the INRIA Rhône-Alpes in Grenoble.

My special thanks go to Véronique and Guillaume for their special collaboration, Jean-Baptiste for his jokes and to all my friends of the ML for their support.

I would also like to thank all people who posed for the database and all people who participated to the experiment.

Finally, I would like to thank my parents and my grandmother for their unconditional support and love.

Contents

Première partie

Observation de la direction du foyer visuel d’attention par ordinateur

-

Résumé français

Chapitre 1

Introduction

La plupart des ordinateurs modernes sont autistes. Peu de nouvelles technologies existent pour recenser les interactions sociales entre des personnes et entre une personne et une machine. En conséquence, les systèmes artificiels distraient souvent les utilisateurs avec des actions inappropriées et n'ont pas ou peu de capacités à utiliser les interactions humaines pour corriger leur comportement.

Un aspect important des interactions sociales est la capacité à observer l'attention humaine. Généralement, les personnes localisent le foyer d'attention des personnes en observant leurs visages et leurs regards. En majeure partie, l'intérêt et l'attention d'une personne peuvent être estimés à partir de l'orientation de sa tête.

Dans cette thèse, nous nous intéressons au problème de l'estimation de l'orientation, ou pose, de la tête sur des images non contraintes. La pose de la tête est déterminée par trois angles : l'inclinaison par rapport au corps (slant), l'inclinaison horizontale (pan) et l'inclinaison verticale (tilt). L'angle slant varie autour de l'axe longitudinal. L'angle tilt varie autour de l'axe latéral, quand une personne regarde de bas en haut. Cet angle est le plus difficile à estimer. L'angle pan varie autour de l'axe vertical, quand une personne tourne sa tête de gauche à droite. Notre objectif est d'estimer ces trois angles, ce qui servira de première base à l'estimation de l'attention.

Beaucoup de techniques d'estimation de regard et de pose de la tête présentes dans la littérature utilisent des équipements spécifiques, comme l'illumination infrarouge, l'électro-oculographie, les casques portables ou des lentilles de contact spécifiques [59, 167, 33]. Des systèmes utilisant des caméras actives ou la vision stéréo sont disponibles dans le commerce [162, 96, 120]. Bien que de telles techniques soient très précises, elles sont généralement chères et trop intrusives pour beaucoup d'applications. Les systèmes basés sur la vision par ordinateur présentent un choix plus accessible et moins intrusif.

Notre but est de proposer une méthode non intrusive et qui ne nécessite pas d'équipement spécifique pour estimer l'orientation de la tête. En particulier, nous nous intéressons aux technologies robustes au changement d'identité sous des conditions d'images non contraintes. Les humains peuvent estimer grossièrement la pose d'un objet à partir d'une image. En outre, l'estimation de l’orientation de la tête à partir d’une image est la base pour une estimation plus précise à partir de plusieurs images.

Les approches pour estimer l’orientation de la tête à partir d’une simple image peuvent être regroupées en 4 familles : les approches géométriques 2D, les approches géométriques 3D, les approches par transformation faciale et les approches par classifieurs. Les approches géométriques 2D utilisent certains traits du visage pour trouver des correspondances et estimer ainsi l’orientation. Ces méthodes sont précises mais nécessitent une bonne résolution de l’image du visage et voient leurs performances se dégrader sur des mouvements de tête amples. Les approches géométriques 3D appliquent un modèle 3D de la tête sur l’image pour retrouver la pose. Ces techniques sont encore plus précises, mais requièrent plus de temps de calcul, une bonne résolution ainsi qu’une forte connaissance préalable du visage. Les approches par transformation faciale utilisent certaines propriétés faciales pour obtenir une estimation de la pose de la tête. De telles méthodes sont simples à mettre en œuvre, mais sont parfois instables et non robustes à l’identité. Les approches par classifieurs résolvent le problème en cherchant la meilleure correspondance avec l’image courante et un modèle préalablement appris. Ces méthodes sont très rapides, mais ne peuvent délivrer qu’une estimation grossière et l’utilisateur n’a pas de retour d’information si le système échoue. Nous développons une approche hybride globale et locale à 2 niveaux pour estimer l’orientation de la tête dont les performances sont comparables aux performances humaines.

1.1 Estimation de la pose de la tête par apparences globale et locale

Dans cette thèse, nous proposons une approche complètement automatique d’estimation de pose de la tête indépendante de l’identité sur des images prises dans des conditions non contraintes. Cette approche combine les avantages des approches globales qui utilisent l’apparence entière de l’image du visage pour la classification et les approches locales qui utilisent les informations contenues dans les voisinages de pixels et leurs relations dans l’image, sans utiliser d’heuristique ni de connaissance préalable sur le visage. Nous présentons un système d’estimation de l’orientation de la tête à 2 niveaux basé sur les mémoires autoassociatives linéaires et les graphes de champs réceptifs gaussiens. Notre méthode marche sur des images non alignées comme dans les conditions réelles et sa performance est comparable aux performances humaines.

Pour mesurer efficacement la performance d’un algorithme d’estimation de pose de la tête, il est nécessaire de le tester sur une base de données représentative. Dans la littérature, les méthodes différentes sont souvent testées sur des bases de données différentes, ce qui rend les comparaisons difficiles. Une base de données représentative doit contenir un nombre suffisant d’orientations pour observer le comportement de l’algorithme sur chaque pose. Cette même base de données doit être symétrique et suffisamment échantillonnée. Si une méthode marche bien sur la plupart des angles, elle peut être adaptée au suivi de pose de la tête en temps réel et en conditions réelles, dans lesquelles l'orientation de la tête n'est pas discrète mais continue.

Dans nos expériences, nous utilisons la Pointing 2004 Head Pose Image Database [39], une base de données échantillonnée de 15 en 15 degrés couvrant une demi-sphère d'orientations, soit des angles pan et tilt variant de -90 à +90 degrés. Cette base contient 15 sujets. Pour chaque sujet, il y a 2 séries de 93 images de pose. L'apprentissage et le test peuvent être faits soit sur les sujets connus en effectuant une validation croisée sur les séries, soit sur les sujets inconnus en effectuant un algorithme Jack-Knife sur les sujets.

Les capacités humaines pour estimer l'orientation de la tête sont largement inconnues. Nous ne savons pas si les humains ont une aptitude naturelle à estimer la pose de la tête à partir d'une simple image ou s'ils doivent être entraînés à cette tâche à partir d'images d'exemple. De plus, nous ne connaissons pas l'exactitude avec laquelle une personne peut estimer les angles pan et tilt. Dans ses études, Kersten [65] montre que les poses face et profil sont utilisées comme des poses clés par le cerveau humain. Comme référence, nous avons évalué les performances d'un groupe de personnes à l'estimation de l'orientation de la tête sur une partie de la Pointing'04 Head Pose Image Database. Ces expériences montrent que notre algorithme obtient des résultats similaires à ceux obtenus par le groupe de personnes.

Dans notre méthode, une première estimation de la pose est obtenue en cherchant la meilleure mémoire autoassociative linéaire correspondant à l'image du visage. Nous combinons cette estimation avec une autre méthode basée sur les régions saillantes du visage pertinentes poour chaque pose. Les régions saillantes sont décrites localement par des champs réceptifs gaussiens normalisés à leurs échelles intrinsèques, données par le premier maximum local du laplacien normalisé. Ces descripteurs ont des propriétés intéressantes et sont moins coûteux à calculer que les ondelettes de Gabor. Les régions saillantes détectées de cette façon permettent la construction d'un modèle de graphe pour chaque pose. Chaque nœud du graphe peut être déplacé localement en fonction de sa saillance et est annoté par une densité de probabilité de vecteurs de champs réceptifs gaussiens normalisés et clusterisés hiérarchiquement, pour représenter les différents aspects que peuvent avoir un même trait du visage selon différentes identités. Les mémoires autoassociatives linéaires donnent une première estimation de la pose. Ce résultat est raffiné en cherchant parmi les poses voisines le meilleur modèle de graphe correspondant. La pose associée au modèle de graphe est sélectionnée comme la pose du visage de la personne.

1.2 Contributions principales de cette thèse

Nos expériences montrent que les humains réussissent à bien reconnaître les poses face et profil, mais moins les poses intermédiaires. Le groupe de personnes a effectué une erreur moyenne de 11.85^o en pan et 11.04^o en tilt. L'erreur minimale se trouve pour la pose 0 degré, ce qui correspond à la vue de face. L'angle pan semble plus naturel à estimer. Ces résultats suggèrent que le système visuel humain utilise face et profil comme des poses clés, comme stipulé dans [65].

Dans notre méthode, la région de l'image correspondant au visage est normalisée dans une image de petite résolution en utilisant un système de suivi de visage. Les mémoires auto-associatives linéaires sont utilisées pour apprendre des prototypes d'orientations de la tête. Ces mémoires sont simples à construire, ne requièrent que peu de paramètres et sont adaptées pour l'estimation de la pose du visage sur des sujets connus et inconnus. Les prototypes peuvent être appris en utilisant un ou deux axes. Avec une erreur moyenne de moins de 10^o en pan et en tilt pour des sujets connus, notre méthode est plus performante que les réseaux de neurones [152], l'Analyse par Composantes Principales et les modèles de tenseurs [145]. Nous obtenons une erreur moyenne de 10^o en pan et 16^o en tilt sur des sujets inconnus. Apprendre les angles pan et tilt ensemble n'améliore pas significativement les résulats. Nous apprenons donc ces angles séparément, ce qui réduit le nombre de prototypes à utiliser. Ces résultats sont obtenus sur des images non alignées. Les prototypes de poses du visage peuvent être sauvegardés et chargés ultérieurement pour d'autres applications. Notre algorithme de première estimation de la pose fonctionne à 15 images par seconde, ce qui est suffisant pour des applications vidéo telles que les interactions homme-machine, la vidéosurveillance et les environnements intelligents.

Cette première estimation est raffinée en décrivant les images du visage par des champs réceptifs gaussiens normalisés à leurs échelles intrinsèques. Les dérivées gaussiennes décrivent l'apparence de voisinages de pixels et présentent un moyen efficace pour détecter les traits du visage indépendamment de leur taille et de leur illumination. De plus, elles ont des propriétés d'invariance intéressantes. Les images de visage sont ainsi décrites par des vecteurs de faible dimension. Les régions saillantes du visage sont découvertes en analysant les régions qui partagent une apparence similaire sur un rayon limité. Nous trouvons que les principaux traits saillants du visage sont : les yeux, le nez, la bouche et le contour du visage. Ces résultats ressemblent aux traits faciaux regardés par les humains selon les études de Yarbus [165].

Les graphes de champs réceptifs gaussiens améliorent l'estimation de la pose obtenue en première estimation. La structure de graphe décrit à la fois l'apparence des voisinages de pixels et leurs relations géométriques dans l'image. Les résultats sont meilleurs en effectuant un clustering hiérarchique en chaque nœud du graphe. Les graphes recouvrant la totalité de l'image du visage sont plus performants que ceux ne recouvrant qu'une partie du visage. Plus grande est la portion d'image recouverte, plus importantes sont les relations géométriques. De plus, paramétrer le déplacement local maximal d'un nœud en fonction de sa saillance résulte en une meilleure estimation que fixer un même déplacement local pour chaque nœud. Un nœud placé sur un trait saillant du visage représente un point pertinent pour la pose considérée et ne doit pas trop se déplacer de son emplacement initial. Au contraire, un nœud placé dans une région peu saillante ne représente pas de point pertinent pour la pose et peut bouger. En utilisant cette méthode, nous obtenons un système d'estimation de la pose de la tête avec une exactitude de 10^o en pan et 12^o en tilt sur des sujets inconnus. Cet algorithme ne requiert pas d'heuristique, d'annotation manuelle ou de connaissance préalable sur le visage et peut être adapté pour estimer l'orientation ou la configuration d'autres objets déformables.

L'estimation de pose du visage est testée sur des séquences vidéo de la IST CHIL Pointing

Database. Le contexte temporel offre un gain en temps de calcul considérable. La pose du visage sur l'image suivante se trouve dans le voisinage de la pose courante. Nous avons obtenu une erreur moyenne de 22.5^o en pan. Les sujets sont différents de ceux de la base de données Pointing'04. L'estimation de l'orientation de la tête peut également servir d'entrée pour des systèmes attentionnels [85].

Chapitre 2

Contenu de la thèse

L'attention visuelle contribue plus que l'attention auditive dans l'attention humaine [129]. De plus, plusieurs études rapportent que le regard fournit des informations importantes sur le foyer d'attention [130, 75]. La direction du regard est déterminée par l'orientation de la tête et la position de la pupille sur l'œil. Durant un regard rapide, il n'y a presque pas de rotation de la tête. Les yeux peuvent mouvoir leur orbite à une vitesse allant jusqu'à 500 degrés par seconde. Cependant, pour un regard soutenu, les muscles des yeux ont besoin d'effort pour se maintenir désaxés. La rotation de la tête soulage alors cet effort. C'est pourquoi la plupart des études montrent que l'orientation contribue généralement plus que la position de la pupille sur l'œil à l'attention visuelle. Dans ses études, Stiefelhagen [138, 130] a trouvé que les gens tournent la tête plus souvent que les yeux dans 69 % des cas et la direction de la tête est la même que celle des yeux dans 89 % en situation de meeting. En outre, détecter les pupilles sur une image requiert une haute résolution de l'image du visage, et les yeux peuvent cligner, ce qui les rend plus difficiles à détecter. C'est pourquoi nous nous intéressons à l'estimation de l'orientation de la tête.

2.1 Approches pour estimer l'orientation de la tête

Le but de cette étude est de déterminer l'orientation, ou pose, de la tête sur des images non contraintes. La pose de la tête est déterminée par trois angles : l'inclinaison par rapport au corps (slant), l'inclinaison horizontale (pan) et l'inclinaison verticale (tilt). Ces trois angles sont illustrés sur la figure 2.1. L'angle slant varie autour de l'axe longitudinal. L'angle tilt varie autour de l'axe latéral, quand une personne regarde de bas en haut. Cet angle est le plus difficile à estimer. L'angle pan varie autour de l'axe vertical, quand une personne tourne sa tête de gauche à droite. Ces trois angles recouvrent complètement les mouvements de la tête.

Beaucoup de techniques d'estimation de regard et de pose de la tête présentes dans la littérature utilisent des équipements spécifiques, comme l'illumination infrarouge, l'électro-oculographie, les casques portables ou des lentilles de contact spécifiques [59, 167, 33]. Des

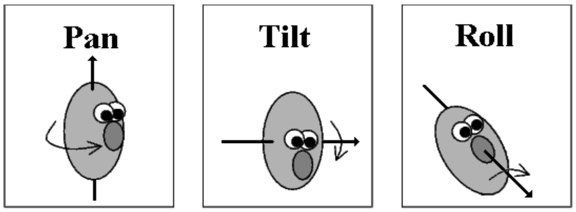

FIG. 2.1 – Les trois angles de rotation de la tête [25].

systèmes utilisant des caméras actives ou la vision stéréo sont disponibles dans le commerce [162, 96, 120]. Bien que de telles techniques soient très précises, elles sont généralement chères et trop intrusives pour beaucoup d'applications. Les systèmes basés sur la Vision par Prdinateur présentent un choix plus accessible et moins intrusif. Les humains peuvent fournir une estimation de la pose à partir d'une simple image. De plus, une bonne estimation de la pose du visage peut améliorer l'estimation de la pose à partir de plusieurs images.

L'estimation de l'orientation de la tête possède beaucoup d'applications dans des domaines variés, mais est un problème difficile et se heurte à certains obstacles. Contrairement à la plupart des problèmes en Vision par Ordinateur, il n'y a pas de cadre de travail unifié pour cette tâche. Presque tous les auteurs traitant du sujet utilisent leur propre cadre de travail et leurs propres métriques. Le premier aspect important pour un système d'estimation de la pose du visage est la résolution minimale à laquelle il peut fonctionner. Certains algorithmes ne peuvent marcher qu'à haute résolution (500x500 pixels), tandis que d'autres peuvent fonctionner avec des images de très petite résolution (32x32 pixels). Ceci nous mène à un autre aspect du problème, les mesures de performance. Il n'y a pas de métriques communes pour la tâche d'estimation de la pose. De plus, la façon dont la précision ou l'erreur moyenne sont calculées n'est pas toujours explicite dans la littérature. De même, la séparation entre les images utilisées pour l'apprentissage et le test n'est pas toujours claire. L'estimation de l'orientation de la tête diffère de l'estimation de l'orientation d'un objet en ce que la tête est déformable et change avec l'identité de la personne. Les variations de couleur de peau, des cheveux, des joues et des autres caractéristiques faciales rendent l'estimation de la pose du visage difficilement robuste aux changements d'identité. Ce problème est simplifié quand le système est conçu pour un utilisateur particulier. Cette remarque nous mène au dernier aspect important du problème : le choix de la base de données. Une base de données fiable pour l'estimation de la pose devrait couvrir un certain nombre d'angles et être bien échantillonnée pour permettre de voir le comportement d'un algorithme sur les différentes poses. Si un système fonctionne correctement pour la plupart des angles, il peut être adapté pour suivre le mouvement de la tête sur des séquences vidéo. Enfin, quand une base de données est employée, nous devons savoir quelles parties sont utilisées pour l'apprentissage et pour le test.

Les approches pour estimer l'orientation de la tête à partir d'une simple image peuvent être regroupées en 4 familles : les approches géométriques 2D, les approches géométriques 3D, les approches par transformation faciale et les approches par classifieurs. Les approches géométriques 2D utilisent certains points du visage pour trouver des correspondances et estimer ainsi l'orientation. Les points du visage de référence sont souvent les yeux [133, 163, 134, 8, 16, 36, 37]. Si ces derniers peuvent fournir une estimation de l'angle horizontal pan, ils ne sont pas suffisants pour estimer l'angle vertical tilt. C'est pourquoi les auteurs utilisent souvent d'autres points comme la bouche [169, 58, 126, 26, 47, 155], les sourcils [103], le nez [48, 17] ou même les trous du nez [142, 143, 4]. Un modèle plus complet utilisant 6 points faciaux a été proposé par Gee & Cipolla [31, 32]. Utiliser plus de points permet d'obtenir une estimation de la pose plus fiable, mais la position de ces points sur le visage peut changer d'une personne à une autre et certains peuvent ne pas être détectés sous des angles de tête trop grands. Ces méthodes sont précises mais nécessitent une bonne résolution de l'image du visage, dépendent de l'algorithme de détection de caractéristiques faciales et voient leurs performances se dégrader sur des mouvements de tête amples.

Les approches géométriques 3D appliquent un modèle 3D de la tête sur l'image pour retrouver la pose. La première technique de correspondance a été proposée par Huttenlocher [55], et améliorée ensuite par Azarbayejani et al. [2] pour suivre le mouvement des objets. Sa performance a augmenté avec l'utilisation de l'algorithme EM avec moindres carrés [15], le flux optique [88] ou l'utilisation de texture [111]. Cependant, le modèle 3D de visage est souvent rigide, alors que le visage humain est déformable et varié. Une méthode permettant d'apprendre un modèle de visage en ligne a été proposée par Vachetti [147]. Les approches géométriques 3D sont très précises, mais requièrent beaucoup de temps de calcul, une bonne résolution de l'image ainsi qu'une forte connaissance préalable du visage pour fonctionner correctement.

Les approches par transformation faciale utilisent certaines propriétés faciales pour obtenir une estimation de la pose de la tête. Ces approches sont génériques et nécessitent peu de calculs. Certains auteurs utilisent la position des cheveux par rapport au visage [14, 154, 121], la dissimilitude entre les deux yeux [18, 22] ou encore l'assymétrie entre les parties gauche et droite du visage [50, 95, 25] pour estimer l'orientation de la tête. Bien que simples à mettre en œuvre, de telles méthodes sont parfois instables et non robustes aux changements d'identité.

Les approches par classifieurs résolvent le problème en cherchant la meilleure correspondance avec l'image courante et un modèle préalablement appris. Une méthode populaire de classification est l'Analyse par Composantes Principales (ACP) proposée par Turk & Pentland [146]. Elle a été utilisée pour l'estimation de la pose de tête par McKenna & Gong [106, 34, 92, 91, 35, 122]. Néanmoins, les images d'entraînement utilisées sont souvent alignées manuellement et l'ACP a tendance à être sensible à l'alignement et aux changements d'identité. D'autres méthodes utilisent des espaces propres d'ondelettes de Gabor [157, 98, 97], des Kernel ACP [77], des modèles de tenseurs, des LEA [145], des KDA [13], des SVM [52, 102, 156], des LGBP [84] ou des réseaux de neurones [116, 136, 132, 130, 135, 152, 131]. Ces méthodes ne nécessitent pas de connaissances préalables sur le visage, mais ont parfois un nombre important de paramètres à régler, et le nombre de dimensions à utiliser ou de cellules dans les couches cachées est déterminé manuellement. Ces méthodes sont rapides, mais ne peuvent délivrer qu'une estimation grossière et l'utilisateur n'a pas de retour d'information si le système échoue.

Nous voyons que les approches pour estimer l'orientation de la tête peuvent généralement

Pose	Approches Locales	Approches Globales
Faible résolution	-	+
Performance	+	-
Grands angles	-	+
Connaissance du visage	-	+
Illumination	+	-
Retour d'information	+	-
Occlusion partielle	-	+
Localisation de points faciaux	+	-

TAB. 2.1 – *Comparaison entre approches locales et globales.*

se diviser en deux catégories : les approches locales qui utilisent l'information contenue dans les voisinages de pixels et les approches globales qui utilisent l'image entière du visage. Les avantages et les inconvénients de ces deux types d'approche sont resumés dans le tableau 2.1. Augmenter la résolution de l'image du visage à traiter peut permettre une combinaison de méthodes globales et locales. À notre connaissance, peu de travaux mêlant les deux types d'approche ont été effectués. Wu & Trivedi [160] ont récemment proposé un système permettant d'obtenir une estimation de la pose avec des KDA, puis de la raffiner en utilisant des graphes élastiques. Cependant, l'utilisation de ces graphes nécessitent d'annoter les points faciaux sur toutes les images. De plus, nous ne savons pas si le choix de chaque point est pertinent pour l'estimation de la pose. Nous proposons une méthode d'estimation de l'orientation de la tête utilisant une approche hybride globale et locale ne nécessitant pas de connaissances préalables sur le visage ni d'annotation manuelle. Nous décrivons cette approche dans les sections suivantes, mais d'abord nous devons établir quelles sont les capacités humaines pour estimer la pose du visage.

2.2 Capacités humaines à estimer l'orientation de la tête

Le but de cette section est de déterminer l'exactitude qui peut être attendue d'un système d'orientation de la tête fiable pour des applications dans des environnements intelligents. Les humains estiment généralement le focus visuel d'attention sur des images à partir de l'orientation de la tête. Cependant, leurs capacités demeurent en majeure partie inconnues. Nous avons demandé à un groupe de personnes d'estimer la pose du visage sur des images. Nous avons ensuite mesuré leurs performances avec différentes métriques. Un résultat important de cette expérience est que les humains sont plus aptes à estimer l'orientation horizontale que l'orientation verticale.

2.2.1 Travaux apparentés

La base psychophysique des aptitudes humaines à estimer l'orientation de la tête demeure en majeure partie inconnue. Nous ne savons pas si les humains ont une capacité naturelle à estimer les angles de la tête ou s'ils acquièrent cette capacité avec l'expérience. À notre connaissance, il y a peu de données disponibles permettant de mesurer les compétences humaines pour cette tâche. Selon Kersten [65], les poses face et profil sont utilisées comme poses clés par le cerveau humain et sont les mieux reconnues. L'image 2.2 présente un exemple de compétition phénoménale de poses ; les poses face et profil sont activées inconsciemment par notre cerveau, mais pas les autres. Nous ne connaissons pas la performance humaine sur les poses intermédiaires et verticales.

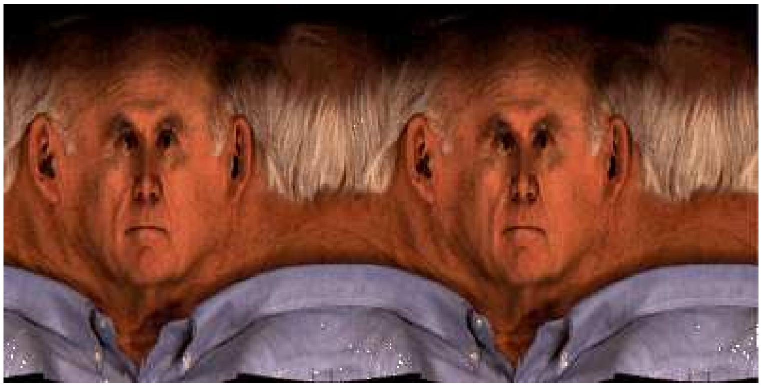

FIG. 2.2 – Projection cylindrique aplatie d'un visage humain [65]. Toutes les poses horizontales sont présentes sur cette image, mais notre cerveau a tendance à ne distinguer que les poses face et profil.

2.2.2 Protocole expérimental

Notre objectif est d'évaluer les performances des humains sur l'estimation de l'orientation de la tête aux angles pan et tilt, pour les comparer ensuite avec celles obtenues par notre système. Pour rendre possible cette comparaison, les deux performances doivent être évaluées sur la même base de données. Nous avons choisi d'utiliser des images de la base de données Pointing 2004 Head Pose Image Database [39]. Cette base de données est échantillonnée tous les 15 degrés en pan, tous les 15/30 degrés en tilt et couvre une demi-sphère de poses allant de -90 à +90 degrés sur les 2 axes. L'angle pan peut donc prendre les valeurs $(0, \pm 15, \pm 30, \pm 45, \pm 60, \pm 75, \pm 90)$, où les valeurs négatives correspondent aux poses droites et les valeurs positives correspondent aux poses gauches. L'angle tilt peut prendre les valeurs $(-90, -60, -30, -15, 0, +15, +30, +60, +90)$, où les valeurs négatives correspondent aux poses basses et les valeurs positives correspondent aux poses hautes. De plus amples détails sur cette base de données se trouvent dans l'annexe A.

Un autre but de notre expérience est de découvrir si un axe est plus pertinent qu'un autre pour les humains. Pour ce faire, nous devons être en mesure de dire si l'estimation de l'angle pan ou de l'angle tilt est naturelle ou non. Si un angle se révèle être plus naturel à estimer, cela signifie que l'axe sur lequel il évolue est plus pertinent pour les humains dans leur vie de tous les jours.

Nous avons mesuré la performance d'un groupe de 72 sujets sur l'estimation de l'orientation de la tête. Dans notre expérience, les sujets étaient répartis en 36 hommes et 36 femmes, âgés de 15 à 80 ans. On demande au sujet d'examiner une image de visage et d'entourer la réponse correspondant à son estimation de la pose. L'expérience est divisée en 2 parties effectuées dans un ordre aléatoire : une pour l'estimation de l'angle pan, une pour l'estimation de l'angle tilt. 65 images pour l'angle pan et 45 images pour l'angle tilt issues de la Pointing'04 Head Pose Image Database sont présentées au sujet pendant une durée de 7 secondes dans un ordre aléatoire, différent pour chaque sujet. Présenter les images selon un ordre aléatoire différent à chaque fois nous permet de mesurer les performances des sujets sur l'estimation de la pose du visage de façon non biaisée sur des images indépendantes, et non sur une séquence d'images prédéfinie. La durée de présentation de 7 secondes est suffisamment longue pour permettre au sujet de chercher sa réponse et suffisamment courte pour obtenir une réponse immédiate de sa part. Il y a 5 images pour chaque angle. Durant l'expérience d'estimation de l'angle pan, des symboles "+" et "-" sont indiqués à côté de l'image, comme le montrent les images de la figure 2.3, pour que le sujet ne confonde pas les poses gauches et droites.

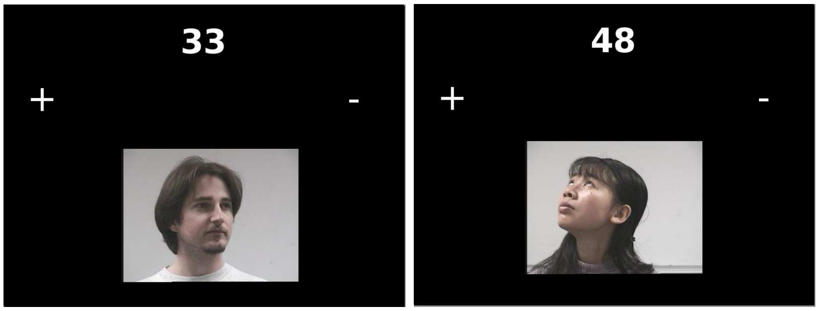

Fig. 2.3 – Exemples d'images de test présentées au sujet pendant l'expérience.

Un autre objectif important de cette expérience est d'obtenir les meilleures performances humaines sur l'estimation de la pose de la tête, pour les comparer ensuite avec les résultats obtenus par notre système. Cependant, nous ne savons pas si cette tâche est naturelle pour les humains. C'est pourquoi les sujets furent divisés aléatoirement en 2 sous-groupes : les sujets "Calibrés" et les sujets "Non Calibrés". Les sujets calibrés ont pu inspecter des images d'exemple étiquetées en orientation aussi longtemps qu'ils le souhaitaient avant de commencer l'expérience. Des exemples d'images d'entraînement sont presentés sur la figure 2.4. Les sujets non calibrés n'ont vu aucune image d'entraînement avant de commencer. Avoir créé ces deux sous-groupes aléatoirement permet de voir si un entraînement préalable augmente les performances des sujets sur l’estimation de l’orientation de la tête.

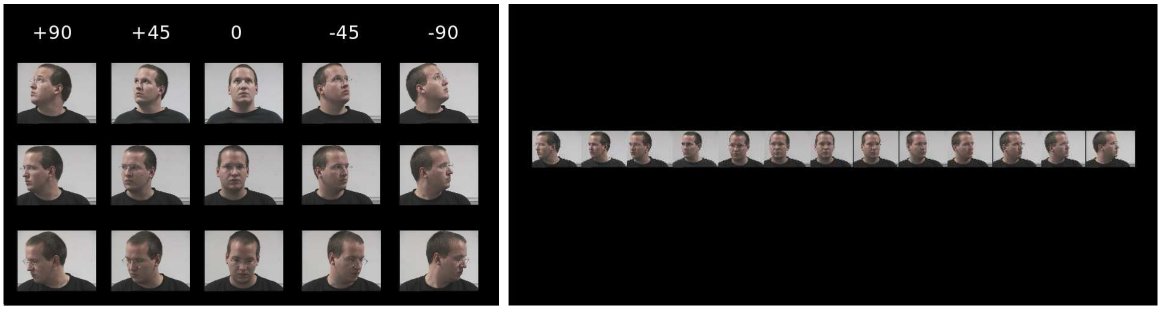

FIG. 2.4 – Exemples d’images d’entraînement montrées aux sujets "Calibrés" pour l’angle pan.

À la fin de notre expérience, nous présentons au sujet une image issue des travaux de Kersten [65]. Cette image est montrée sur la figure 2.2 et représente la projection cylindrique aplatie d’un visage humain sur l’axe pan. Tous les angles pan sont visibles sur cette image. Nous demandons au sujet d’entourer les angles qu’il voit sur l’image. Le but de cette question est de confirmer l’utilisation des poses face et profil comme poses clés par le cerveau humain

2.2.3 Résultats et discussion

Pour mesurer les performances humaines, nous devons définir des métriques. La métrique principale est l’erreur moyenne en pan et en tilt. Cette mesure est définie par la moyenne des différences absolues entre la pose théorique $p(k)$ et la pose $p^*(k)$ estimée par le sujet (2.1) pour l’image k. N est le nombre total d’images sur chaque axe. Nous calculons également l’erreur maximale sur chaque axe pour chaque sujet (2.2). Une autre mesure intéressante est le taux de classification correcte, défini par le nombre de bonnes réponses sur le nombre total de réponses (2.3). Comme l’échantillon d’images de la base de données utilisée contient le même nombre d’images pour chaque pose, nous pouvons calculer une autre métrique : l’erreur moyenne par pose (2.4). Cette métrique permet de voir les poses qui sont bien reconnues par les sujets.

$$ErreurMoyenne = \frac{1}{N} \cdot \sum_{k=1}^{N} \|p(k) - p^*(k)\| \tag{2.1}$$

$$ErreurMax = max_k \|p(k) - p^*(k)\| \tag{2.2}$$

$$ClassificationCorrecte = \frac{Card\{ImagesClassifiees\}}{Card\{Images\}} \tag{2.3}$$

$$ErreurMoyenne(P) = \frac{1}{Card\{Images \in P\}} \cdot \sum_{k \in P} \|p(k) - p^*(k)\| \tag{2.4}$$

Nous avons calculé ces métriques pour tous les sujets et tous les sous-groupes. Les résultats sur les axes pan et tilt sont presentés dans les tableaux 2.2 et 2.3. L’erreur moyenne est de

11.9 degrés en pan et 11 degrés en tilt. L'erreur maximale varie entre 30 et 60 degrés, ce qui est supérieur au pas d'échantillonnage de 15 degrés. Ceci prouve que la base de données est suffisamment échantillonnée pour les sujets.

Pour mettre en relief des différences significatives de performances entre les groupes, nous avons effectué un test d'hypothèse en utilisant un test de Student-Fisher avec un seuil de confiance de 95 %. Les détails de cette opération se trouvent en Annexe B. Les sujets calibrés ne sont pas significativement meilleurs que les sujets non calibrés pour l'estimation de l'angle pan. Par contre, la différence est significative pour l'angle tilt. Les sujets calibrés sont significativement meilleurs que les sujets non calibrés pour l'estimation de cet angle. Ce résultat montre que l'estimation de l'angle pan semble être naturelle, contrairement à celle de l'angle tilt. Ceci peut être dû au fait que les gens tournent plus souvent la tête de gauche à droite que de haut en bas pendant les interactions sociales [135, 64, 128]. Les humains font plus attention aux changements d'orientation de tête sur l'axe horizontal.

Mesures	Erreur Moyenne	Erreur Maximale	Classification Correcte
Tous les sujets	11.85^o	44.79^o	41.58 %
Sujets Calibrés	11.79^o	42.5^o	40.73 %
Sujets Non Calibrés	11.91^o	47.08^o	42.44 %

TAB. 2.2 – *Résultat de l'évaluation sur l'axe pan*

Mesures	Erreur Moyenne	Erreur Maximale	Classification Correcte
Tous les sujets	11.04^o	45.1^o	53.55 %
Sujets Calibrés	$\mathbf{9.45}^o$	39.58^o	59.14 %
Sujets Non Calibrés	$\mathbf{12.63}^o$	50.63^o	47.96 %

TAB. 2.3 – *Résultat de l'évaluation sur l'axe tilt*

L'erreur moyenne par pose en pan et en tilt est montrée sur la figure 2.5. Les sujets reconnaissent bien les poses face et profil, mais moins bien les poses intermédiaires. La pose la mieux reconnue est la pose frontale. Ce fait est confirmé par la présentation de l'image cylindrique de visage de Kersten à la fin de l'expérience. 81% des sujets n'ont pas vu de poses autres que face et profil sur cette image. Ces résultats montrent que les poses face et profil sont utilisées par le système visuel humain comme des poses clés, comme suggéré dans [65].

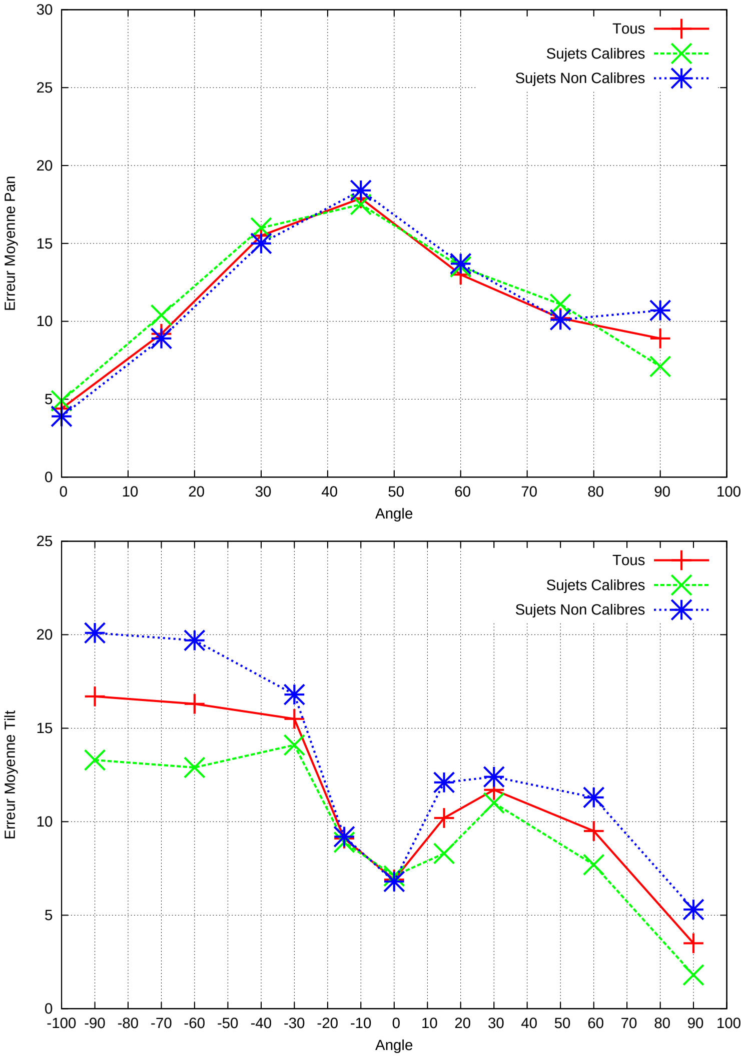

FIG. 2.5 – Erreur moyenne par pose en pan et en tilt de différents groupes.

2.3 Suivi robuste de visage

Cette section décrit le système de suivi de visage temps réel utilisé dans la thèse. Cet algorithme, présenté en détail dans [37], est utilisé pour la détection des visages dans la base de données Pointing 2004, bien que toute autre détection robuste, comme Ada-Boost [151], puisse être utilisée pour cette étape. Nous recherchons d'abord les régions de l'image correspondant au visage à l'aide d'un histogramme de chrominance de peau. Le calcul de la chrominance (r, g) d'un pixel (x, y) est effectué en normalisant les composantes rouge et verte du vecteur de couleur (R, G, B) par son intensité lumineuse $R + G + B$. La densité de probabilité conditionnelle des vecteurs de chrominance (r, g) d'appartenir à une region de peau peut être estimée en utilisant un histogramme. La règle de Bayes nous donne une relation directe entre un pixel (x, y) et sa probabilité $p((x, y) \in Peau|r, g)$ d'être placé dans une région de peau. En effectuant le quotient des histogrammes de l'image entière et de peau, nous obtenons une meilleure répartition de cette probabilité en fonction des autres objets présents sur l'image. Nous obtenons ainsi une carte de probabilité sur toute l'image :

$$\begin{aligned} p((x, y) \in Peau|r, g) &= \frac{p(r, g|(x, y) \in Peau)p((x, y) \in Peau)}{p(r, g)} \\ &= \frac{Histogramme_{peau}(r, g)}{Histogramme_{image}(r, g)} \end{aligned}$$

Pour suivre le visage dans une image, celui-ci doit se retrouver isolé. Sa position, sa taille et son orientation sont estimées et suivies à l'aide d'un filtre de Kalman d'ordre 0 [61]. Le processus de tracking prédit une région d'intérêt (RDI) dans laquelle doit se trouver le visage et qui sera multipliée par une fenêtre gaussienne. Cette opération permet de focaliser la recherche uniquement sur le visage suivi et d'accélérer le temps de calcul. Dans la RDI seront calculés les premier et second moments de la carte de probabilité ainsi obtenue. Ces moments délimitent une ellipse sur l'image correpondant à la région du visage. Cette région est appelée visage estimé. Un exemple de suivi de visage est illustré sur la figure 2.6. La différence entre le visage estimé à l'image courante et le visage estimé à l'image précédente permet de calculer le visage prédit à l'image suivante et la nouvelle RDI. Cette étape est appelée prédiction-vérification. À l'initialisation, le visage prédit peut être égal soit à une sélection manuelle de l'utilisateur, soit à l'image entière. Pour détecter le visage sur les images ne contenant qu'un seul visage, le système est lancé sans intervention de l'utilisateur sur l'image entière jusqu'à ce que le visage estimé se soit stabilisé, ce qui est généralement le cas après 10 itérations. Le système de suivi de visage fonctionne en temps-réel sur des images de 384x288 pixels sur Pentium 800 MHz.

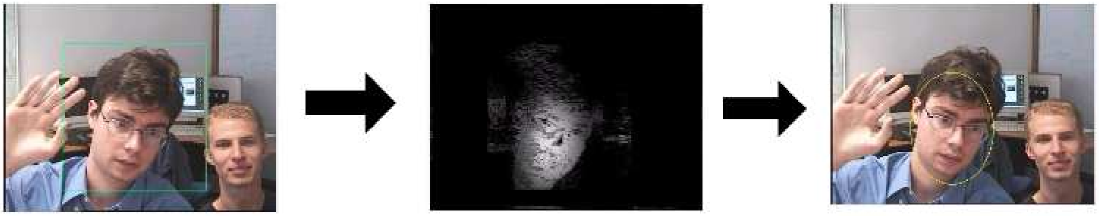

FIG. 2.6 – De gauche à droite : RDI d'un visage dans l'image, Calcul de la carte de probabilité avec fenêtre gaussienne dans la RDI, Ellipse délimitant le visage dans l'image.

À partir des premier et second moments du visage estimé, nous pouvons normaliser l'image du visage en taille et en inclinaison dans une imagette de plus petite résolution en niveaux de gris. La normalisation offre plusieurs avantages. Tout d'abord, elle permet aux opérations suivantes d'être indépendantes de la taille et de l'inclinaison de l'image d'origine. Les temps de calcul ne dépendent alors plus que de la taille de l'imagette. De plus, cette opération permet de ne conserver que les changements d'intensité lumineuse. Un dernier avantage important est de rendre tous les visages droits, et ainsi de pouvoir localiser les mêmes points faciaux à peu près dans les mêmes régions pour chaque pose. Dans nos expériences, les imagettes ont une taille de 23x30 pixels. Un exemple de normalisation d'une image de visage est montré sur la figure 2.7. Toutes les opérations ultérieures ont lieu dans cette imagette. La normalisation de la région du visage est une étape utile à notre système d'estimation de pose de la tête.

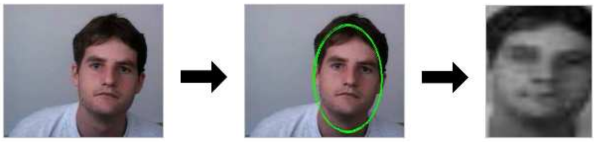

FIG. 2.7 – Détection et normalisation de la région de l'image correspondant au visage.

2.4 Estimation de la pose de la tête par apparence globale

Dans cette section, nous utilisons les imagettes normalisées du visage obtenues par le système de suivi robuste pour apprendre des prototypes d'orientations de la tête. Les imagettes représentant la même pose sont injectées dans une mémoire autoassociative, entraînée par la règle d'apprentissage de Widrow-Hoff. La classification des poses se fait en comparant l'image du visage d'origine et les images reconstruites par les prototypes. La pose dont l'image reconstruite est la plus similaire à l'image source est sélectionnée comme pose courante.

2.4.1 Mémoires autoassociatives linéaires

Les mémoires autoassociatives linéaires sont un cas particulier de réseaux de neurones à une couche où les entrées sont associées à elles-mêmes en sortie. Elles ont été utilisées pour la première fois par Kohonen pour sauvegarder et charger des images [70]. Ces objets associent des images à leur classe respective, même si les images sont dégradées ou une partie en est cachée. Une image x' en niveaux de gris est décrite par son vecteur normalisé $x = \frac{x'}{\|x'\|}$. Un ensemble de M images composées de N pixels d'une même classe est sauvegardé dans la matrice $X = (x_1, x_2, ..., x_M)$ de taille N x M. La mémoire autoassociative de la classe k est représentée par la matrice de connexion W_k, de taille N x N. Le nombre de cellules dans la matrice est égal au nombre de pixels de l'image au carré. Son calcul a donc une complexité de $O(N^2)$. La réponse d'une cellule est égale à la somme de ses entrées multipliées par les poids de la matrice. L'image reconstruite y_k est donc obtenue en calculant le produit de l'image source x par la matrice de connexion W_k :

$$y_k = W_k \cdot x \tag{2.5}$$

La similarité de l'image source et d'une classe d'images k est estimée comme le cosinus de leurs vecteurs x et y_k :

$$cos(x, y) = y^T.x = \frac{y'^T.x'}{\|y'^T\|\|x'\|} \tag{2.6}$$

Comme les vecteurs x et y sont normalisés en énergie, leur cosinus est compris entre 0 et 1, où un score de 1 représente une correspondance parfaite.

La matrice de connexion W_k est initialisée avec la règle d'apprentissage de Hebb :

$$W_k = X_k \cdot X_k^T = \sum_{i=1}^{M} x_{ik} \cdot x_{ik}^T \tag{2.7}$$

Les images reconstruites avec cette règle sont égales à la première eigenface de la classe d'images. Pour augmenter la performance de classification, nous entraînons les mémoires autoassociatives linéaires avec la règle de Widrow-Hoff.

2.4.2 Règle d'apprentissage de Widrow-Hoff

La règle d'apprentissage de Widrow-Hoff est une règle de correction locale améliorant la performance des associateurs [148]. À chaque présentation d'une image, chaque cellule de la matrice de connexion modifie ses poids en corrigeant la différence entre la réponse obtenue et la réponse désirée. Les images X d'une même classe sont présentées itérativement avec un pas d'adaptation jusqu'à ce qu'elles soient correctement classifiées. La matrice de connexion W devient ainsi sphéricalement normalisée [1]. La règle de correction de Widrow-Hoff est décrite par l'équation :

$$W^{t+1} \;=\; W^t + \eta(x - W^t \cdot x)x^T \tag{2.8}$$

où η est le pas d'adaptation et t l'itération courante. Pour rendre les mémoires adaptatives et pour les faire tenir compte des variations intraclasses, nous utilisons un nombre d'itérations ι.

La figure 2.8 montre des exemples d'images reconstruites avec les règles de Hebb et Widrow-Hoff. La mémoire entraînée par la règle de Hebb délivre la même réponse pour les images reconstruites. En conséquence, le cosinus entre l'image source et l'image reconstruite n'est pas assez discriminant pour la classification. La mémoire entraînée avec la règle de Widrow-Hoff

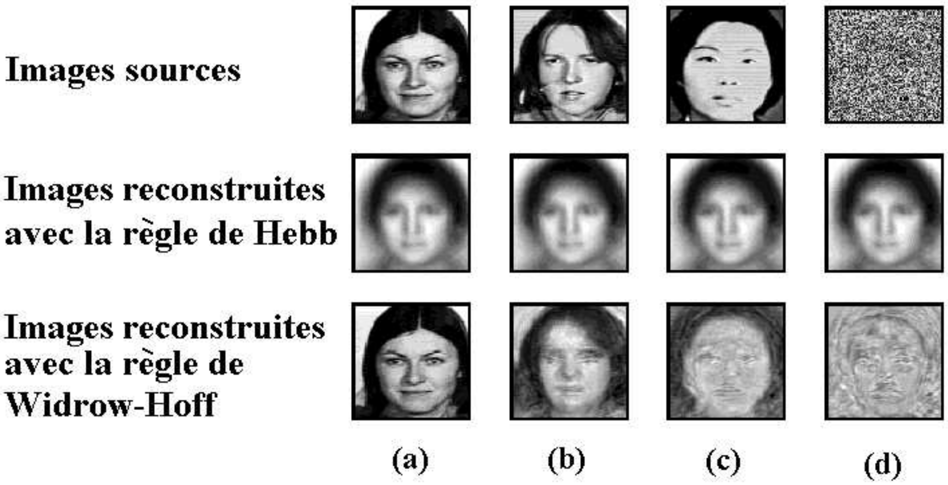
Images sources
Images reconstruites avec la règle de Hebb
Images reconstruites avec la règle de Widrow-Hoff
(a)
(b)
(c)
(d)

FIG. 2.8 – Reconstruction d'images avec des mémoires autoassociatives linéaires entraînées par les règles de Hebb et de Widrow-Hoff. La classe d'images à reconnaître est une classe de visages de femmes caucasiennes. (a) est une image de la base d'apprentissage. (b) est une image de la classe mais non apprise. (c) n'appartient pas à la classe. (d) est une image randomisée et n'appartient pas non plus à la classe [148].

reconstruit les images en les dégradant peu si elles appartiennent à la classe apprise, mais en les dégradant beaucoup si elles n'appartiennent pas à la classe. Le cosinus entre l'image source et l'image obtenue est ainsi plus discriminant. Avec un bon choix du pas d'adaptation η et du nombre d'itérations ι, une image peut être bien reconstruite, même en cas d'occlusion partielle.

La règle d'apprentissage de Widrow-Hoff a montré de bons résultats dans des problèmes classiques de vision tels que la reconnaissance du visage, du sexe et de l'ethnicité. Le nombre de composants principaux ou de dimensions à utiliser n'ont pas besoin d'être définis, pas plus que le choix d'une structure ou du nombre de cellules dans une couche cachée. Seuls deux paramètres doivent être réglés. Nous construisons des prototypes d'orientations de la tête en entraînant des mémoires autoassociatives linéaires par la règle d'apprentissage de Widrow-Hoff.

2.4.3 Application à l'estimation de la pose de la tête

Nous considérons chaque pose du visage comme une classe d'images. Une mémoire autoassociative W_k est entraînée pour chaque pose k. Nous utilisons la base de données Pointing 2004, où se trouvent un même nombre d'images par pose. Nous calculons les cosinus entre l'image source et les images reconstruites par les mémoires. La pose dont le cosinus est le plus élevé est sélectionnée comme pose courante.

Les poses peuvent être apprises de deux façons : séparément ou en groupe. Dans l'entraînement des poses séparées, nous apprenons une mémoire pour chaque angle sur un axe en faisant varier l'angle sur l'autre axe. Chaque mémoire capture l'information d'un seul angle sur un seul axe. Tous les angles pan sont appris en faisant varier les angles tilt, et inversement. Nous obtenons ainsi 13 prototypes pour l'angle pan et 9 prototypes pour l'angle tilt. Le pas d'adaptation η utlisé est de 0.008 en pan et 0.006 en tilt.

Dans l'entraînement des poses groupées, les angles pan et tilt sont appris ensemble. Chaque mémoire est apprise par un ensemble d'images de visage de la même pose et contient l'information d'un couple d'angles pan et tilt. Nous obtenons ainsi 93 prototypes. Le pas d'adaptation η utilisé est de 0.007.

La base de données Pointing 2004 permet de mesurer la performance de notre système sur des sujets connus et inconnus. Cette base de données contient 2 sets de 15 personnes. Pour tester sur des sujets connus, nous effectuons une validation croisée sur les sets : le premier set est pris comme base d'apprentissage, tandis que le second est pris comme base de test, et inversement. Ainsi, toutes les personnes sont apprises dans la base d'apprentissage. Pour tester sur des sujets inconnus, nous utilisons la méthode dite du Jack-Knife : pour chaque personne, toutes les images sont utilisées comme base d'apprentissage sauf celles de ladite personne, qui seront utilisées pour le test. La personne à tester change à chaque itération. Ainsi, aucune image de la base d'apprentissage ne contient des images de la personne à tester.

Nous utilisons les mêmes métriques que dans la section 2.2 : l'erreur moyenne, le taux de classification correcte et l'erreur moyenne par pose. Nous définissons une autre métrique, le taux de classification correcte en pan à 15 degrés près. Une image est classifiée correctement à 15 degrés près si la différence $\|p(k) - p^*(k)\|$ n'excède pas 15 degrés :

$$ClassificationCorrecte15 = \frac{Card\{ImagesClassifiees15^o\}}{Card\{Images\}} \tag{2.9}$$

Au-delà de 70 itérations, l'erreur moyenne en pan et en tilt stagne. Nous utilisons donc un nombre d'itérations $\iota = 70$ dans nos expériences.

2.4.4 Résultats et discussion

Nous comparons les performances de notre méthode avec celles obtenues par d'autres méthodes de l'état de l'art. Pour le test sur les sujets connus, nous comparons nos résultats avec ceux des modèles de tenseurs, des ACP, des LEA [145] et des réseaux de neurones [152]. Pour le test sur les sujets inconnus, nous comparons nos résultats avec ceux de l'algorigthme du plus proche voisin. Cet algorithme recherche l'image la plus proche dans la base d'apprentissage. Les différentes performances sont montrées dans les tableaux 2.4 et 2.5.

Les prototypes d'orientations de la tête sous forme de mémoires autoassociatives linéaires obtiennent de bonnes performances sur les sujets connus et inconnus. La comparaison avec l'algorithme de recherche du plus proche voisin montre l'utilité de regrouper les images représentant la même pose, résultant en un gain en performances et en temps de calcul. Ces résultats

Métrique	Tenseur	ACP	LEA	RN	**Sép. MAAL**	**Grp. MAAL**
Erreur Moyenne Pan	12.9^o	14.1^o	15.9^o	12.3^o	**7.6**^{o}	**8.4**^{o}
Erreur Moyenne Tilt	17.9^o	14.9^o	17.4^o	12.8^o	**11.2**^{o}	**8.9**^{o}
Classification Pan 0^o	49.3 %	55.2 %	45.2 %	41,8 %	**61.2 %**	**59.4 %**
Classification Tilt 0^o	54.9 %	57.9 %	50.6 %	52.1 %	**54.2 %**	**62.4 %**
Classification Pan 15^o	84.2 %	84.3 %	81.5 %	-	**92.4 %**	**90.8 %**

TAB. 2.4 – *Évaluation de performance sur les sujets connus. RN fait référence aux Réseaux de neurones et MAAL aux Mémoires AutoAssociatives Linéaires [40, 145, 152].*

Métrique	Sép. PPV	Grp. PPV	**Sép. MAAL**	**Grp. MAAL**
Erreur Moyenne Pan	14.1^o	13.9^o	**10.1**^{o}	**10.1**^{o}
Erreur Moyenne Tilt	15.9^o	21.1^o	**15.9**^{o}	**16.3**^{o}
Classification Pan 0^o	40.9 %	40.9 %	**50.3** %	**50.4 %**
Classification Tilt 0^o	41.9 %	41.5 %	**43.9 %**	**45.5 %**
Classification Pan 15^o	80 %	80.1 %	**88.8 %**	**88.1 %**

TAB. 2.5 – *Évaluation de performance sur les sujets inconnus. PPV fait référence à l'algorithme du Plus Proche Voisin et MAAL aux Mémoires AutoAssociatives Linéaires*

montrent aussi que l'entraînement de poses groupées n'améliore pas significativement les performances. De plus, le système fonctionne plus rapidement à 15 images par seconde avec les 22 prototypes appris séparément qu'à 1 image par seconde avec les 93 prototypes appris en groupe. Par la suite, nous n'utiliserons plus que les prototypes dont les angles pan et tilt ont été appris séparément.

L'erreur moyenne par pose est montrée sur la figure 2.9 et comparée aux performances humaines de la section 1.2.2. Les performances de notre système sont plus stables sur l'angle pan que les performances humaines. Les erreurs minimales se trouvent aussi aux poses face et profil. Notre méthode est significativement plus performante que les humains pour l'estimation de l'angle pan, et similaire pour l'estimation de l'angle tilt sur des sujets connus. Cependant, les humains demeurent meilleurs pour l'estimation de l'angle tilt sur des sujets inconnus. Augmenter la taille de l'imagette normalisée n'améliore pas significativement les résulats. Les prototypes de poses délivrent de bons résultats sur les poses hautes, moins sur les poses basses. Ceci est dû au fait que les cheveux deviennent plus visibles sur les images de poses basses, l'apparence globale peut alors beaucoup changer d'une personne à une autre. Les résultats de l'estimation sur des sujets inconnus peuvent être améliorés en augmentant la taille de l'imagette du visage. Cependant, les mémoires autoassociatives linéaires ont une complexité quadratique en fonction de la taille de l'imagette. Nous utilisons une autre méthode basée sur les apparences locales de

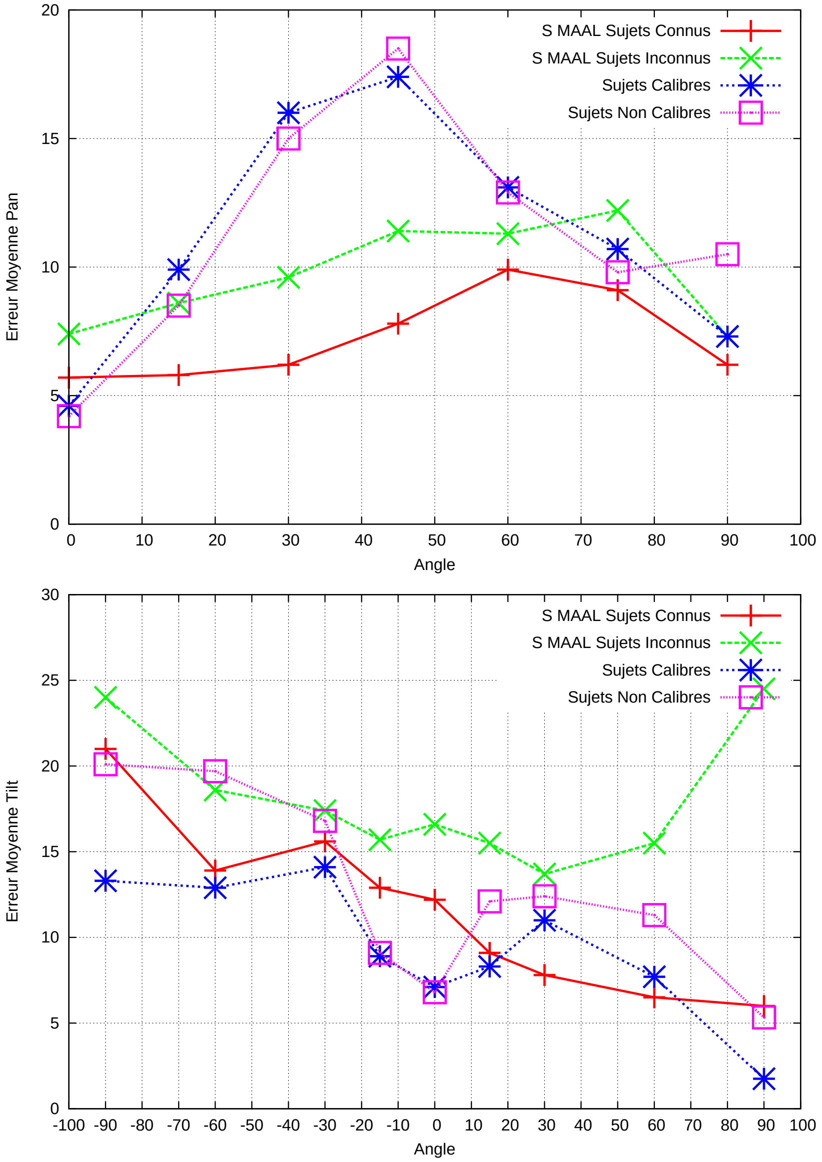

FIG. 2.9 – Erreur moyenne sur les axes pan et tilt.

l'image du visage pour augmenter les performances de l'estimation.

2.5 Détection des régions saillantes du visage

Dans cette section, nous décrivons les imagettes de visage à l'aide de champs réceptifs gaussiens. Ces champs réceptifs permettent de décrire l'apparence locale d'un voisinage de pixels à une échelle donnée. Normalisés à leurs échelles intrinsèques, les vecteurs de réponse aux champs réceptifs gaussiens apparaissent comme des détecteurs fiables de traits du visage robustes à l'illumination, la pose et l'identité. Ces traits du visage peuvent être plus ou moins saillants pour la pose considérée.

2.5.1 Champs réceptifs gaussiens

Le terme "champ réceptif" désigne un récepteur capable de décrire les motifs locaux de changements d'intensité dans les images. De tels descripteurs sont utilisés en Vision par Ordinateur sous des noms différents : mesure locale du n^e ordre [69], vecteurs de caractéristiques iconiques [113], points d'intérêt naturels [118] et SIFT [82]. Dans la suite, les champs réceptifs gaussiens désigneront des fonctions linéaires locales basées sur les dérivées gaussiennes d'ordre croissant.

La réponse $L_{k,\sigma}$ d'une image I en niveaux de gris à un champ réceptif gaussien $G_{k,\sigma}$ d'échelle σ et de direction k est égale à la convolution $L_{k,\sigma} = I \otimes G_{k,\sigma}$. L'ensemble des valeurs $L_{k,\sigma}$ forme le vecteur de caractéristiques L_σ :

$$L_\sigma = (L_{1,\sigma}, L_{2,\sigma}, ..., L_{n,\sigma})$$

L'ordre et la direction, représentés par k, fait référence au type de derivée du champ réceptif et a la forme $x^i y^j$. La figure 2.10 montre une description d'un voisinage de l'image par un champ réceptif gaussien. Pour chaque pixel (x, y), la dérivée gaussienne d'échelle σ s'exprime par la formule :

$$G_{x^i y^j,\sigma}(x, y) = \frac{\partial^i}{\partial x^i} \frac{\partial^j}{\partial y^j} G_\sigma(x, y) \tag{2.10}$$

En 2 dimensions, le noyau gaussien est défini par :

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

L'espace de vecteurs obtenu par les champs réceptifs est appelé espace d'apparence locale ou espace de caractéristiques. Deux voisinages d'apparence locale similaire sont représentés par deux vecteurs proches dans l'espace des caractéristiques. Pour mesurer la similarité en apparence locale de deux voisinages, nous calculons leur distance de Mahalanobis dans cet espace. Les noyaux gaussiens possèdent des propriétés d'invariance intéressantes pour la description d'image comme la séparabilité, la similarité sur les échelles et la différentiabilité. Le calcul d'un champ réceptif sur un voisinage de pixels est linéaire.

Les dérivées de premier ordre décrivent l'orientation locale des lignes dans l'image, tandis que la courbure locale des lignes est perçue par les dérivées du second ordre. Nous ne prenons pas en compte les dérivées d'ordre 0 pour rester robuste aux changements d'intensité lumineuse. Les dérivées d'ordre strictement supérieur à 2 n'apportent de l'information que si une structure

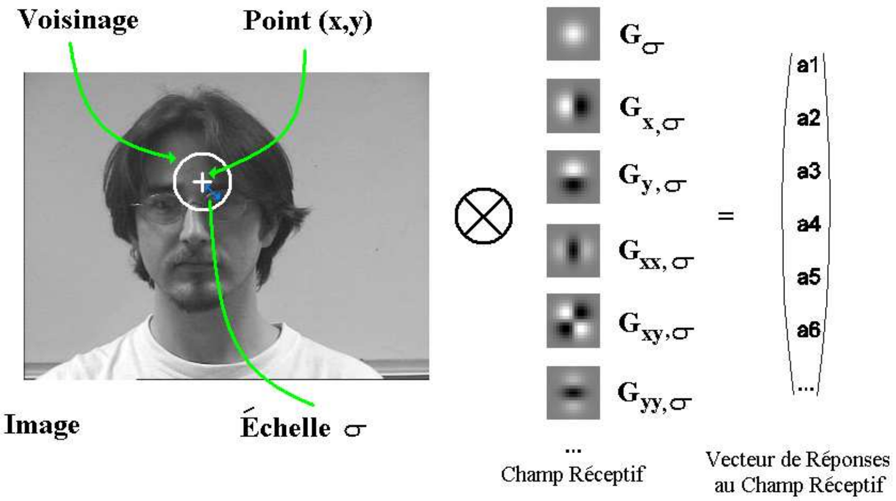

FIG. 2.10 – Exemple de description d'un voisinage dans l'image par un champ réceptif gaussien.

importante est détectée dans les termes du second ordre [68]. Pour cette raison, nous ne prenons en compte que les termes du premier et du second ordre. Nous obtenons alors un vecteur de caractéristiques à 5 dimensions : $L_\sigma = (L_{x,\sigma}, L_{y,\sigma}, L_{xx,\sigma}, L_{xy,\sigma}, L_{yy,\sigma})$.

Pour analyser les voisinages de pixels à une échelle appropriée, nous utilisons la méthode proposée par Lindeberg [78]. Les échelles calculées sont appelées échelles intrinsèques[1]. Un profil d'échelle $\sigma(x, y)$ est construit à chaque pixel (x, y) en collectant les réponses à l'énergie normalisée du laplacien, définie ci-dessous par :

$$\nabla^2 G_\sigma = \sigma^2 (G_{\sigma,xx} + G_{\sigma,yy}) \tag{2.11}$$

Les profils d'échelle admettent chacun au moins un maximum local. La valeur minimale $\sigma_{opt}(x, y)$ des maxima locaux d'un profil $\sigma_{opt}(x, y)$ est choisie comme échelle intrinsèque du pixel (x, y). Quand deux images sont zoomées, le quotient des échelles intrinsèques du même pixel des deux images est égal au rapport de zoom. C'est pourquoi l'énergie normalisée du laplacien est invariante aux changements d'échelle. Sur chaque image de visage, nous calculons l'échelle intrinsèque des pixels et obtenons ainsi une description de ceux-ci par un ensemble de vecteurs à 5 dimensions $L_{\sigma opt} = (L_{x,\sigma opt}, L_{y,\sigma opt}, L_{xx,\sigma opt}, L_{xy,\sigma opt}, L_{yy,\sigma opt})$.

2.5.2 Détection des régions saillantes d'un visage

Notre objectif est de concevoir des descripteurs locaux robustes aux changements d'illumination, de pose et d'identité pour détecter les régions saillantes du visage pour pouvoir en-

[1] ou échelles caractéristiques

suite estimer sa pose. Pour détecter de tels points, de nombreuses méthodes ont été proposées comme les textons [89], les caractéristiques génériques [93, 118, 79], les caractéristiques propres [149], les blobs [50] ou points de selle et les maxima de l'intensité lumineuse [107]. Cependant, ces descripteurs sont sensibles à l'illumination et peuvent fournir un nombre trop abondant de points. Les points d'intérêt naturels définis par Lindeberg [78] ne décrivent que des structures circulaires et ne sont pas appropriés aux objets déformables, dont les structures changent de forme d'une pose à l'autre.

En recherchant la notion de saillance dans la littérature, nous avons trouvé deux définitions. Une définition intuitive d'un objet saillant est un objet qui attire l'attention. Une définition mathématique de la saillance a été donnée par Walker dans [153] : un objet saillant est un objet dont les caractéristiques sont isolées dans un espace dense dans lequel elles évoluent. L'espace à 5 dimensions formé par les vecteurs de réponses aux champs réceptifs gaussiens est dense. Cependant, les vecteurs obtenus sur les images de visage sont souvent groupés en un bloc, ce qui rend difficile l'isolation d'un groupe de vecteurs particulier. De plus, un groupe de vecteurs isolé dans l'espace de caractéristiques n'est pas forcément isolé sur l'image. Une région saillante dans l'image ne doit couvrir qu'une petite portion de celle-ci, sinon elle n'est plus saillante.

Nous proposons la définition suivante pour les régions saillantes d'une image : une région est saillante si ses pixels voisins ont une apparence locale similaire dans un rayon limité. Quand le rayon est trop grand, la région est trop grande et donc non saillante. Quand le rayon est trop petit, la région est considérée comme outlier. Cette définition comporte deux paramètres : la taille des régions saillantes δ et le seuil de similarité d_S. Deux voisinages de pixels sont considérés d'apparence locale différente si leur distance de Mahalanobis dépasse ce seuil. Pour chaque pixel (x, y), nous calculons sa distance de Mahalanobis avec les pixels $(x + \iota_x \delta x, y + \iota_y \delta y)$ délimitant la région. Les variables (ι_x, ι_y) peuvent prendre les valeurs $\{-1, 0, 1\}$ et représentent les 8 directions cardinales. Si les 8 distances dépassent le seuil de similarité d_S, alors le pixel est considéré comme faisant partie d'une région saillante. Si seulement une ou deux distances sont inférieures au seuil, alors le pixel fait sans doute partie d'une crête ou d'une ligne d'intérêt. Si la plupart des distances sont inférieures au seuil, alors le pixel fait partie d'une région non saillante ou d'un outlier. Des exemples de profil d'apparence locale de régions faciales sont montrés sur la figure 2.11. La condition de saillance d'un pixel est resumée ci-dessous :

$$\forall (\iota_x, \iota_y) \in \{-1, 0, 1\}^2 - (0, 0) \qquad d_M(F(x, y), F(x + \iota \delta x, y + \iota \delta y)) > d_S \tag{2.12}$$

Nous utilisons un seuil de similarité de $d_S = 1$ et un rayon de $\delta = 10$ pixels pour la détection des régions saillantes des images de visage. La performance de notre méthode est comparée à celles obtenues par d'autres détecteurs sur la figure 2.12. Les champs réceptifs gaussiens donnent de bons résultats et la détection des régions saillantes apparaît robuste à la pose et à l'identité. Les régions saillantes obtenues couvrent principalement les régions du visage correspondant aux yeux, au nez, à la bouche et au contour du visage. Ces résultats ressemblent à ceux obtenus par Yarbus sur les régions du visage les plus examinées par les humains. La position

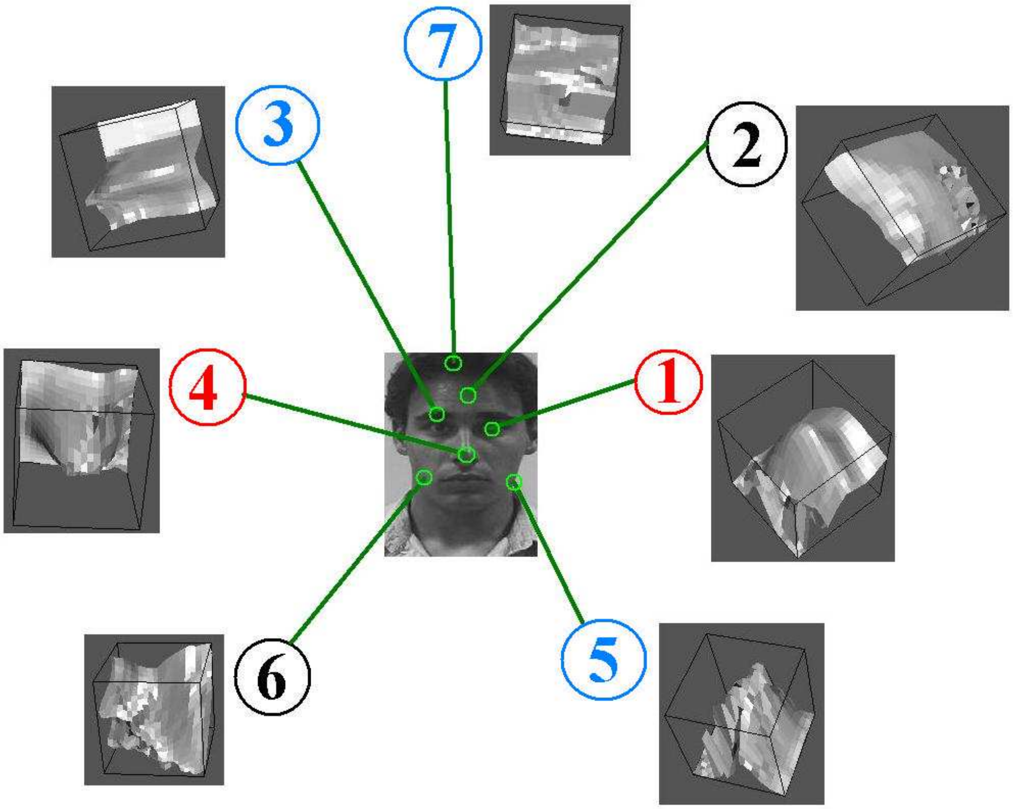

FIG. 2.11 – Apparences locales de traits du visage : (1) Yeux, (2) Front, (3) Sourcil, (4) Nez, (5) Contour du visage, (6) Joue, (7) Cheveux. Les régions (1) et (4) apparaissent comme des blobs et sont considérées comme saillantes, les régions (3) et (5) apparaissent comme des crêtes, les autres régions ne présentent pas de structures similaires et ne sont donc pas considérées comme saillantes.

des régions saillantes par rapport au visage peut apporter des informations supplémentaires pour l'estimation de l'orientation de la tête. Dans la section suivante, nous construisons une structure basée sur ces régions ainsi que sur leurs descripteurs.

2.6 Estimation raffinée de la pose de la tête par apparence locale

Cette section explique l'utilisation de graphes saillants à base de vecteurs de réponses aux champs réceptifs gaussiens normalisés à leurs échelles intrinsèques. La structure de graphe a des propriétés intéressantes car elle décrit à la fois les informations de texture et leur relations géometriques dans l'image. Les nœuds du graphe sont étiquetés par des vecteurs de faible

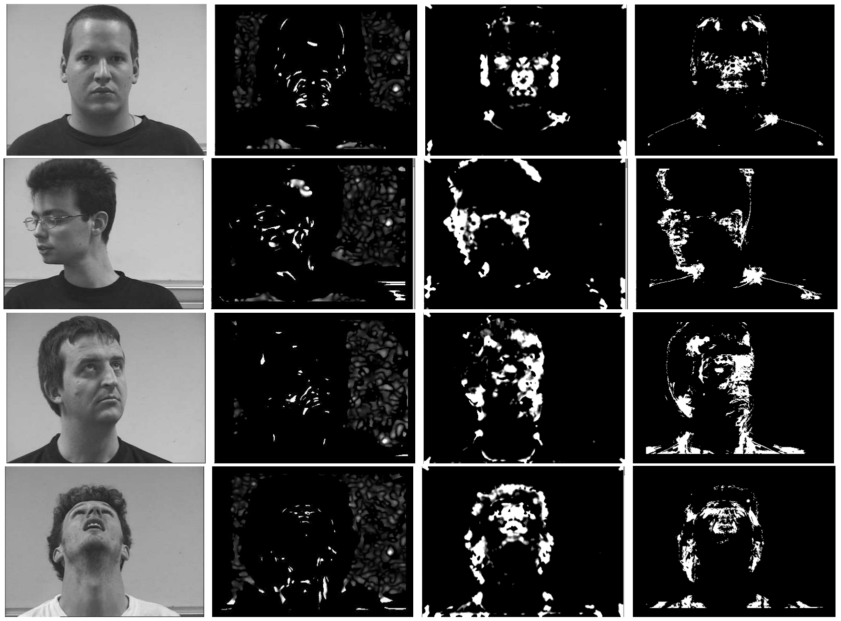

FIG. 2.12 – Exemples de cartes de saillance du visage. De gauche à droite : Image originale 1/4 PAL, Points d'intérêt naturels de Lindeberg à une échelle de 5 pixels, Points de Harris [45], Régions saillantes du visage obtenues par champs réceptifs gaussiens.

dimension clusterisés hiérarchiquement et peuvent se déplacer selon la saillance des points faciaux qu'ils représentent. La première estimation de la pose du visage obtenue dans la section 2.4 est raffinée en recherchant le graphe le plus similaire à l'image de visage courante.

2.6.1 Structure de graphes saillants

La position relative des régions saillantes du visage par rapport à la tête peut fournir des informations importantes sur son orientation. Cependant, l'estimation directe de la pose à partir de celles-ci est rendue difficile par :

- les changements d'emplacement des régions dus aux changements d'identité ;
- les changements d'apparence des régions dus aux changements d'identité ;
- les changements d'emplacement des régions dus à l'alignement imparfait des imagettes.

Pour faire face à ces problèmes, nous adaptons les graphes élastiques introduits par Von der Malsburg [158] pour en faire des graphes à base de champs réceptifs gaussiens.

Un graphe G se compose d'un ensemble de N nœuds n_j etiquetés par leurs descripteurs X_j. Dans la littérature, les ondelettes de Gabor jouent le rôle de descripteurs. L'utilisation de champs réceptifs gaussiens fournit une description similaire avec un coût inférieur en temps de calcul. L'estimation de l'orientation de la tête a précédemment été implémentée avec des graphes élastiques [24, 90, 71, 160]. Néanmoins, ces méthodes requièrent une bonne résolution de l'image du visage. De plus, les graphes de visage sont construits de façon empirique. Nous ne savons si le choix de la position des nœuds et des arêtes est pertinent pour l'estimation de la pose. Entraîner une nouvelle personne ou une nouvelle pose nécessite d'etiqueter manuellement les nœuds et les arêtes du graphe. Comme nous ne voulons pas d'annotation manuelle dans notre système, nous utilisons des graphes dont les nœuds sont répartis régulièrement sur l'imagette du visage.

Nous étendons les graphes utilisés dans [42]. La structure de graphe décrit à la fois les informations de texture et leur relations géométriques dans l'image. Nous utilisons les vecteurs $L_{\sigma opt(x,y)}(x, y)$ de réponses à 5 dimensions aux champs réceptifs gaussiens normalisés à leurs échelles intrinsèques obtenus dans la section précédente comme descripteurs des nœuds n_j du graphe. Nous contruisons un modèle de graphe pour chaque pose du visage $Pose_i$ en rassemblant toutes les réponses des nœuds. Chaque nœud n_j est etiqueté par un ensemble de M vecteurs $\{X_{jk}\}$, où M est le nombre d'images dans la base d'apprentissage. Cet ensemble de vecteurs décrit les apparences possibles du point facial trouvé à l'emplacement du nœud n_j. La transformation d'un graphe en modèle de graphe est montrée sur la figure 2.13.

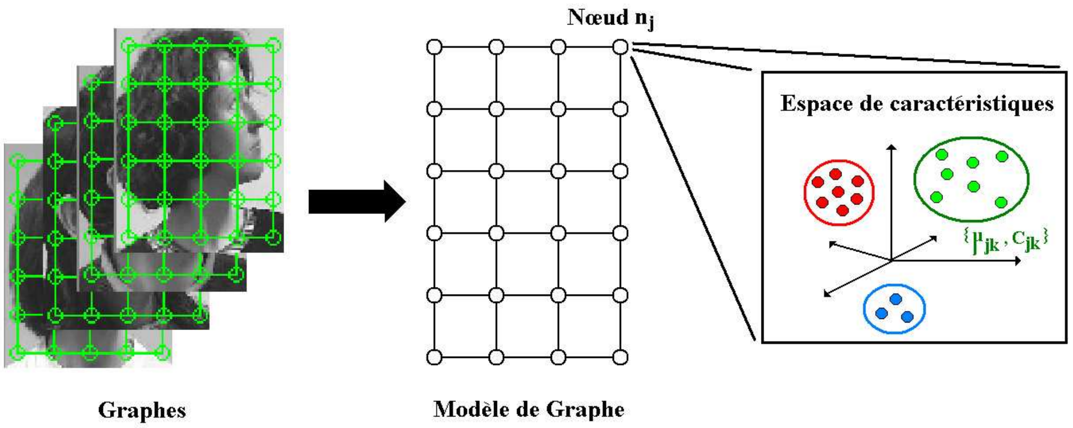

FIG. 2.13 – Transposition de graphes sur les images de visage de même pose en modèle de graphe.

Le même point facial peut avoir différents aspects selon les personnes. Pour une meilleure représentation des apparences possibles d'un même point, nous effectuons un clustering hiérarchique [60] sur les nuages de points obtenus dans l'espace de caractéristiques à chaque nœud, qui contiennent alors chacun K clusters A_i de centre μ_i et de matrice de covariance C_i. Dans nos expériences, nous utilisons un facteur maximal de distances calculées de $\kappa = 2.5$. L'opération de clustering hiérarchique sur les vecteurs de réponses aux nœuds du modèle de graphe permet de mieux tenir compte des changements d'apparences dus aux changements d'identité.

Pour prendre en compte les variations de positions des points faciaux dues au non-alignement des imagettes et aux changements d'identité, les modèles de graphe peuvent être déformés localement en cherchant durant la phase de correspondance d'un nœud le point facial le plus similaire dans une petite fenêtre, comme proposé dans [109]. La taille de la fenêtre ne doit pas excéder la distance ld_{max} entre les nœuds, pour préserver leur ordre.

Les modèles de graphe de champs réceptifs gaussiens sont l'extension intuitive des régions saillantes du visage obtenues dans la section précédente. Une région de l'image est considérée comme saillante si ses pixels voisins partagent une apparence similaire dans un rayon limité δ. Le déplacement local des nœuds correspond à ce rayon δ. Nous proposons de définir le déplacement local maximal d'un nœud en fonction de la saillance du point facial qu'il représente. Les régions saillantes sont détectées sur chaque image de visage. En additionnant les régions obtenues puis en les divisant par le nombre d'images, nous obtenons une carte de saillance pour chaque pose, comme illustré sur la figure 2.14.

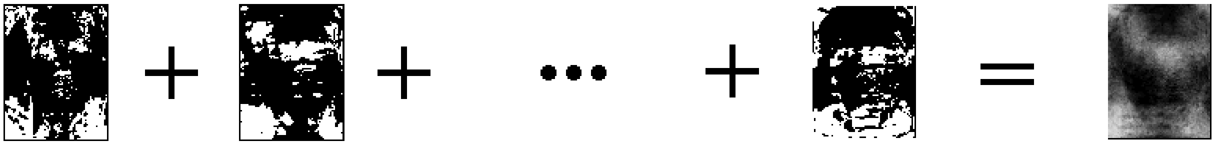

FIG. 2.14 – Exemple de régions saillantes détectées sur des images de même pose et leur combinaison pour obtenir une carte de saillance. Les pixels sombres représentent des régions non saillantes tandis que les pixels clairs représentent des régions saillantes.

La carte de saillance donne une relation directe entre un pixel (x, y) et sa saillance $S(x, y)$ comprise entre 0 et 1. Plus un pixel est saillant, plus son emplacement est pertinent pour la pose considérée. La rigidité d'un nœud du graphe est proportionnelle à sa saillance. Un nœud placé à un point saillant est important et ne doit pas trop bouger de son emplacement initial. À l'opposé, un nœud placé à un point non saillant ne représente pas de traits du visage pertinent pour la pose et peut se mouvoir avec un déplacement local maximal égal à la distance entre 2 nœuds ld_{max}. Nous appelons les modèles de graphe ainsi construits les graphes saillants. En notant (x_j, y_j) l'emplacement du nœud n_j, le déplacement local maximal $ld(n_j)$ s'écrit :

$$ld(n_j) = (1 - S(x_j, y_j)) \cdot ld_{max} \tag{2.13}$$

2.6.2 Application à l'estimation de la pose de la tête

Les mémoires autoassociatives linéaires décrites dans la section 2.4 permettent d'obtenir une première estimation de l'orientation de la tête. Nous raffinons cette estimation en recherchant parmi les poses voisines de celle obtenue en première estimation le graphe saillant le plus similaire à l'image de visage courante. Dans nos expériences, nous avons utilisé des graphes de

12x15 nœuds. La complexité en temps de calcul est proprotionnelle au nombre de nœuds, qui ne peut dépasser le nombre de pixels de l'imagette. Elle est donc linéaire par rapport au nombre de pixels. Nous comparons la performance des graphes saillants à d'autres types de graphe :

– **MAAL**
  Mémoires AutoAssociatives Linéaires entraînées séparément.
– **Graphes Saillants**
  Graphes décrits dans cette section.
– **Graphes 1-Cluster**
  Graphes où les apparences des nœuds ne sont pas clusterisées hiérarchiquement mais représentées par un seul cluster.
– **Graphes Orientés**
  Graphes localisés sur la région de l'image du visage supposée contenir des traits saillants. Des exemples peuvent être vus sur la figure 2.15.
– **Graphes Fixes**
  Graphes dont les nœuds sont fixes, ce qui revient à considérer chaque point de l'image comme saillant.
– **Graphes Naïfs**
  Graphes dont les nœuds peuvent se mouvoir avec le déplacement maximal, ce qui revient à considérer chaque point de l'image comme non saillant.

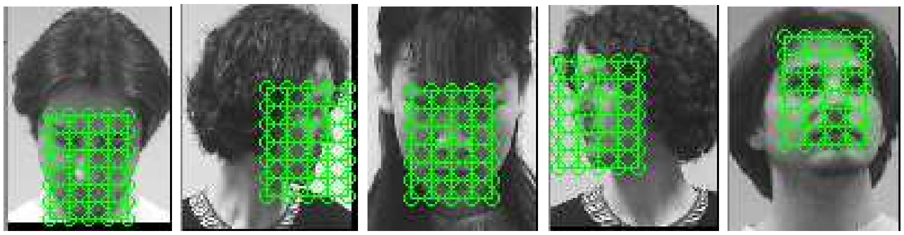

FIG. 2.15 – Exemples de graphes orientés. Les centres des graphes sont calculés en fonction de la pose du visage.

2.6.3 Résultats et discussion

La performance des différentes méthodes est montrée sur le tableau 2.6. L'utilisation de graphes saillants combinés avec les mémoires autoassociatives linéaires donnent les meilleurs résultats. L'estimation de l'angle tilt est la plus améliorée. La combinaison des deux approches fonctionne mieux que l'utilisation d'une seule approche.

Les graphes saillants sont meilleurs que les graphes 1-Cluster. Ce résultat démontre l'utilité de représenter les changements d'aspect dus à l'identité par un clustering hiérarchique de vecteurs de caractéristiques.

Méthode	Erreur Moyenne Pan	Erreur Moyenne Tilt
Graphes Saillants	16.2^o	16.2^o
MAAL	10.1^o	15.9^o
MAAL + Graphes 1-Cluster	11.5^o	13.5^o
MAAL + Graphes Orientés	10.8^o	13.5^o
MAAL + Graphes Fixes	12.7^o	14.9^o
MAAL + Graphes Naïfs	12.2^o	13.5^o
MAAL + Graphes Saillants	**10.1^o**	**12.6^o**

TAB. 2.6 – *Performance des différentes méthodes. MAAL fait référence aux Mémoires AutoAssociatives Linéaires. La résolution des images est de 75x100 pixels.*

Les graphes saillants sont meilleurs que les graphes orientés. Ce résultat montre que plus le graphe couvre d'informations géométriques sur l'imagette du visage, plus il sera performant.

Les graphes saillants sont meilleurs que les graphes fixes. Ce résultat témoigne de l'utilité d'autoriser les nœuds du graphe à se déplacer pour tenir compte des déplacements de points faciaux dus aux changements d'identité et au non-alignement des imagettes.

Les graphes saillants sont meilleurs que les graphes naïfs. Ce résultat montre qu'en limitant le déplacement des nœuds en fonction de leur saillance, la correspondance et la discrimination des poses s'en trouvent améliorées. Les régions saillantes sont plus discriminantes pour l'estimation de l'orientation de la tête que les régions non saillantes.

Avec une erreur moyenne de 10.1 degrés en pan et 12.6 degrés en tilt sur les sujets inconnus, notre système offre une performance comparable à celle obtenue par les humains. L'erreur moyenne par pose est illustrée sur la figure 2.16. Les erreurs obtenues par notre algorithme sont plus homogènes que celles obtenues par les humains. Notre système est meilleur pour reconnaître les poses intermédiaires, mais les humains restent meilleurs pour reconnaître les poses face et profil. Cela confirme que le système visuel humain utlise les poses face et profil comme poses clés.

Les graphes saillants améliorent les résultats obtenus par les mémoires autoassociatives linéaires. La complexité linéaire des graphes saillants leur permet de prendre le relais sur les mémoires autoassociatives linéaires, qui ont une complexité quadratique, quand la résolution de l'image augmente. Notre système d'estimation de la pose du visage utilise les apparences globale et locale des images, est complètement automatique, n'utilise ni d'heuristique, ni de connaissances préalables sur le visage, ne nécessite pas d'étiquetage manuel et peut être adapté à l'estimation de l'orientation d'autres objets déformables.

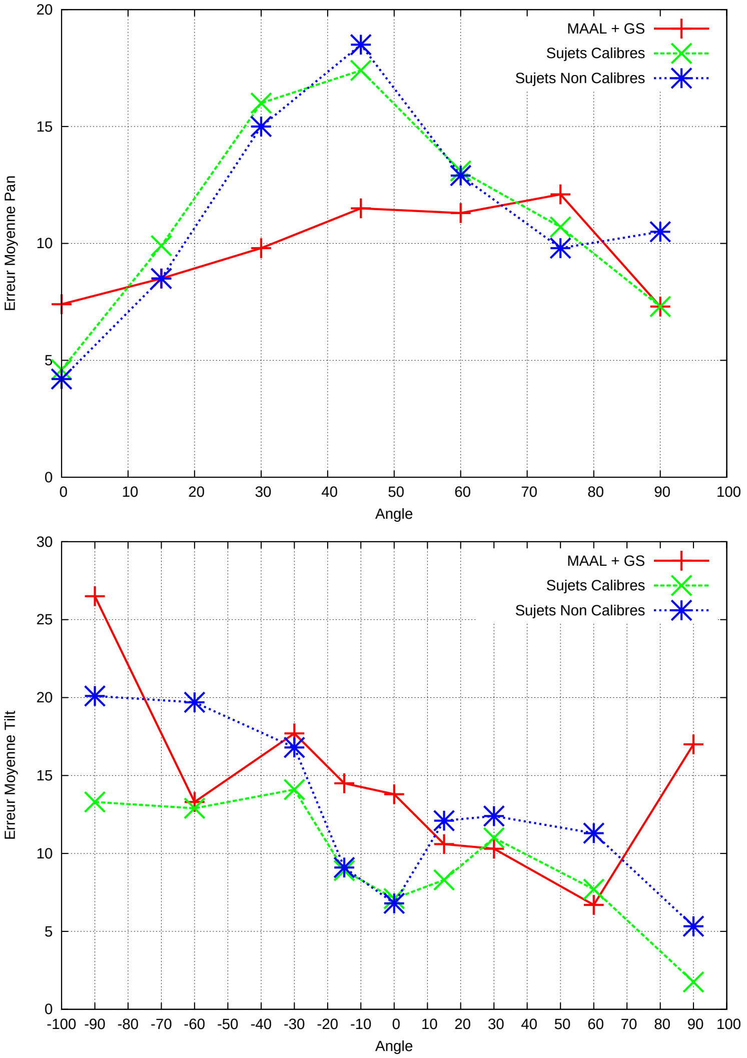

FIG. 2.16 – Erreur moyenne par pose sur les axes pan et tilt.

Chapitre 3

Conclusions et perspectives

En se basant sur les approches globales et locales de Vision par Ordinateur, nous avons approfondi un système d'estimation d'orientation de la tête utilisant les mémoires linéaires autoassociatives et les graphes saillants de champs réceptifs gaussiens. Apprendre des prototypes de poses à partir d'images de visage non contraintes est un moyen simple, rapide et efficace pour obtenir une première estimation de l'orientation. Avec cette approche, les angles pan et tilt peuvent être appris séparément. Cette estimation est améliorée en utilisant des graphes dont les nœuds contiennent des vecteurs de champs réceptifs gaussiens. Les nœuds peuvent être déplacés localement de manière à maximiser la ressemblance tout en conservant leurs relations spatiales. L'estimation de la pose est raffinée en recherchant le modèle de graphe le plus similaire parmi les poses voisines de celle trouvée en première estimation. La performance globale est comparable à la performance humaine.

3.1 Résultats principaux

Dans nos expériences, le groupe de personnes a effectué une erreur moyenne de 11.85^o en pan et 11.04^o en tilt. Nous avons découvert un résultat intéressant sur l'angle pan. Les personnes ont une bonne aptitude à reconnaître les poses face et profil, mais les performances se dégradent sur les poses intermédiaires. L'angle pan semble plus naturel à estimer. L'erreur minimale se trouve pour la pose 0^o, ce qui correspond à la vue de face. Ces résultats suggèrent que le système visuel humain utilise face et pofil comme des poses clés, comme stipulé dans [65]. L'âge des sujets ne semble pas influencer le résultat.

Dans notre méthode, la région de l'image correspondant au visage est normalisée en position, taille et inclinaison dans une image de petite résolution en utilisant un système de suivi de visage. Les mémoires autoassociatives linéaires sont utilisées pour apprendre des prototypes d'orientations de la tête. Ces mémoires sont simples à construire, ne requièrent que peu de paramètres et sont adaptées pour l'estimation de la pose du visage sur des sujets connus et inconnus. Les prototypes peuvent être appris en utilisant un ou deux axes. Avec une erreur moyenne de moins de 10^o en pan et en tilt pour des sujets connus, notre méthode est plus performante que les réseaux de neurones [152], l'Analyse par Composantes Principales et les modèles de tenseurs [145]. Nous obtenons une erreur moyenne de 10^o en pan et 16^o en tilt sur des sujets inconnus. Apprendre les angles pan et tilt séparément réduit le nombre de prototypes à utiliser tout en ne dégradant pas la performance. Ces résultats sont obtenus sur des images non alignées. Les prototypes de poses du visage peuvent être sauvegardés et chargés ultérieurement pour d'autres applications. Notre algorithme de première estimation de la pose fonctionne à 15 images par seconde, ce qui est suffisant pour des applications vidéo telles que les interactions homme-machine, la vidéosurveillance et les environnements intelligents.

Cette première estimation est raffinée en décrivant les images du visage par des champs réceptifs gaussiens normalisés à leurs échelles intrinsèques. Les dérivées gaussiennes décrivent l'apparence de voisinages de pixels et présentent un moyen efficace pour détecter les traits du visage indépendamment de leur taille et de leur illumination. De plus, elles ont des propriétés d'invariance intéressantes. Les images de visage sont ainsi décrites par des vecteurs de faible dimension. Les régions saillantes du visage sont découvertes en analysant les régions qui partagent une apparence similaire sur un rayon limité. Nous trouvons que les principaux traits saillants du visage sont : les yeux, le nez, la bouche et le contour du visage. Ces résultats ressemblent aux traits faciaux regardés par les humains selon les études de Yarbus [165].

Les graphes de champs réceptifs gaussiens améliorent l'estimation de la pose obtenue en première estimation. La structure de graphe décrit et l'apparence des voisinages de pixels, et leurs relations géométriques dans l'image. Les résultats sont meilleurs en effectuant un clustering hiérarchique en chaque nœud du graphe. Les graphes recouvrant la totalité de l'image du visage sont plus performants que ceux ne recouvrant qu'une partie du visage. Plus grande est la portion d'image recouverte, plus importantes sont les relations géométriques. De plus, paramétrer le déplacement local maximal d'un nœud en fonction de sa saillance résulte en une meilleure estimation que fixer un même déplacement local pour chaque nœud. Un nœud placé sur un trait saillant du visage représente un point pertinent pour la pose considérée et ne doit pas trop se déplacer de son emplacement initial. Au contraire, un nœud placé dans une région peu saillante ne représente pas de point pertinent pour la pose et peut bouger. Les graphes saillants améliorent surtout la performance en tilt, peu en pan. Ceci montre que l'information de l'inclinaison horizontale de la tête est fournie en majeure partie par l'assymétrie du visage, contenue dans l'apparence globale. En utilisant cette méthode, nous obtenons un système d'estimation de la pose de la tête avec une exactitude de 10^o en pan et 12^o en tilt sur des sujets inconnus. Cet algorithme ne requiert pas d'heuristique, d'annotation manuelle ou de connaissances préalables sur le visage et peut être adapté pour estimer l'orientation ou la configuration d'autres objets déformables.

L'estimation de pose du visage est testée sur des séquences vidéo de la IST CHIL Pointing Database. Le contexte temporel offre un gain en temps de calcul considérable. La pose du visage sur l'image suivante se trouve dans le voisinage de la pose courante. Nous avons obtenu une erreur moyenne de 22.5^o en pan. L'orientation de la tête est souvent utilisée par les humains pour estimer le focus visuel d'attention sur des images fixes et des séquences vidéo. En particulier, nos expériences ont montré que son inclinaision horizontale était plus pertinente que son inclinaison verticale pour les humains. Nous avons conçu un système permettant de délivrer une performance similaire à celle des humains sur les mêmes données. Les résultats que nous avons obtenus montrent que notre approche est adaptée à l'estimation de l'orientation de la tête dans des environnements intelligents, pour prédire les interactions entre personnes et objets. Notre algorithme peut aussi servir d'entrée pour des systèmes attentionnels [85].

3.2 Extensions

Notre système d'estimation de la pose de la tête a démontré de bonnes performances sur des images fixes et des séquences vidéo. La première étape de la méthode est de normaliser la région de l'image correspondant au visage en taille et en inclinaison pour travailler sur des imagettes de visage. En conséquence, le temps de calcul devient indépendant de la taille de l'image source. Néanmoins, le suivi de visage peut également introduire un problème. La hauteur du cou diffère d'une personne à une autre. Ceci produit des variations sur les imagettes de visage et peut biaiser l'estimation de l'angle d'inclinaison verticale tilt. De plus, comme le système de suivi est basé sur la chrominance, il peut parfois suivre une région différente d'un visage mais dont la chrominance est similaire à celle de la peau humaine. Il peut également capturer les régions adjacentes au visage de même chrominance, comme par exemple une personne mettant ses mains près du visage. L'algorithme Raster-Scan developpé par Peters [109] peut localiser la région du visage en déplaçant le graphe sans déplacer ses nœuds localement. Cependant, pour délimiter correctement la région, la taille du visage doit être connue. En mettant le système de suivi et le Raster-Scan dans une boucle, la normalisation et l'alignement pourraient être améliorés.

En suivant la même idée, les graphes saillants pourraient voir si un point du visage est caché ou non. En supprimant la contribution du nœud correspondant à ce point, l'estimation pourrait être améliorée. Si jamais il y a trop de points cachés, on ne se base que sur le résultat des mémoires autoassociatives linéaires, robustes à l'occlusion partielle.

De la même façon que nous détectons les régions saillantes du visage comme des blobs d'apparence à l'échelle intrinsèque, nous devrions décrire également les crêtes du visage comme des crêtes d'apparence. Une nouvelle méthode de description de crêtes basée sur l'énergie du laplacien a été récemment démontrée [144]. Ces crêtes pourraient servir d'arêtes dans les graphes. Combiner nœuds et arêtes pourrait augmenter la performance de l'estimation.

Les mémoires autoassociatives linéaires sont perturbées par les changements d'illumination globaux, mais pas locaux. Au contraire, les champs réceptifs gaussiens sont perturbés par les changements d'illumination locaux, mais pas globaux. En intégrant ces deux approches dans une boucle, chacune pourrait donner un indice de confiance en son estimation. En prenant en compte ces indices, nous pourrions choisir quelles méthodes utiliser.

Augmenter la résolution des imagettes de visage augmente la précision et peut permettre l'estimation continue de la pose. Dans notre étude, seules des poses discrètes ont été entraînées et sélectionnées par choix du meilleur score. Les poses continues pourraient être obtenues par interpolation des meilleurs scores. Ceux obtenus par les poses avoisinantes de la pose gagnante constituent un bon choix pour l'interpolation.

Une base de données vidéo pour l'estimation de l'orientation de la tête à partir de 4 caméras est apparue récemment [152]. La pose du visage du confériencier est annotée manuellement dans les 8 directions cardinales. Les estimations de la pose à partir de 4 points de vues différents pourraient être combinées pour obtenir une estimation plus fiable.

En conclusion, nous ne devons pas oublier que l'orientation de la tête ne représente qu'une partie de l'attention humaine. La position de la pupille sur l'œil contribue à la direction du regard, mais ne peut être détectée que sur des images de haute résolution. Cependant, l'attention humaine est difficile à définir parce qu'elle comprend aussi bien le foyer d'attention visuel que le foyer d'attention auditive, l'intention, la nature et l'implication du sujet dans sa tâche. Les systèmes pour estimer l'attention commencent à apparaître, et l'estimation de la pose de la tête peut servir d'entrée à de tels systèmes [85]. Ces approches peuvent fournir des informations importantes pour l'Interaction Homme-Machine et l'observation d'activités humaines.

Part II

Machine Observation of the Direction of Human Visual Attention - Complete English Version

Chapter 1

Introduction

Informatic technologies are autistic. Few technologies currently exist to endow artificial systems with a reliable ability to sense social interactions, whether interactions occur between humans or between human and machine. As a consequence of the inability to evaluate user attention or interest, artificial systems often distract people with inappropriate actions and have little or no ability to use human interaction to correct their behaviour.

An important aspect of social interaction is the ability to observe human interest and attention. Humans locate the focus of attention of people to a large extent by observing their faces and their gazes. To a large part, interest and attention of a person can be estimated from the orientation of the head.

In this thesis, we adress the problem of head pose, or orientation, estimation on unconstrained single images. Head orientation is determined by three angles: roll, tilt and pan. The roll angle represents the person's head inclination with regard to the body and varies around the longitudinal axis. The tilt angle stands for the vertical inclination of the face and varies around the lateral axis, when a person looks up and down. This angle is the most difficult of the three to estimate. The pan angle corresponds to the horizontal inclination of the face and varies around the vertical axis, when the person turns his head left and right. Our goal is to propose methods to estimate these angles, as a first step towards estimating visual focus of attention.

Many of the techniques proposed in the literature for estimating gaze and head pose orientation employ special equipment, such as infrared illumination, electro-oculography, head mounted devices or specific contact lenses [59, 167, 33]. Commercial systems are available using active cameras and stereo vision [162, 96, 120]. Although such techniques deliver high precision, they tend to be expensive and too intrusive for many applications. Computer vision-based systems present a less intrusive approach. We are particularly interested in estimating the head orientation in order to estimate human visual attention in intelligent environments.

Our goal is to propose a non intrusive method for head pose estimation that does not require specific equipement. In particular, we are interested in automatic technologies that are robust to identity and operating under unconstrained imaging conditions. Humans can deliver a rough estimate of the pose of an object from a single image. Furthermore, head pose estimation from single images is the first step for accurate head pose estimation from several images.

Approaches to head pose estimation from single images can be divided in 4 main families : 2D geometrical approaches, 3D geometrical approaches, facial transformation based approaches and template based approaches. 2D geometrical approaches use detected facial landmarks to find correspondences and compute pan and tilt angles. These methods are accurate but requires high resolution of the face and cannot accomodate wide head movements. 3D geometrical approaches apply a 3D model of the head to recover its 3D rotation. Such techniques are more accurate than 2D methods, but require more computational time as well as a strong prior knowledge of the geometrical structure of the face. Facial-transformation-based approaches use facial properties to obtain an estimation of the head orientation. Such methods are easy to compute, but tend to be unstable and identity-dependant. Template-based approaches consider the problem as a classification problem solved by matching the current image with the most similar template. Such methods are very fast, but can only deliver a coarse estimation of the pose and the user has no feedback about what happened if the system fails. In this thesis, we develop a hybrid coarse-to-fine approach for head orientation estimation whose performance are comparable to human performance.

1.1 Coarse-to-fine head pose estimation

In this thesis, we propose a fully automatic approach for head pose estimation on images taken under unconstrained imaging conditions independant of the identity of the person. This approach combines the advantages of global approaches which use the appearance of the whole image for classification and local approaches which use information contained in neighbourhood of pixels and their relations in the image without using any heuristics or prior knowledge about the face. We present a coarse-to-fine head pose estimation system based on linear auto-associative memories and Gaussian receptive fields graphs robust to changes in identity. Our method works on non-aligned face images as in real conditions and its performance is comparable to human performance.

To properly measure the performance of a head orientation algorithm, we need to evaluate the method on a representative database. Different methods are often tested with different databases, which makes fair comparison difficult. Such a database should contain adequate data and a full range of poses. This allows us to evaluate the behaviour of the method on each pan and tilt angle. Such a database should also be symmetrically and sufficiently sampled. If the method works well with many angles, it can be adapted to real-time head pose tracking in real conditions, in which the head angle is not discrete, but continuous.

Our experiments use the Pointing 2004 Head Pose Image Database [39], a densly sampled database covering a half-sphere of poses from -90 to +90 degrees in pan and tilt angles. The head pose database consists in 15 sets of images. Each set contains 2 series of 93 images of the same person at different orientations. Training and testing can be done either on known users by applying a cross-validation on both sets or unknown users by applying a Jack-Knife algorithm on persons.

Humans are known to estimate visual focus of attention through the head pose, but their abilities for estimating human head orientation are largely unknown. It is unclear whether humans have a natural ability to estimate the head pose of people in single images, or whether people must be trained for such a task using sample annotated images. Furthermore, we do not know the accuracy with which a person can deliver estimates for pan and tilt angles of an observed head. Kersten [65] reports that front and profile poses are used as key poses by the human brain. As a benchmark, or reference, we evaluated the ability of a group of people on head orientation estimation from a sample of the Pointing'04 Head Pose Image Database. These experiments show that our proposed method yields results similar to human abilities.

With our method, a coarse estimation of the head orientation is obtained by searching the best prototypes which match the current image. We combine this with a method based on defining salient facial regions relevant for each head pose. Salient regions are locally described by Gaussian receptive fields normalized at intrinsic scale, given by the local maximum of the normalized Laplacian. These descriptors have interesting properties and are less expensive to compute than Gabor wavelets. Salient facial regions found by Gaussian receptive fields enable the construction of a model graph for each pose. Each node of the graph can be locally displaced according to its saliency within the image and is labelled by a probability density function of normalized Gaussian receptive field vectors hierarchically clustered to represent various aspects of the same feature under identity changes. Linear auto-associative memories deliver a coarse estimation of the pose. This result is refined by searching among the coarse pose neighbors the salient grid graph thus providing the best match. The pose associated with its model graph is selected as the head pose of the person in the image.

1.2 Contributions of the dissertation

Experiments show that humans perform well at recognizing frontal and profile views of faces, but not for intermediate views. In our experiments, the human average error per pose is 11.85^o in pan and 11.04^o in tilt. Minimum human error in pan is found at 0 degrees, which corresponds to a straight or frontal view. Pan angle appears to be more natural to estimate. These results suggest that the human visual system uses front and profile views as key poses, as proposed in [65].

In our method, face region images are normalized to produce low resolution imagettes using a robust face tracker. Linear auto-associative memories are used for learning prototypes of head pose images. Because such memories are relatively simple to construct and require few parameters, they appear to be well suited for head orientation estimation for both known and unknown subjects. Prototypes are trained either on one or two axis. With an average error of less than 10^o in pan and tilt angles on known subjects, our method performs better than neural networks [152], PCA and tensor models [145]. We achieve an error of 10^o in pan and 16^o in tilt for unknown subjects. Learning pan and tilt angles together does not increase much the performance, we thus learn pan and tilt separately, which reduces the number of prototypes used. Results show that our system can handle alignment problems. Head pose prototypes can be saved and restored for other applications. Our coarse head pose estimation algorithm runs at 15 frames per second, is reliable enough with video sequences for applications such as man-machine interactions, video surveillance and intelligent environments.

Head orientation estimation can be improved by describing face images using Gaussian receptive fields normalized to intrinsic scale. Gaussian derivatives describe the appearance of neighbourhoods of pixels and are an efficient means to compute scale and illumination robust local features. Furthermore, they have interesting invariance properties. Face images are described using low dimensional feature vectors. Salient facial regions of the face robust to identity and pose can be recovered by analyzing regions which share the same apperance on a limited radius. We found that the salient facial features detected by normalized Gaussian receptive fields were eyes, nose, mouth and face contour. These results resemble those obtained by humans according in studies described by Yarbus [165].

Gaussian receptive field grid graphs refine the pose obtained by the coarse estimate system. The graph structure describes both neighbourhoods of pixel appearance and their geometric relations in the image. Describing the appearance of each node with hierarchical clustering gives better results. We also found that graphs covering the whole face image provide better performance than graphs applied only on parts of the image. The larger the region covered by the graph, the more geometric relation information it captures. Furthermore, setting the local maximum displacement for nodes according to their saliency provides better results than having a fixed value. A node placed at a salient fixation represents something relevant for the pose and does not need to move too much from its original location. On the other hand, a node placed at a non-salient location represents an irrelevant feature and can be moved with a maximal displacement equal to the distance between 2 nodes, in order to keep geometric relation. Using this method, we obtain a coarse-to-fine head pose estimation with 10^o in pan and 12^o in tilt for unknown users. This algorithm does not use any heuristics, manual annotation or prior knowledge on the face and can be adapted to estimate the pose of configuration of other deformable objects.

Head pose estimation on video sequences has been tested using the IST CHIL Pointing Database. The temporal context provides a crucial gain of performance as well as a significant computational time reduction. The head pose at the next frame is expected to be found in neighbouring poses of the previous pose. We found an average error of 22.5^o in pan. Subjects are different from the Pointing'04 database. Head pose estimation can also serve as an entry for attentional systems [85].

1.3 Overview of the dissertation

Chapter 2 gives an overview of existing vision methods for estimating head orientation. Studies has shown that visual focus of attention has more influence than auditive focus of attention

[129]. The direction of people's gaze in images can be estimated from the head orientation and the position of the pupils with regard to the eyes. During a rapid gaze, head rotation is limited because rotating the eyes is faster and requires less energy than rotating the head. However, the human ocular muscles require more effort when the gaze is directed off center. Thus the head tends to turn to center the eyes in order to relieve effort of the eye muscles during longer fixations. Head orientation is a reliable indicator of sustained attention.

This chapter details principal aspects of the head pose estimation task. Many head pose estimation algorithms work with multiplesensors, including infrared illumination, stereo images or active cameras. These approaches deliver an accurate estimate in their estimation, but require specific equipment or are excessively intrusive. Our goal is to propose a head pose estimation algorithm without specific equipement that is as non-intrusive as possible. We target head pose estimation with single images. Estimating the head orientation of a person in general is a problem with many facets. Unlike many computer vision problems, there is no unified framework for this task. Almost every author proposes his own framework and metric.

Chapter 3 adresses the problem of human abilities to head pose estimation. People can give a rough estimate of head orientation from single images. However, the psycho-physical basis for this task remains unknown. We do not know whether humans have a natural ability to estimate the head pose of people in single images, or whether people must be trained for such a task using sample annotated images. Furthermore, we do not know the precision at which a person can deliver values for pan and tilt angles of the head either. Kersten [65] reports that front and profile poses are particularly well recognized by humans. These poses are used as key poses by the human brain. We measure human performance for this task using a densly sampled database of discrete head poses, the Pointing '04 Database [39]. The goal of this chapter is to determine which kind of precision can be expected from an head orientation estimation system in Man-Machine Interaction applications.

We have evaluated the performance of a group of people on head orientation estimation. This experiment investigated which angle is the most relevant for people. We measure the performance of a group of 72 human subjects on head pose estimation. In our experiment, we tested 36 men and 36 women, ranging from age 15 to 80. The people are asked to examine the image, and to circle an answer on a sheet of paper corresponding to their best estimate of the observed pose. Images from the Pointing 2004 Head Pose Image Database were presented in random order to the subject for 7 seconds, with a different order for each subject.

In our experiments, human displayed an average error of 11.85^o in pan and 11.04^o in tilt. Estimation of head pan angle appears to be natural for humans, whereas tilt angle estimation is not. In situations where people talk to each other, pan angle provides good cues on visual focus of attention [128]. This fact is even more relevant when people are sitting, because theirs heads are roughly at the same height. We also found that humans perform well at recognizing front and profile views, but not for intermediate views. The average error per pose in pan can be roughly modelled by a Gaussian centered at 45 degrees. These results tend to show that the human brain uses front and profile views as key poses, as suggested in [65].

Chapter 4 serves to introduce the robust face tracker and detector system used in our experiments. Rather than cropping and aligning manually face image regions, we detect them by using this system. This algorithm is an initial step to detection and normalization of a face region in video sequences and single images. Our tracker uses pixel level detection of skin colored regions using a Bayesian estimation of the probability that a pixel corresponds to skin based on its chrominance. A prediction-verification step is done using a zeroth order Kalman filter [61]. The process runs at video-rate.

Face detection and normalization is a crucial preprocessing step for head pose estimation. Once the face region is tracked, first and second moments are used to normalize facial images in size and slant orientation ans project them onto low resolution imagettes. A result of this normalization is that all images in the training data have the same size, which makes computation time of further operations independant from the original image size. In addition, such normalization allows us to have facial regions to the same location in the imagette for a given head pose.

Chapter 5 explains the coarse head pose estimation procedure. Normalized face imagettes of the same head pose are used to train an auto-associative memory that acts as a head pose prototype. Linear auto-associative memories are a particular case of one-layer linear neural networks where input patterns are associated with each other. Auto-associative memories associate images with their respective class, even when the image has been degraded or partially occluded. Such networks were first introduced by Kohonen [70] to save and recall images.

To enhance the accuracy of estimation, we use the Widrow-Hoff correction rule to train head pose prototypes. The Widrow-Hoff correction rule is a local supervised learning rule aiming at increasing the performance of associators [148]. Only few parameters are required. Head poses are trained either separately or together. Classification of head poses is obtained by comparing normalized face imagettes with those reconstructed by the prototype. The head pose whose prototype obtains the highest score is selected.

Training and testing can be done on known or unknown users. We obtain results comparable to human performance in both pan and tilt angles. Learning poses and pan and tilt angles separately provides a significant gain of computational time without loss of performance. Results obtained on unknown users show that our system generalizes well to previous unseen subjects and is robust to identity.

Chapter 6 describes perception of face images with Gaussian receptive fields formed from Gaussian derivatives. A receptive field is a local linear function that reponds to intensity changes of a certain form and orientation at different scales in images. Features of intermediate complexity that are robust to scale, illumination and position changes are used by primates for vision and object recognition. Our objective is to design such local descriptors. Gaussian derivatives are an efficient means of describing appearance of neighbourhoods with scale and illumination robust local features. Furthermore, they have interesting invariance properties, such as separability, scalability and differentiability.

Lindeberg [78] proposes a method to select appropriate local scales to describe image features. For a given image region, these relevant scales are called intrinsic scales. Local maximas in the scale profile computed at every neighborhood of pixels provides one or more intrinsic scales. The scale profile of a feature point is obtained by collecting its responses to the normalized Laplacian energy at varying scales. Scale invariant receptive fields are obtained by projection of image neighbourhoods on a vector of Gaussian derivatives normalized with their intrinsic scales. Every pixel of the face image is therefore analyzed at an appropriate scale.

Face images and their salient regions are described using low dimensional feature vectors. We propose the following definition for salient regions: A region is salient on an image when its neighbouring pixels share a similar appearance only over a limited radius. Gaussian normalized receptive fields appear to be a good detector for salient facial regions robust to illumination, pose and identity.

Chapter 7 explains the adaptation of elastic bunch graphs introduced by Von der Malsburg et al. [158] to Gaussian receptive field graphs. Elastic bunch graphs were initially developed for face recognition. This structure has interesting properties for image matching under changing conditions. A graph is described by a set of nodes labelled by their descriptors and their edges. In the literature, Gabor Wavelets which describe both geometrical and textural information on the image are often used as descriptors. However, we have found that Gaussian derivatives provide similar information at a much lower computational cost.

Head pose estimation has been demonstrated on varying number of poses using elastic bunch graphs [24, 160]. Nevertheless, such systems require high resolution image of the face. Furthermore, such graphs are constructed empirically for each pose. Training a new person requires to manually label graph nodes and edges on all face images. As we do not want to use manual annotation in our system, we apply grid graphs to recover head pose from facial features.

The same facial point can have different appearances with regard to the person. The result is an assembly of clouds of points in the feature space on every node of the graph. To model such different aspects of the same feature, we apply a hierarchical clustering to the receptive fields vector responses for the same node. Each node of the graph can be locally displaced according to its saliency in the image. One salient grid graph is constructed per pose. The head pose estimation system based on linear auto-associative memories delivers a coarse estimate for the pose. We refine this estimate by searching for the best salient grid graph from its neighboring poses. The pose whose probability gives the best score is selected as the head pose. We obtained a coarse-to-fine head pose estimation with 10^o in pan and 12^o in tilt for unknown users, achieving a precision comparable to human performance.

Chapter 8 presents extensions of our system. The first part of the chapter describes the use of linear auto-associative memories on people detection in video surveillance sequences. This is the first step to person and face tracking. Our method works at low resolution and requires very few parameters. This approach inherits strong points of appearance based vision: simplicity and independance to the detection technique. We compare the performance of our system to three other statistical algorithms using the IST CAVIAR database.

Head pose estimation on video sequences is developed in the second part of this section. Head pose prototypes are created using linear auto-associative memories trained separately in pan and tilt. The use of video sequences introduces a new element to the task: the temporal context. Temporal context provides an important gain in performance as well as a significant reduction in computational time. The head pose at the next frame is expected to be found in neighbouring poses of the previous pose. With the use of head pose prototypes, we can restrict the research of the current head pose to neighbour poses, which is less time consuming. We use the IST CHIL database to test our system.

The third part of the chapter extends the use of head orientation estimation to focus of attention detection and privacy violation. Head pose estimation can provide input to attentional systems. The attentional model developed by Maisonnasse [85] can be used to detect when someone pays attention to a device and transgresses privacy. The PRIMA Robust Tracker [12] is used to track people and objects. The system detects entities in the environment and projects their positions to environmental coordinates using an homography. Head pose estimation could be a good indicator of people's attention and privacy violation.

Chapter 9 concludes this thesis by summarizing the main results and perspectives.

Chapter 2

Estimating visual focus of attention

Visual focus of attention contributes more to human attention than auditive focus of attention [129]. In addition, many studies suggest that human gaze provides useful cues about focus of attention [130, 75]. For this reason, we are interested in techniques for estimating and tracking head orientation. The first part of this chapter presents the gaze and head pose estimation problems and its applications. The second part concerns important aspects of the problem: face image resolution, accuracy of estimation, robustness to identity and choice of a database for performance evaluation. A literature review on head pose estimation is presented in the third part. The final section motivates the coarse-to-fine approach developed in this thesis.

2.1 Estimating gaze of people in images

The direction of people's gaze as captured in images can be estimated from the head orientation and the position of the pupils with regard to the eyes. During a quick glance, there is little or no head rotation. Eye rotations may be as fast as 500 degrees per second and require relatively little energy. Thus rapid glances tend to depend only on eye motion. This is the case, for example, when a person is scanning a web page or reading a book.

Despite this fact, it is not surprising that most studies show that the head orientation contributes generally more than eye movement to gaze direction. Stiefelhagen [138, 130] reports that in meeting situations, people turn their heads rather than their eyes in 69% of time and the head orientation direction is equal to the gaze in 89% of time. This fact is easily understandable if we consider that while eye motion is very fast, the eye muscles requires energy to remain off center during a prolonged time, and head motion compensates for this effort. Thus humans tend to rotate their head to recenter the eyes during longer gazes characteristic of sustained attention.

Kingstone [66] asked subjects to gaze at a target after seeing an image of someone looking elsewhere. Most subjects had an unvoluntarily reflex to change their gazes to the scene position where the person on the image was looking. Langton [74] showed people images of subjects having their head orientation identical or opposite to their gaze. He concluded that people take more time estimating the gaze of subjects when the head orientation is different from the eye gaze direction. Head pose perception strongly influences the human perception of gaze. Moreover, eye blinking can disrupt eye trackers and prevents pupil detection. In his studies, Stiefelhagen [138] reports that eye blinking happens 20% of time in meeting situations. In any case, detecting pupils on images requires a relatively high resolution image of the face and often requires cameras directly in front of the eyes to have an accurate image of the pupil. Using head pose as an indicator of attention allows us to avoid such intrusive equipment.

2.1.1 Definition of the problem

The goal of this study is to estimate a person's head pose, or head orientation, with low resolution unconstrained single images. Head pose is determined by 3 angles: roll (also called slant), tilt (also called pitch) and pan (also called yaw). These 3 angles are illustrated in figure 2.1. The roll angle represents the person's head inclination with regard to the body and varies around the longitudinal (or forward) axis. The tilt angle stands for the vertical inclination of the face, when a person looks up and down, and varies around the lateral (or sideways). This angle is the most difficult to estimate. The pan angle represents the horizontal inclination of the face, when the person turns his head left and right, and varies around the vertical axis. These 3 angles cover the complete 3D movement of the human head.

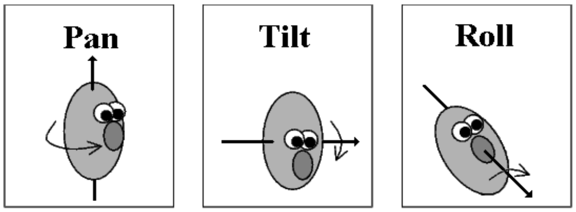

Figure 2.1: The 3 rotation angles of the human head [25].

2.1.2 Why monocular vision ?

A variety of tehniques may be used to estimate gaze and head pose orientation. Infrared illumination presents the advantage of accurately localizing the pupils of people in an image [59, 167]. However, such methods are intrusive as they require irradiation of the eye with infrared illumination. Unvolontary eye movements during infrared illumination may expose the retina, resulting in small eye lesions and may pose a health hazard [124]. There are other commercially avalaible systems to track eye-gaze. Electro-oculography measures the potential of the electro-static field rotating around the eyeball to detect the position of the pupil [33]. However, some studies [11] report problems with these techniques. Lighting adaptations of the eye change the value of the potential, which causes this method to fail in case of illumination variations. One method is to use specially constructed contact lenses. Although such techniques deliver high precision, they are too intrusive for many applications. Computer Vision can avoid such intrusive approaches.

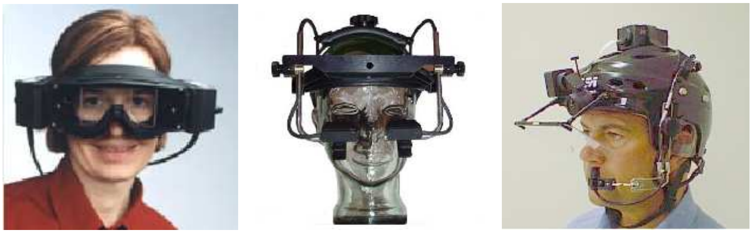

Figure 2.2: Example of commercial eye gaze tracking devices [129]

Many computer vision techniques are inspired from theories of human vision and work with stereo images [162, 96, 63] or images taken from active cameras [21, 87]. As we have seen, these approaches can provide an accurate estimate of gaze direction, but require specific equipment. Some complex systems, such as FaceLAB [120] report less than one degree accuracy, but use several sensors, high quality cameras and are very expensive. Our goal is to propose a head pose estimation algorithm without specific equipement and as non-intrusive as possible.

Humans can deliver a rough estimate of the pose from a single image. Furthermore, head pose estimation from single images are the first step for intelligent multi-camera systems. Accurate pose estimation from a single image can improve pose estimation from multiple cameras.

2.1.3 Applications

The task of estimating and tracking focus of attention can serve as an important component for systems for man-machine interaction, video conferencing, lecture recording, driver monitoring, video surveillance and meeting analysis. Human head pose is associated with actions and interpreted differently with regard to the context. It can also be useful for other computer vision tasks where the effect of the head pose needs to be compensated. Such tasks include person identification and facial expression analysis. Using local or global approaches for these problems cannot prevent similarity measures between 2 different views of the same person from decreasing as the difference in head pose increases. Therefore the head pose needs to be estimated prior to the recognition or facial analysis process.

The head pose of a person can provide important cues for estimating visual focus of attention in meetings, for example if the speaker is facing the audience or his slides. It can also serves as hand free cursor [142] control for man machine interfaces for handicapped people. Head pose estimation is also used for driver monitoring [53, 8, 120]. When a driver becomes tired, his ability to maintain visual attention degrades. The system detects such signs and tells him to stop and have a rest. Another area of application is to study where people look to analyze

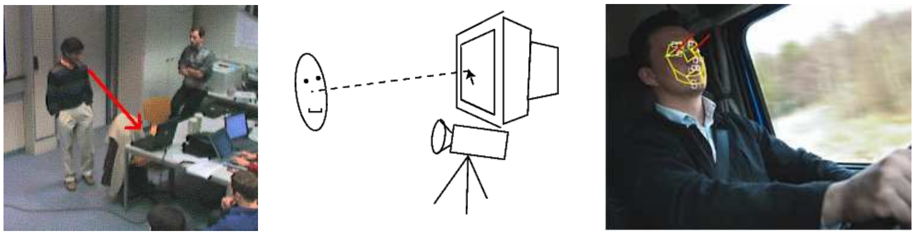

Figure 2.3: Example of applications of head pose estimation systems

their attention during human to machine or human to human interactions. The main advantage of computer vision based approaches is that, as the subject does not need to wear any specific device, his head and eyes movements are natural. These kinds of experiments are useful to know which part of a web page or of a shop window is the most relevant. A study has been made on air-traffic controllers [9] to see which screens were the most observed. Head pose can be used to represent a perspective cursor for handicapped people to control multi displays interfaces [99].

We are particularly interested in estimating the head orientation in order to estimate human visual attention in smart environments. The direction of the head pose can serve as a good indicator to determine at which objects people are paying attention or which interactions can they have with objects and other people. Head orientation estimation has many applications in various domains, but is a very difficult problem.

2.2 Issues when estimating head pose from single images

Pose recognition for any class of objects must overcome many obstacles. The task is even more difficult in the case of deformable objects, and especially the human head. Estimating the head orientation of a person in general is a problem with many facets. Unlike many computer vision problems, there is no unified framework for this task. Almost every author presents a new framework and new metrics. In this section, we review important aspects of the head pose estimation problem. They appear both in estimation from case of single images and estimation from video sequences. These difficulties must be resolved for any system that seeks to estimate or track the 3D movements of the human face.

2.2.1 Image resolution

Any head pose estimation system requires a minimal face image resolution to work. This minimal resolution varies greatly from one technique to another and is not always made explicit in the literature. Some systems require high resolution images of the face region (500x500 pixels), while others can accomodate low resolutions (32x32 pixels). However, no known system has been demonstrated to estimate head pose with images containing less than 10x10 pixels. Even the human eye is unable to tell where the subject is looking at such a low resolution. This suggest that the required face image resolution is related to another issue: what accuracy in the head orientation estimation can be expected ?

2.2.2 Accuracy of estimation

The accuracy of estimation is generally the first result of any head pose tracking system. Every serious work in the literature dealing with head orientation delivers a value for accuracy, generally in degrees. With this value, the reader has an idea of the quality of the system. So we should think it would be sufficient enough to compare the given accuracy in each paper to obtain the best head pose estimation system. However, even when the accuracy is specified, the method used to determine accuracy is not always explicitly stated.

There is no general or predefined metric for the head pose estimation task. Furthermore, the range of poses is sometimes not specified. Having a better accuracy over a smaller range of angles is easier than for larger ranges of angles. For example, a system delivering an accuracy of 5 degrees and working for pan angles from -20 to +20 degrees is not really more capable than a system delivering a accuracy of 10 degrees and working for pan angles from -90 to +90 degrees. The acurracy of estimation leads us to the training and test data issue.

2.2.3 Robustness to identity

Head pose estimation differs from object pose recognition in a number of ways. As mentionned earlier, the human face is a deformable object and can have many expressions. One of the most challenging properties of human faces is that their appearance can vary significantly from one person to another. Thus, intrinsic facial characteristics must be separated from head pose. The variety of appearance of skin colour, the chin and cheeks make robustness to identity for head pose estimation very difficult to obtain. Hair is the most variable part of the face and can occlude important facial features. For the same head pose, two persons may not have the same features visible. In addition, not only local aspects, but also the global aspect of the human face may vary over individuals. For example, the proportion of the neck with regard to the head and the dimensions of the face vary under face orientation and identity.

Many of these difficulties are greatly simplified when a system is intended for a particular user. This remark is not specific to head pose estimation and is generally valid for any man-machine interface algorithms. The robustness to identity is also linked to another point: the choice of a representative database.

2.2.4 Database Selection

Different methods are often tested with different databases. Furthermore, there exist very few databases annotated with head orientation. A good head pose database should contain the same amount of data for each pose. This allows us to see the behaviour of the method on each pan and tilt angle. This database should cover a wide range of poses. In many works in the literature, the capacity of the method to handle wide angles is sometimes not explicitly stated. Finally, such a database should be symmetrically and sufficiently sampled. If the method works well with many angles, it can be adapted to real-time head pose tracking in real conditions, in which the head angle is not discrete, but continuous.

Some commercial head mounted devices such as the FASTRAK system [56] developed by Polhemus Inc. provide measurements of the 3D rotation of the head with a precision of less than 3 degrees. Example images of people wearing this device are shown on figure 2.4. However, the device is visible in all face images, which influences head pose estimation on real conditions, because users do not usually wear such devices. The data used for training or testing is a crucial information which is sometimes not mentionned in the literature. When the database is presented, we must know which parts were used for the training and for the testing.

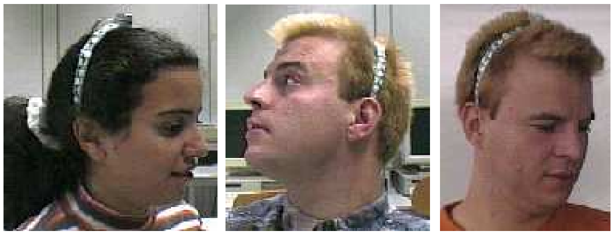

Figure 2.4: Sample images of people wearing the FASTRAK device [129]

2.3 Existing methods

Approaches to head pose estimation from single images can be divided into 4 main families : 2D geometrical approaches, 3D geometrical approaches, facial transformation based approaches and template based approaches. 2D geometrical approaches use detected facial landmarks to find correspondences and compute pan and tilt angles. 3D geometrical approaches apply a 3D model of the head to recover its 3D rotation. Facial transformation based approaches use facial properties to obtain an estimation of the head orientation. Template based approaches consider the problem as a classification problem by matching the current image with the most similar template. We explain our coarse-to-fine approach in the last section.

2.3.1 2D Geometrical approaches

2D Geometrical approaches represent the most intuitive way to estimate the head orientation. The main idea of these techniques is to detect a set of salient facial features and to use their respective location in the face region to compute pan and tilt angles. Some of these approaches use only the relative position of the eyes with regard to the face to estimate the head orientation. Eyes are either detected by iterative thresholding [133, 163, 134, 8, 16] or receptive fields [36, 37]. The pan angle α_h can be theoretically computed from the positions (x_{OK}, y_{OK}) of the two detected eyes, as shown in figure 2.5. By considering the face as represented by its centre of gravity (μ_x, μ_y) and its top view as represented by a circle of radius l, we define $l_0 \leq l$ as the face width at the height of the eyes y_{OK}. The value of l_0 can be calculated from the height of the eyes and the face ellipse. The distance ratio $\frac{x_{OK}-\mu_x}{l_0}$ standing for the location of the eye k with regard to the face is included between -1 and 1. By considering the top view of the face, the pan angle is computed with a simple trigonometric transform (2.4) :

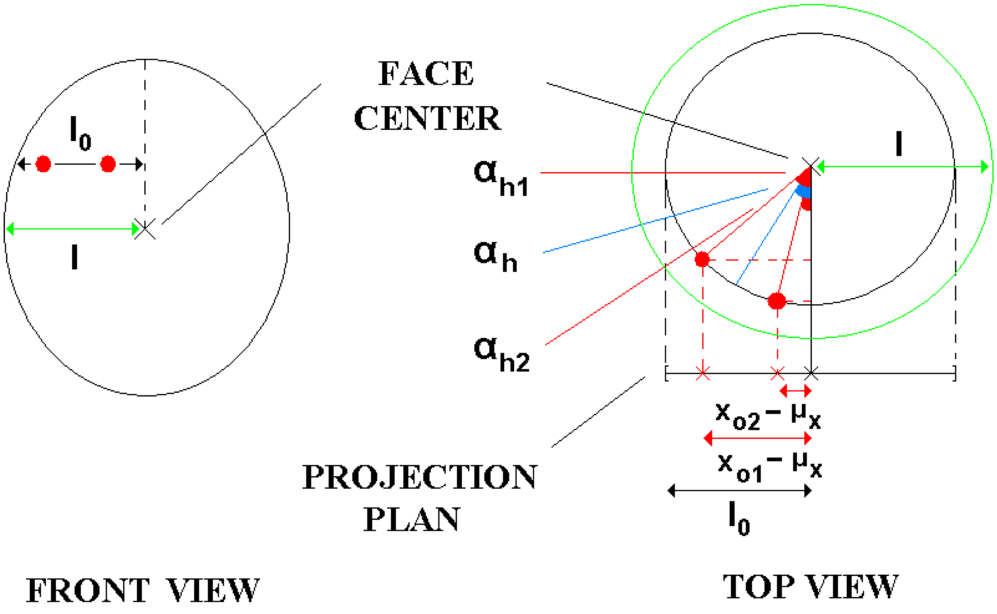

Figure 2.5: Direct pan angle computation from eyes position with regard to the face

$$x_{o1} - \mu_x = l_0 \cdot sin(\alpha_{h1}) \tag{2.1}$$

$$x_{o2} - \mu_x = l_0 \cdot sin(\alpha_{h2}) \tag{2.2}$$

The pan angle is defined as:

$$\alpha_h = \frac{\alpha_{h1} + \alpha_{h2}}{2} \tag{2.3}$$

and becomes:

$$\alpha_h = \frac{sin^{-1}(\frac{x_{O1}-\mu_x}{l_0}) + sin^{-1}(\frac{x_{O1}-\mu_x}{l_0})}{2} \tag{2.4}$$

However, both eyes need to be visible in the image to compute the pan angle. As the location of the eyes varies a lot from one person to another, the distance between them is not constant.

When only one eye is detected, the location of the other cannot be predicted, and the method becomes useless.

Both eyes are visible from frontal to near-profile poses, which corresponds to a pan angle between -45 and +45 degrees. This technique can not handle wide pan angles. Furthermore, as the vertical position of the eyes also varies from one person to another, the tilt angle cannot be computed just by using their location with regard to the face. Specifying a height μ_y at which everybody looks straightforward, which corresponds to $\alpha_v = 0$, is a difficult task. Eye position in the face varies substantially with identity and head pose, as shown by figure 2.6. A solution would be to calibrate people during system initialization, but this method is too intrusive. Head pose estimation from eye location also suffers from identity problem and is not valid over wide angles. Furthermore, the location of eyes is insufficient to estimate the tilt angle. Just as the pan angle requires at least 2 independant pieces of horizontal information, the tilt angle requires at least 2 independant pieces of vertical information to be calculated.

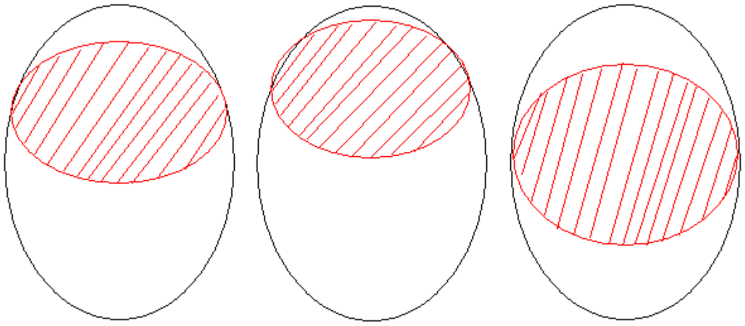

Figure 2.6: Eye position variation with regard to head pose for 3 different people

To compute the tilt angle, many authors suggest using other facial landmarks with addition to the eyes. Such facial landmarks are generally the mouth [169, 58, 126, 26, 47, 155], the nosebridge [62], the eyebrows [103], the nose [48, 17] or even nostrils [142, 143, 4]. A more complete feature based face model using six facial features was proposed by Gee & Cipolla in [31, 32]. Although using a larger number of features allows the computation of a more efficient estimate of pan and tilt angle, the location of these features shows considerable variations over the identity of the person and the head orientation, as we can see in figure 2.7. People must be calibrated at the initialization of the process. Furthermore, the problem of feature occlusion in wide angles is also present.

Because a precise calibration is hard to obtain, some approximations such as weak or affine perpective can be useful. The weak perspective hypothesis is a simple way to compute the 3D rotation of the head. It assumes that all feature points considered are coplanar. This assumption has been applied in addition to manual [164, 105] or automatic [30, 29, 80, 17] facial feature detection to head pose estimation. The set of feature points can also be labelled more accurately by a grid [24, 23, 90, 71] or detected by using mathematical properties, such as saddle points and blobs [107], Gabor jets [168] or maximas in likelihood maps [7]. Heinzmann & Zelinsky used the affine perspective to estimate gaze orientation [46]. However, as the weak perspective considers the face as a flat rigid object, such estimation is not always reliable, especially for wide head movements. In particular, the edge of the nose is not coplanar with other facial landmarks such as eyes and mouth. Furthermore, the weak perspective assumption is an approximation that breaks down when the subject is not far enough from the camera. The affine perpective assumption also requires the subject to be sufficiently close to the camera. In any case, feature-based methods have difficulties to accomodate wide angles, and depend on the process for finding the facial landmarks finding process and require a high resolution image of the face to work, ie. at least 300x300 pixels. Partial occlusion of features is also problematic. Furthermore, we do not know if the choice of features is relevant for head pose estimation.

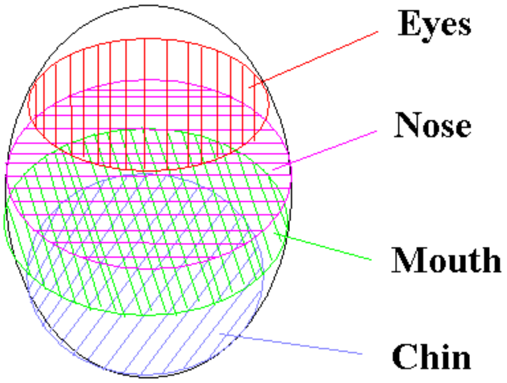

Figure 2.7: Example of facial features variation in the face with regard to head pose

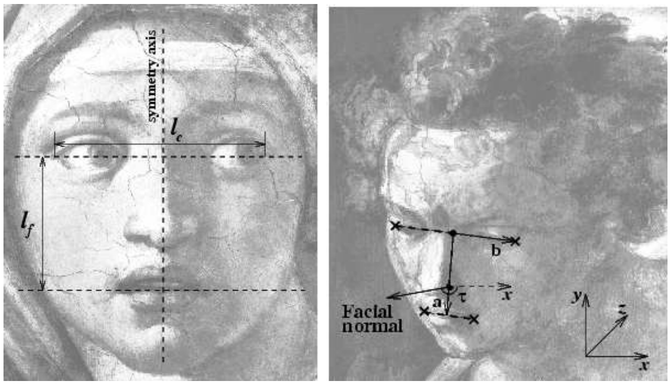

Figure 2.8: Gee & Cipolla's facial model and its application to head orientation estimation. The facial normal is computed from a set of facial features [30]

2.3.2 3D Geometrical approaches

3D Geometrical approaches require a 3D model of the face to be avalaible a priori or computed online. Examples of head models can be seen on figure 2.9. Head pose is estimated by finding correspondences between feature points on the image and on the model. By computing explicitly the reprojection of these points to a plan, we can obtain the 3 rotation angles representing the head orientation. Such methods allow wider head movements than 2D Geometrical approaches. A 3D matching technique was first proposed on objects by Huttenlocher [55], and then by Azarbayejani et al. [2] to estimate the 3D motion of an object. The higher the number of feature points, the higher the precision of the reconstruction. Saddle points and blobs were first used as facial feature points [3]. Such matching techniques can be improved by using algorithms such as EM with least-square fitting [15], optical flow [88] or texture matching with Downhill Simplex [111]. However, illumination variations of the face can greatly influence the results of the algorithms.

The illumination problem can be compensated by taking into account the albedo [9] or by using a geodesic lambertian model with iterative error correction[57]. All these approaches are known to work very well with all types of non-deformable objects, where prior models remain unchanged in the 3D space representation and all transformations are rigid. However, the face is a highly deformable object. When a person turns his head left and right, the neck and the chin modify their appearance on the image. This deformation is even more apparent when the person moves his head up and down. Prior models of the face can not take such deformations into account. Head pose variations are non-rigid. Besides, a single generic head model can not be adapted to all individuals, as the shape differences can be important. With such techniques, changes in identity lead to changes in pose estimation results.

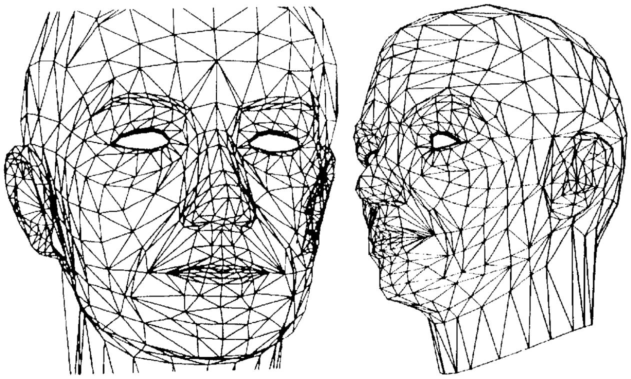

Figure 2.9: Example of 3D wire head models

To improve pose estimation, especially for human faces, some authors use some specific facial features such as eyes, nose and mouth [58, 17, 46, 83] or face edges and curvatures [123]. These approaches present the advantage of being more generic to identity, but have the drawback of requiring a sufficiently high resolution image of the face to allow facial feature detection. Moreover, the features have to be visible on the image for correspondence matching, otherwise the pose estimation is disrupted. Another important point is that the human face can express many emotions. Changes of facial expression in the face can affect the accuracy of the localization of feature points on the image and then influence the reprojection and the accuracy of the estimation.

Rather than using a rigid head model, an online head model can be computed. Large variations in pose and occlusions can be handled by matching a complex grid on the whole face [147]. An example of such a grid can be seen on figure 2.10. Head pose tracking is considered as a problem of local boundary adjustement. However, this technique, as well as other 3D model-based approaches, works only with very high resolution images and is computationally very expensive and still require a 3D model, which is not always available.

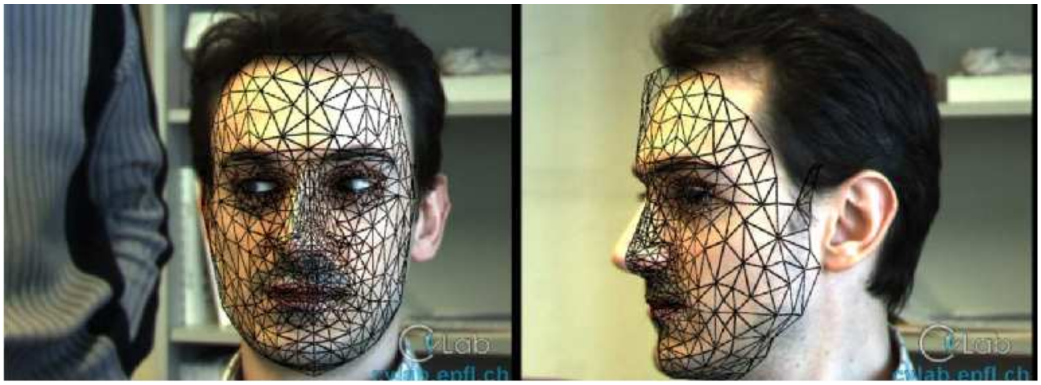

Figure 2.10: Vacchetti et al.'s grid based face tracker. The system requires a very high resolution face image to work [147]

2.3.3 Facial Transformation based approaches

Rather than constructing a model of the face and trying to recover the 3D rotation of the head by using correspondences of facial points, facial transformation based approaches aim at exhibiting an explicit function to compute the head orientation using some facial properties. The main advantages of such approaches are that they are more general and use less detection preprocessing than geometrical approaches. Some authors use hair location [14, 154, 121] with regard to the face to estimate pan and tilt angles. Although these methods work well on a single image and require no calibration, the pose estimation can be disrupted if the subject's hair is not symmetric. Other methods use the similarity between the appearance of the two eyes [18, 22] or between the iris and the eye [108] to estimate the pose. Such techniques work well if both eyes are visible on screen, but fail otherwise. To avoid the problem of eye detection, some authors propose to measure the whole assymetry of the face [50, 95, 25] to compute the head orientation, as in figure 2.11. However, hair is the most variable part of the face and can disrupt the estimation. Furthermore, only the assymetry between the left and the right parts of the face is measured. We have seen that at least 2 pieces of vertical information are required to measue the tilt angle. So such approaches can not deliver a precise estimation of the tilt angle.

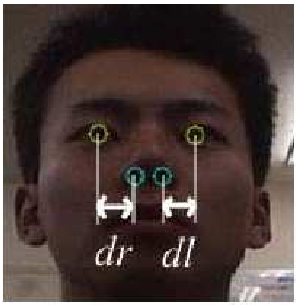

Figure 2.11: Miyauchi et al.'s facial features tracker. The assymmetry of the face is used to compute the head pose [95]

To recover the whole 3D rotation of the head, Yao et al. [164] suggest analysing head movements by considering the transformation of the ellipse delimiting the head of the subject. Although their technique is very simple and efficient, it still requires calibrating during initialization and a good resolution of the image of the face. Kruger et al. [72] map a set of Gabor wavelets on the face to obtain the head orientation. Contrary to previous approches, the mapping is done without facial feature extraction. Gabor wavelets describe both geometrical and textural information on the image. However, their method only works over a limited range of poses. Facial transformation based approaches are very simple to compute, do not require specific model construction and are very fast. The main disadvantage is that, as they only consider one or two facial properties, they can be very unstable and results may vary from a person to another. Facial expression and illumination changes may also be problematic for such methods because facial properties may change while the head orientation remains the same.

2.3.4 Template based approaches

Template based methods are popular approaches which consider head pose estimation as an image classification problem. Unlike previous methods, such global techniques often use the entire image of the face to deduce the head orientation. Once the facial region is detected, it serves as an input and is injected to a nearest-neighbor search with face templates already constructed. The head pose associated with the template which obtains the best match is selected. The denser the training set of templates is, the more accurate the estimate will be. Main advantages of template-based methods are that they can work at low resolution and no model need to be manually constructed, the face only has to be detected.

A popular global approach for image classification is the well-known Principal Component Analysis (PCA). This technique, made popular by Turk & Pentland for face identification [146], was extended to head pose estimation by McKenna & Gong [106, 34, 92, 91, 35, 122] and later refined in other works such as [127, 53, 20]. An example of pose templates across the view-sphere is shown on figure 2.12. A common result of such studies is that the first Principal Component (PC) often captures the illumination direction and the information about left/right of the pan angle and the second PC captures the information front/profile. However, training imagettes are generally cropped and aligned manually and PCA tend to be sensitive to alignment and identity of the subject.

Other approaches use some local features such as the location of eyes in eigenfaces images to estimate the pan angle [51] or gabor wavelets eigenspaces [157, 98, 97]. Other subspaces such as Kernel PCA [77], tensor models, LEA [145], KDA [13] and Local Gabor Binary Patterns [84] have also been used for head pose estimation, as well as multi-resolution template matching. This technique was first used in [6] to recognize human head movements such as no and yes. Nevertheless, these methods only work on a limited range of poses and the number of dimensions to use is still manually determined.

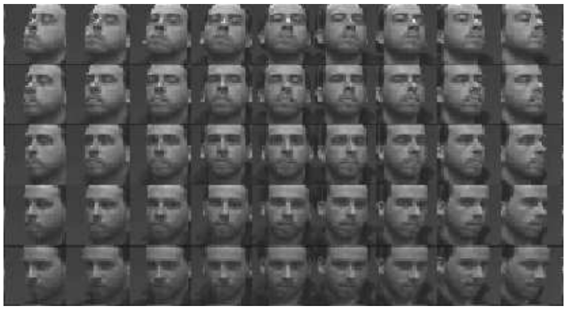

Figure 2.12: Face images of a person from discrete views across the view-sphere [106]

To take into account identity variations, Support Vector Machines (SVM) have been used to estimate head orientation [52, 102, 156]. As with PCA, images must be aligned and SVM are computationaly expensive to train. Niyogi & Freeman [104] use a structured tree search algorithm to separate identity and pose, but their method works in a limited range of poses. Verma et al. [150] use probabilist detectors for frontal and profile poses to obtain a coarse estimate of the head pose. Wu & Toyama [161] use Gabor Wavelets probability to obtain on the head orientation. However, neuronal methods have been found to deliver better results than probabilist methods [10]. Neural networks have the advantage that they can take intra-class variations into account. In their first application to head pose estimation, they were used to detect frontal faces on images [140]. Multi-layer perceptrons with error back propagation were applied later for discrete [116] and continuous [136, 132, 130, 135, 152, 131] head pose estimation. Rather than using the entire image of the face, other techniques use imagettes of facial features and put them as entries in a neural network [149, 112]. However, template based methods only deliver a coarse estimation of head orientation. The number of cells in hidden layers is still chosen arbitrarily and the functioning of neural nets cannot be seen by the developer, and thus

Pose	Local Approaches	Global approaches
Low Resolution	-	+
High Accuracy	+	-
Wide Angles	-	+
No Model Construction	-	+
Global Illumination	+	-
Error Feedback	+	-
Partial Occlusion	-	+
Feature Localization	+	-

Table 2.1: *Comparison between local and global approaches*

it is difficult to inspect exactly what such systems measure.

2.3.5 Coarse-to-Fine approach

We have seen that all the previous approaches can be roughly divided into 2 main groups : local and global approaches. The repartition of head pose estimation approaches can be seen in figure 2.13. Local approaches use information contained in the neighboorhood of pixels, whereas global approaches use the entire image of the face to estimate the head orientation. Local approaches present the advantages of delivering precise values for pan and tilt angles and are robust to illumination. Moreover, most of these methods include the localization of principal facial features as preprocessing. This allows us to understand why the pose estimation fails in certain cases. However, the main drawbacks of local approaches are that they often require a high resolution image for the facial features to be detected, have problems with wide variations of head movements and are not robust to identity. Moreover, inaccurate detection and partial occlusion of facial features disrupt the pose estimation process.

Global approaches better accomodate intra-class variations. Because they work on the entire face region, no specific model needs to be constructed and only face detection preprocessing is required. This means that these methods do not need accurate landmark detection and can be robust to partial occlusion. In addition, global approaches are able to work at lower resolutions and to handle a wider range of head angles. Nevertheless, only a coarse estimate of the head orientation can be obtained and the user does not have any control of what happens if the system fails. In most cases, these techniques are sensitive to illumination. Table 2.1 sums up the advantages and disadvantages of local and global approaches.

The complementary nature of global and local approaches suggests their use in a two stage process. To our knowledge, very little work using both global and local approaches has been done on head pose estimation. We have seen that increasing the face image resolution can increase the estimation accuracy. Computing the pose from a low resolution image to a bigger

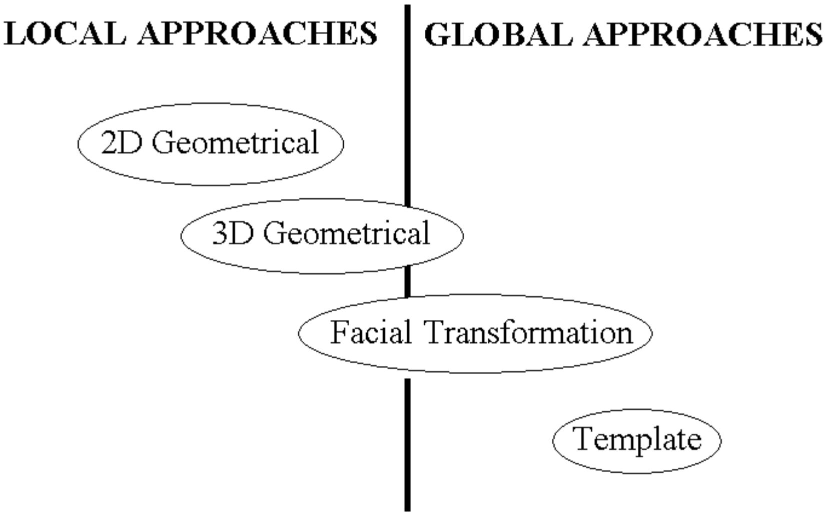

Figure 2.13: Repartition of head pose estimation approaches between local and global

resolution image is a coarse-to-fine process. Wu and Trivedi [160] have recently proposed a two-level head pose estimation system in which a first coarse estimation of the pose is done using Kernel Discriminant Analysis (KDA). The estimation is then refined using Gabor wavelets and Elastic Bunch Graph matching [158] by constructing a graph for each head pose. This methods provides good results, but training and test data are randomly separated. In addition, Gabor wavelets are computationally expensive. Furthermore, graphs are manually constructed for each person and pose. We do not know if the choice of graphs' nodes located at some facial points and of graphs' egdes is relevant for head pose estimation. Training a new person requires to label manually graph nodes and edges on all his face images.

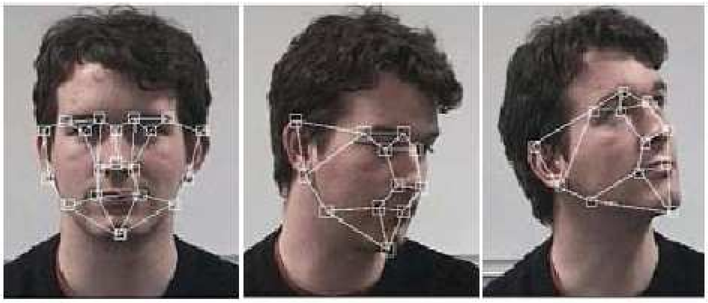

Figure 2.14: Examples of elastic bunch graph matching on a face [159]

We propose a new coarse-to-fine approach to estimate head orientation on unconstrained images without using any heuristics or prior knowledge on the face. This method is easy, fast, robust to partial occlusion and can be adapted to deformable objects other than the human head. Coarse head pose prototypes are put into linear auto-associative memories [148], which are a particular case of one-layer neural network. These prototypes are learned using the Widrow-Hoff rule [1], which is a local correction rule minimizing the error between the reconstructed image and the desired response. Linear auto-associative memories require very few parameters and offer the advantage that no cells in hidden layers have to be defined and class prototypes can be saved and recovered for all kinds of applications. The use of hidden layers in neural networks prevents the system from recovering prototypes. We obtain a coarse estimation of the head orientation by searching the best prototype which match the current image.

We also search for salient facial regions relevant for each head pose. Such salient regions are locally described by Gaussian receptive fields normalized at intrinsic scale. These descriptors have interesting properties and are less expensive than Gabor wavelets. Salient facial features found by Gaussian receptive fields allow the construction of a model graph for each pose.

In ou method, linear auto-associative memories deliver a coarse estimation of the pose. We then search among the coarse estimates for a neighboring graph that obtains the best match. The pose associated with this model graph is selected as the head pose of the person on the image. We describe this approach in the following chapters, but first we need to establish human abilities to estimate head pose.

Chapter 3

Human Abilities for Head Pose Estimation

The goal of this chapter is to determine the accuracy that can be expected from a head orientation estimation system in intelligent environments. Humans are known to estimate visual focus of attention through the head pose, but their abilities remain largely unknown. As a baseline, we have measured human performance for this task using the same sampled database of discrete head poses with which our automatic methods have been tested. The first part of the chapter presents studies related to this topic. We describe the goals of our experiment in the second part. The experimental protocol is detailed in the third section, followed by a discussion of performance evaluation. The result of our experiments show that humans demonstrate a much greater ability to estimate side to side orientation than up and down orientation.

3.1 Related work

This section reviews previous work related to visual perception of images by humans. We are particularly interested in understanding how people examine and interpret images representing persons visually attending to a target.

3.1.1 Human Vision Process

Human gaze is characterized by periods of fixation followed by rapid shifts in direction. This phenomenon, known as saccadic eye movement and is a ballistic movement. Once initiated, the target location cannot be modified and movement occurs between 30 and 120 ms after initiation. Inter-saccadic fixations have a duration of 200 to 600 ms and visual processing of the retina takes place during this period.

Saccades can be conscious or unconscious and are the only movement of the human body whose duration is constant [110]. Occulography allows us to obtain the scan paths realised by gaze on images. Yarbus [165] has study saccadic eye movement fixation and the eye scan pattern followed by gaze. An important result from the study of Yarbus is that the path realised by gaze

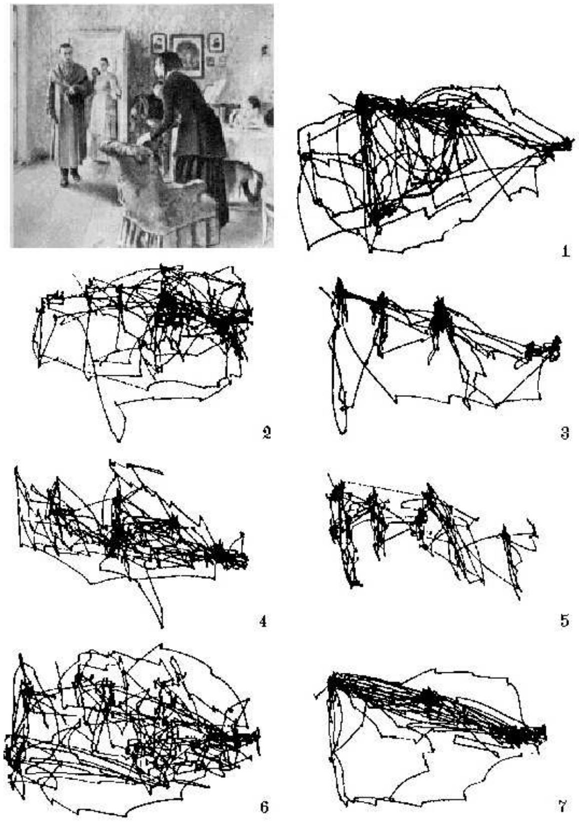

Figure 3.1: The unexpected visitor. The diagrams show records of eye movements of the same subject with different tasks. The subject has 3 minutes to: 1. freely examine the image, 2. estimate the material circumstances of the family, 3. give the ages of people, 4. guess what the family has been doing before the arrival of the unexpected visitor, 5. remember the clothes worn by people, 6. remember the location of people and objects in the scene, 7. estimate how long the unexpected visitor has been away from his family [165].

differs with regard to the task asked from the subject. Figure 3.1 shows different records of eye movements made by Yarbus. The human gaze tries to solve the task by analyzing relevant parts of the scene. We can see that the nature of the task greatly influences the nature of eye movements. For example, saccadic gaze paths located only in certain locations of the image are the result of specific local tasks, such as estimating the age or remembering the clothes of people. Global tasks such as free examination or remembering the whole scene generate a homogeneous gaze path in the image. An interesting result was found when subjects were asked the question: how long has the unexpected visitor been away from his family ? In this case, fixations were directed to the faces of the persons in the image, as if directed by face orientation of the persons depicted in the painting. Their head orientation is directed towards each other. This observation leads us to wonder if humans have abilities to estimate the head pose of people in images.

3.1.2 Human Head Pose Estimation

The psycho-physical basis for human abilities for estimation of head orientation remains unknown. We do not know for example whether humans have a natural ability to estimate the head pose of people in single images, or whether people acquire such an ability through experience. Furthermore, we do not know the accuracy with which a person can deliver values for pan and tilt angles of the head.

To our knowledge, there is no data avalaible to test human competences for head pose estimation. Kersten [65] reports that front and profile poses are particularly well recognized by humans. These poses are used as key poses by the human brain. This observation is true not only for heads, but also for objects in general. Figure 3.2 is an example of phenomenal competition of head poses. Front and profile poses are often unconsciously activated by our brain, for example in social interactions. This is especially true for front poses. In his study, Steinzor [128] reports that two people facing each other are more likely to interact. A more precise experiment to determine at which accuracy head orientation can be estimated was made by Galev and Monk [28]. They asked subjects to look at a sampled grid of points. However, the range of poses used in their experiment was very limited.

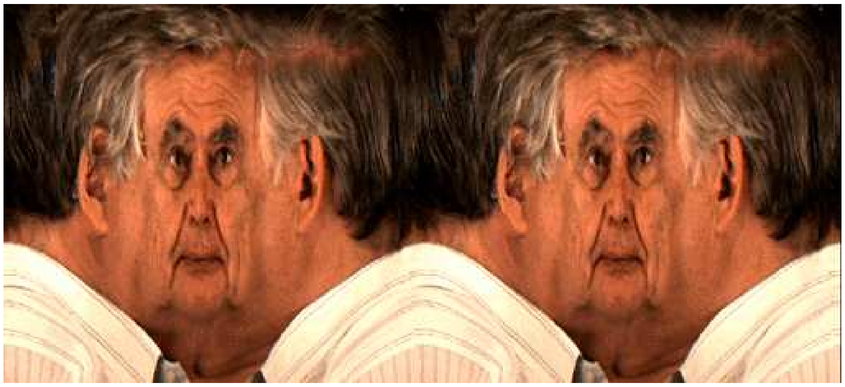

Figure 3.2: Flattened cylindrical projection of a human head [65]. All views are visible in this image, but our brain tends to cut it in patches of front and profile poses.

3.2 Experimental goals

We propose to evaluate the performance of a group of people on head orientation estimation by using a densly sampled database covering a half-sphere of poses. The goal of our experiment has been to assess the performance of people for head pose estimation in pan and tilt angles, for comparison with the results obtained by our computer vision-based approach. We want to know which sufficient accuracy can be expected from a head orientation estimation system. To make the comparison between human and machine performance possible, both experiments must be achieved on the same database. Images of the Pointing 2004 Head Pose Image database [39] are used to evaluate the abilities of humans for head orientation estimation. The Pointing 2004 Head Pose Image database is a densly sampled database covering a half-sphere of poses from -90 to +90 degrees in pan and tilt angles. Further details on the database can be found in Appendix A.

An additional goal of this experiment was to determine which axis is the most significant for people. To do this, we must be able to tell whether pan and tilt angle estimation tasks are natural for humans or not. If one angle turns out to be more natural to estimate than the other, it will signify that this angle is more relevant than the other for human people in their everyday lives.

3.3 Experimental protocol

We measured the performance for a group of 72 human subjects for head pose estimation. In our experiment, we tested 36 men and 36 women, ranging from age 15 to 80. The test is done using a pen and sheets of paper. Subjects were are asked to examine the image, and to circle an answer indicating pose estimation. This answer is selected as the response of the subject to pan or tilt angle estimation. A photo illustrative of the conditions of the experiment is shown on Figure 3.3.

The head orientation task consists in two parts: one for pan angle estimation, and the other for tilt angle estimation. Images from the Pointing 2004 Head Pose Image Database were presented in random order to the subject for 7 seconds, with a different order for each subject. If the images were shown in the same linear order, we would have measured the performance of subjects on the same sequence of images, and our experiment would have been biased. Presenting the images in a random different order allows us to measure the performance of the subject on the head pose estimation on a set of independent images.

The data set used in this experiment is a subset of the Pointing 2004 Head Pose Image Database. A sample of this subset is shown on Figure 3.4. Each angle varies from -90 to +90 degrees, with a step of 15 degrees for pan and 15 and 30 degrees for tilt. The two parts of the experiment are done in random order to avoid bias in our experiment. Pan angle ranges over the values (-90,-75,-60,-45,-30,-15,0,+15,+30,+45,+60,+75,+90), where negative values correspond to right poses and positive values correspond to left poses. During the pan angle estimation test, symbols "-" and "+" are present on each side of the image to prevent the subject from mistaking left and right poses. Tilt angle can take the values (-90,-60,-30,-15,0,+15,+30,+60,+90). Negative values correspond to bottom poses and positive values correspond to top poses. Both angles vary during pan and tilt estimation task. The data set consists in 65 images for pan axis and 45 images for tilt axis, which allows the participants to have 5 images for each pose.

We want an immediate response from the subject to the presentation of images. The duration

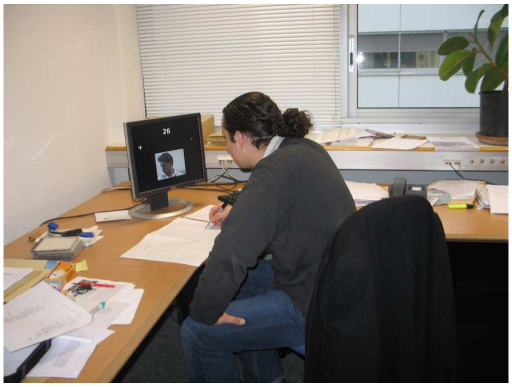

Figure 3.3: Experimental conditions

of 7 seconds is convenient because it is both long enough to allow the subject to seek the current image and to select his response and short enough to have the subject give an immediate response.

Another important goal of this experiment is to obtain the best human performance for head pose estimation, in order to compare it with the results obtained by our system. However, we do not know whether people have or develop a natural ability for this task, or whether people must be trained for head pose estimation using example annotated images. To avoid this bias, the subjects were divided into 2 groups of 36 persons. People in the first group may inspect labelled training images as long as they wish before beginning each part of the experiment. Examples of such images are shown on figure 3.5. People in the second group were not provided any opportunity to see training images before the experiment. The first and second groups are respectively referred to as "Calibrated" and "Non-Calibrated" subjects. Thus, four groups are constructed: 18 "Calibrated" women, 18 "Calibrated" men, 18 "Non-Calibrated" women and 18 "Non-Calibrated" men. Randomly creating these two groups allows us to determine if training significantly increases human performance on head pose estimation on each axis. If this is not the case for a certain axis, it will mean that people have a natural ability to evaluate head pose on this axis, and that this angle is relevant for them.

Some vision tasks are known to become more difficult with growing age. Another goal of our experiment was to determine if this is the case for head pose estimation from single images. We investigated whether the age of the subject influences his abilities for head pose estimation. People are asked to write their age down before the beginning of the task. To perform this type of estimation, the subject must know elementary notions of spatial geometry. The youngest person who took part to the experiment was 15 years old.

At the end of the experiment, we presented another image taken from the works of Kersten [65] representing a flattened image of a cylindrical project of a human head in pan axis. This image is shown on figure 3.6. All views of the head are available twice on this single image. The subject is asked to indicate which pan angles he sees from this image. The goal of this question

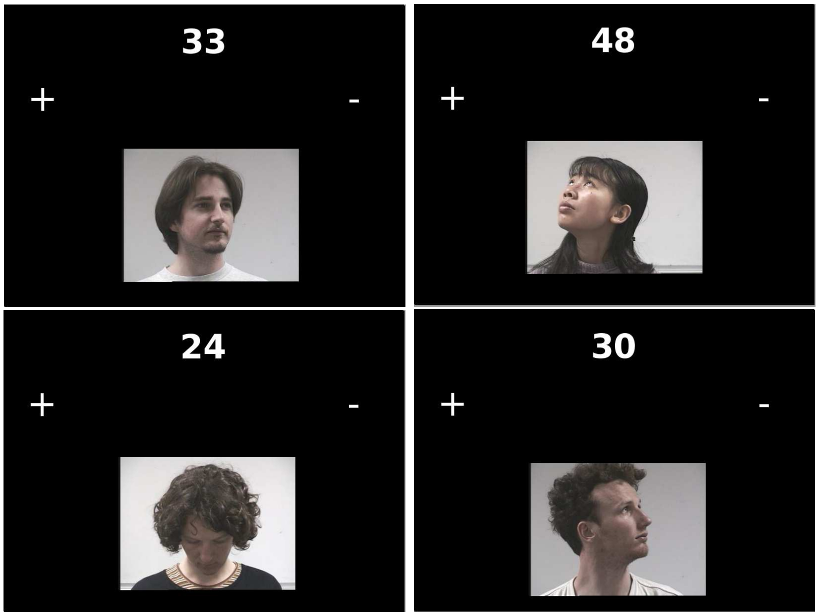

Figure 3.4: Examples of test images presented to the subject during the experiment

was to confirm the use of key poses by the human brain. As we want to avoid responses from people familiar with the field, we asked the subject if he had already seen this kind of image after the experiment. As a conclusion, people indicated on their test paper if they think that they have learned to estimate the head pose on each axis during the experiment.

3.4 Results and discussion

In this section, we describe human performance on head pose estimation. Specific evaluation measures were designed for this task. These results give an idea of the accuracy required for this task in a Man-Machine Interaction context. We also compare performance of groups of populations using statistical tests. In particular, we want to determine if examining training images before the experiment provides better results and if there is an angle which is more natural to estimate for humans.

3.4.1 Evaluation Measures

To determine human performance, we must define evaluation criterions. The main evaluation metric is the mean absolute error for pan and tilt angles k. This error defined by averaging absolute differences between theoretical value $p(k)$ and the value $p^*(k)$ given by the subject (3.1) for the image k. N is the total number of images for each axis. As the sampling is not uniform for tilt angle, the difference is obtained by considering median values for each range

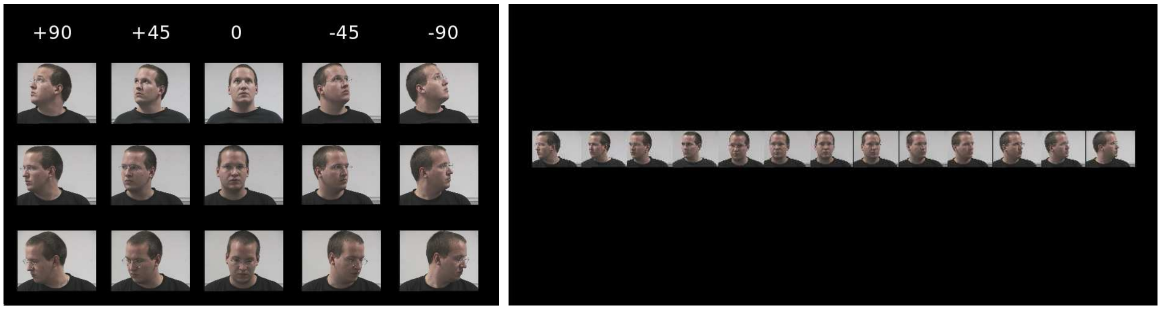

Figure 3.5: Examples of training images shown to "Calibrated" subjects for pan angle

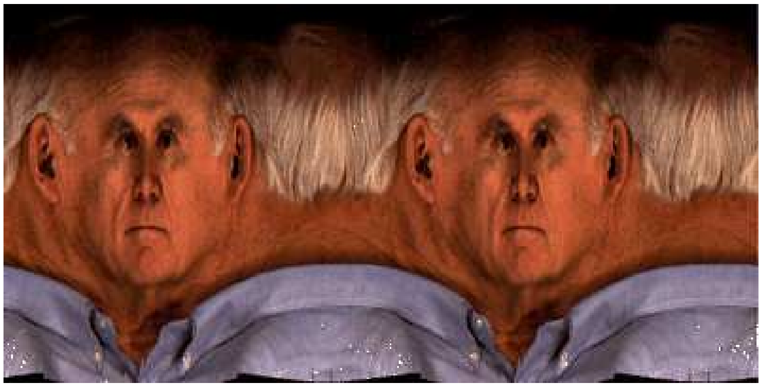

Figure 3.6: The flattened cylindrical projection of a human head [65] presented to the subject at the end of our experiment

of poses. The computation of the absolute difference for tilt angle is summed-up in Table 3.1. We also compute the maximum absolute error on each axis for each subject (3.2). Another interesting measure is the correct classification rate. This is defined as the ratio of the number of correct answers to the total number of answers (3.3).

The subset taken from the Pointing 2004 Head Pose Image Database contains the same amount of data for each pose. This allows the computation of another interesting evaluation measure, the mean absolute error per pose (3.4). This is defined by averaging absolute differences between expected value and the value given by the subject for each pose P. This metric shows the repartition of errors among head poses and can highlight specific poses particularly well recognized by humans.

$$\begin{aligned} MeanAbsoluteError &= \frac{1}{N} \cdot \sum_{k=1}^{N} \|p(k) - p^{*}(k)\| && (3.1) \\ MaxAbsoluteError &= max_k \|p(k) - p^{*}(k)\| && (3.2) \end{aligned}$$

$$CorrectClassificationRate = \frac{Card\{ImagesCorrectlyClassified\}}{Card\{Images\}} \tag{3.3}$$

$$MeanAbsoluteError(P) = \frac{1}{Card\{Images \in P\}} \cdot \sum_{k \in P} \|p(k) - p^*(k)\| \tag{3.4}$$

Tilt angle	-90^o	-60^o	-30^o	-15^o	0^o	$+15^o$	$+30^o$	$+60^o$	$+90^o$
-90^o	0	30	56.25	75	90	115	123.75	150	180
-60^o	30	0	26.25	45	60	75	93.75	120	150
-30^o	60	30	0	15	30	45	63.75	90	120
-15^o	75	45	18.75	0	15	30	48.75	75	105
0^o	90	60	33.75	15	0	15	33.75	60	90
$+15^o$	105	75	48.75	30	15	0	18.75	45	75
$+30^o$	120	90	63.75	45	30	15	0	30	60
$+60^o$	150	120	93.75	75	60	45	26.25	0	30
$+90^o$	180	150	123.75	115	90	75	56.25	30	0

Table 3.1: *Absolute error computation for tilt angle. The top row is the value given by the subject. The left column is the expected tilt angle*

3.4.2 Human Performance

We computed the evaluation measures for all subjects and average it for each category of people. Results for pan and tilt angles are presented in Tables 3.2 and 3.3. Global human mean error is 11.9 degrees for pan and 11 degrees for tilt. The average classification rate is 53.6% in tilt and 41.6% in pan, which is below the 50% performance rate. Maximum error varies from 30 to 60 degrees on both axis, which is superior to the gap of 15 degrees between two poses. This proves that the database is sufficiently sampled for subjects.

We want to know if there are significant differences in performance for groups of people. We constructed hypothesis tests with a confidence threshold of 95% by applying a test of Student-Fisher. Details of this statistical operation are shown in Appendix B. Results of comparison of human perfomances between populations on pan and tilt axis can be seen in Table 3.4. Calibrated people do not perform significantly better in estimating pan angle. However, the difference is significant for estimating tilt angle. This result shows that head pose estimation appears to be natural in pan, but not for tilt. This may be due to the fact that people twist their heads left and right more often than up and down during social interactions. In situations when people talk to each other, pan angle provides information about visual focus of attention [135, 64, 128]. Head pose changes in tilt become meaningless. This fact is even more relevant when people are sitting, because theirs heads are roughly at the same height. Humans are more used to considering head pose changes in pan. Furthermore, the best human performance is obtained by "Calibrated" subjects.

Men obtain better results in pan angle, but similar results in tilt angle as women. We do not know if "Calibrated" and "Non-Calibrated" subjects really learn to estimate the head pose during the experiment. As shown in Table A, only two out of three people feel they have improved their estimation during the task. Furthermore, those people do not have better performance than others in pan and tilt angles.

Pan Evaluation Measures	Mean Absolute Error	Avg. Max Error	Correct Classification
All Subjects	11.85^o	44.79^o	41.58 %
Calibrated Subjects	11.79^o	42.5^o	40.73 %
Non-Calibrated Subjects	11.91^o	47.08^o	42.44 %
Men	11.09^o	42.5^o	44.15 %
Women	12.61^o	47.08^o	39.02 %
Subjects who learn	11.71^o	45.6^o	42.25 %
Subjects who do not learn	12.17^o	42.95^o	40.07 %
Best Performance	7.62^o	30^o	52,31 %
Worst Performance	18.46^o	60^o	26.15 %

Table 3.2: *Pan evaluation measures results*

Tilt Evaluation Measures	Mean Absolute Error	Avg. Max Error	Correct Classification
All Subjects	11.04^o	45.1^o	53.55 %
Calibrated Subjects	**9.45**^{o}	39.58^o	59.14 %
Non-Calibrated Subjects	**12.63**^{o}	50.63^o	47.96 %
Men	10.53^o	43.96^o	55.43 %
Women	11.54^o	46.25^o	51.67 %
Subjects who learn	11.29^o	47^o	52.84 %
Subjects who do not learn	10.62^o	41.94^o	54.73 %
Best Performance	4.83^o	30^o	75,56 %
Worst Performance	21.08^o	60^o	56.25 %

Table 3.3: *Tilt evaluation measures results*

The average error per pose in pan is shown on figure 3.7. We found an interesting result for this axis. Humans perform well at recognizing front and profile views, but not for intermediate views. The average error per pose in pan can be roughly modelled by a Gaussian centered at

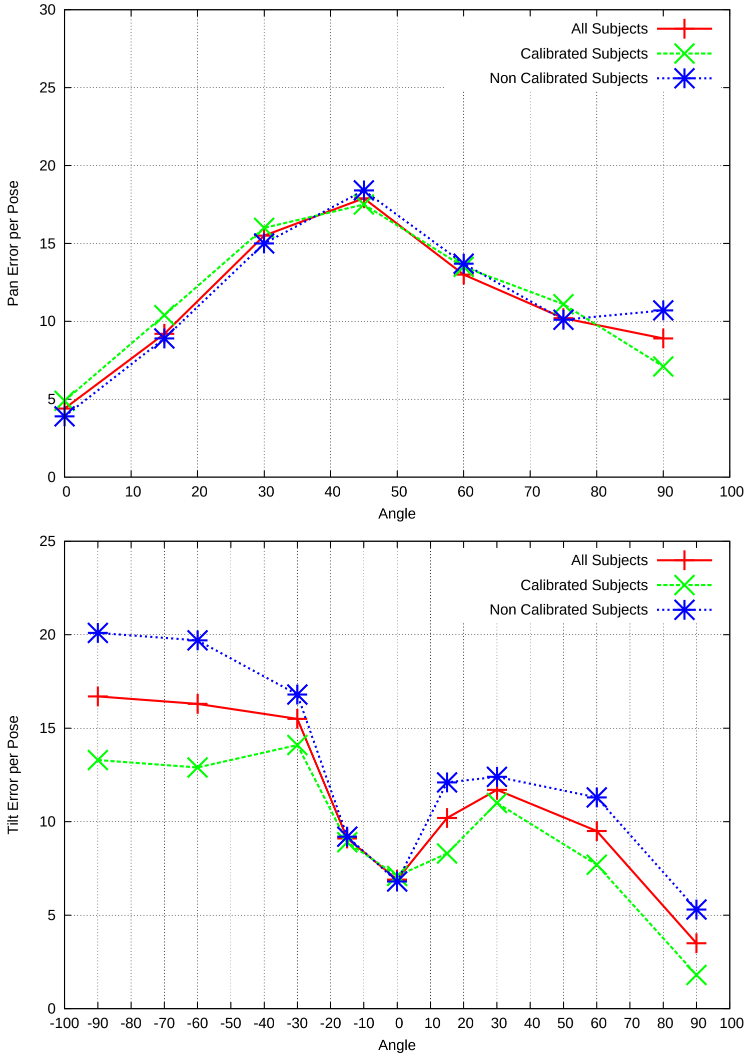

Figure 3.7: Pan and Tilt Error per pose of different populations

45 degrees. Minimum error in pan is found at 0 degrees, which corresponds to the front pose. Furthermore, during our experiment, we observe that most people did not use intermediate pan poses such as 30,45 and 60 degrees. This fact is confirmed by the presentation of Kersten's head cylindrical image (Figure 3.6) at the end of our experiment. All subjects were asked to indicate which views they were able to see on this image. Results are presented in Table 3.5. As we can see, everybody saw front poses, most people saw profile poses, but less than one out of five subjects saw intermediate poses. These results show that the human brain uses front and profile views as key poses, as suggested in [65].

Figure 3.7 also shows the average error per pose on tilt axis. Humans perform better for top angles than for bottom angles. The minimum error can be found at +90 degrees, whereas the

Is there a significant difference...	in Pan axis ?	in Tilt axis ?
Calibrated Subjects > Non-Calibrated Subjects	NO	YES
Men > Women	YES	NO
Subjects who learn > Subjects who do not learn	NO	NO

Table 3.4: *Performance comparison between groups of people*

maximal error is found at -90 degrees. This may be due to the fact that, when a face is nodding downward, hair dominates a large surface of the apparent face, providing more information about side to side angle, and less for tilt angle.

The last goal of this study is to determine if the age of the participant has an influence on his performance at head pose estimation. Figure 3.8 shows the repartition of pan and tilt average error for each subject with regard to their age. We want to know if the variables age and average error in pan and tilt axis are correlated. To perform this, we compute the unbiased correlation coefficient for each angle. Details of this operation can be found in Appendix B. We found a coefficient of 0.25 in pan and 0.11 in tilt. The age of the subject does not seem to influence their results on head pose estimation task.

Poses	Detection Rate
Front	100 %
Profile	73 %
Intermediate	19 %

Table 3.5: *Detection rate of different poses on Kersten's cylindrical head image. We only take people who have not seen such image before the experiment into account.*

Subjects who learn to estimate pan angle	69 %
Subjects who learn to estimate tilt angle	63 %

Table 3.6: *Percentage of people who think they learn to estimate pan and tilt angle during the experiment*

We measured the performance of 72 human subjects on head pose estimation from single images of a densly sampled database. The subjects were divided into 2 groups to see whether this task was natural for them or not. With adapted eveluation measures, we explicited the accuracy of estimations on each pan and tilt angle. Our experiment tends to show that tilt angle

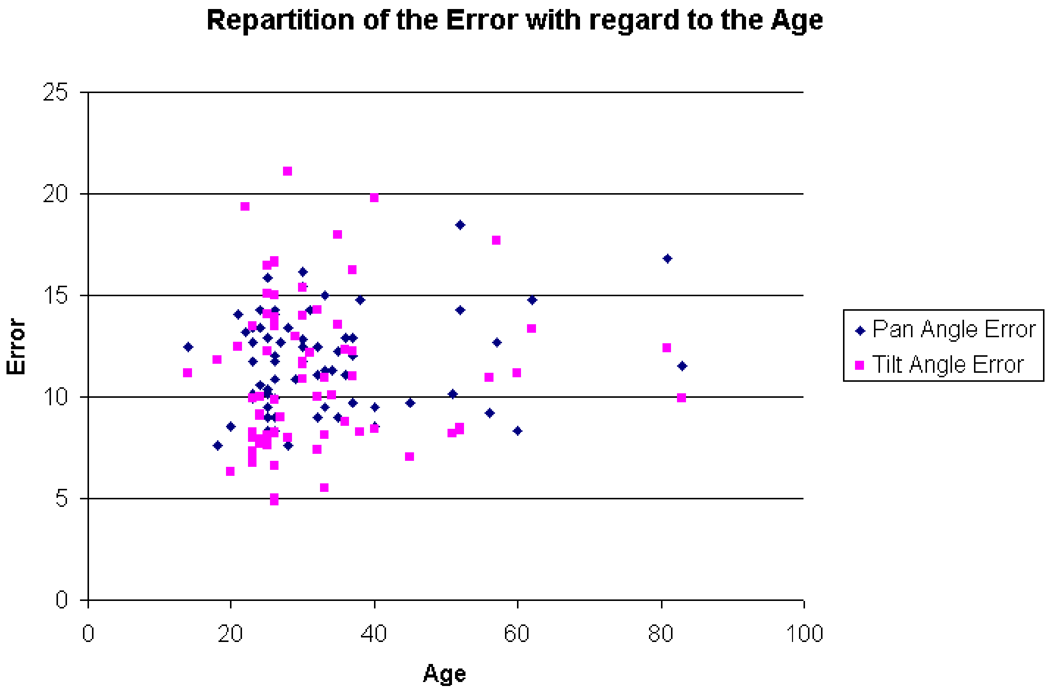

Figure 3.8: Repartition of the error in pan and tilt angle with regard to the age of subjects

estimation is not natural for humans whereas pan angle estimation is. Front and profile views are particurly well recognized, but abilities degrade for intermediate views. The age of the subject does not seem to influence human abilities for head pose estimation. We now have a baseline for comparison with results obtained by computer vision-based approaches. Our system will be tested on the same database.

Chapter 4

A Robust Face Tracker

This chapter describes the robust video rate face tracker and detector used in this thesis. We do not want to manually crop face regions in the images, as it requires human intervention. Moreover, cropping results may vary from one person to another. To avoid human intervention and to simulate head pose tracking in real conditions, face images of the database are detected using this system. This algorithm provides an initial detection and normalization of a face region in video sequences and single images. Our tracker uses pixel level detection of skin colored regions using a Bayesian estimation of the probability that a pixel corresponds to skin based on its chrominance. A prediction-verification step is performed using a zeroth order Kalman filter. The face tracker is used to normalize facial images into small imagettes.

4.1 Pixel Level Detection

In our experiments we use a robust video rate face tracker to focus processing on face regions, although any reliable face detection process, such as Ada-Boost [151] could be used for this step. To detect a face, we first detect skin regions within the image using a probabilistic detection of skin chrominance. The human face is a highly deformable surface and can be illuminated under several conditions. If we assume a nearly lambertian reflection fuunction for skin, the intensity component is defined by the changes with surface orientation, whereas the body reflection component models the characteristic color of the object. The exact chrominance of the skin of an individual is determined by the product of the spectrum of skin pigments and spectrum of illumination. While face regions may have strong variations in intensity, their chrominance will remain constant. As a result, the chrominance of an object therefore provides an invariant signature for its identity, whereas intensity represents information about the surface orientation and changes.

We compute the chrominance by normalizing the red and green components of the (R, G, B) color vector by the intensity $R + G + B$. Normalizing intensity removes the variations due to angle between the local surface normal and the illumination source. We use an intensity

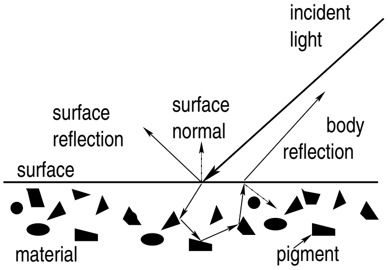

Figure 4.1: Dichromatic reflection model. Pigments near the object surface modify the body reflection [67]

normalized chrominance space (r, g). The chrominance values are computed as follows, as proposed by Schiele [114]:

$$r = \frac{R}{R+G+B}$$
$$g = \frac{G}{R+G+B}$$

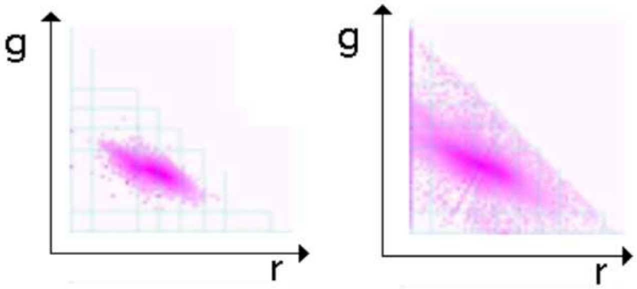

Figure 4.2: Examples of density histograms. The left histogram represents a skin probability density. The right histogram represents a total image density

The conditional probability densities for the (r, g) vector to belong to skin regions and for all the image can easily be estimated using histograms. Bayes' rule shows that the ratio of these histograms provides a lookup table that maps the normalized chrominance to the conditional probability of skin $p((x, y) \in Skin|r, g)$ that a pixel (x, y) of chrominance (r, g) belongs to a skin region. This lookup table gives us a direct relation between intensity normalized color and probability:

$$p((x, y) \in Skin|r, g) = \frac{p(r, g|(x, y) \in Skin)p((x, y) \in Skin)}{p(r, g)}$$

This probability will be denoted $P_{skin}(x, y)$. The skin probability map is obtained by computing the skin probability for each pixel in a determined region. An example of probability map is shown on figure 4.3. By defining the following terms:

- N_{total} : Number of pixels on the image
- N_{skin} : Number of pixels on the image part of a skin region
- $Histogram_{total}(r, g)$: Cell (r, g) of the histogram of the whole image
- $Histogram_{skin}(r, g)$: Cell (r, g) of the histogram of skin regions

We obtain:

$$\begin{aligned} p((x,y) \in Skin) &= \frac{N_{skin}}{N_{total}} \\ p(r,g) &= \frac{1}{N_{total}} Histogram_{total}(r,g) \\ p(r,g|(x,y) \in Skin) &= \frac{1}{N_{skin}} Histogram_{skin}(r,g) \end{aligned}$$

The skin probablity $P_{skin}(x, y)$ can be expressed as the ratio of the skin histogram and the total histogram:

$$\begin{aligned} P_{skin}(x,y) &= \frac{N_{skin}}{N_{total}} \cdot \frac{Histogram_{skin}(r,g)}{N_{skin}} \cdot \frac{N_{total}}{Histogram_{total}(r,g)} \\ &= \frac{Histogram_{skin}(r,g)}{Histogram_{total}(r,g)} \end{aligned}$$

This ratio allows us to have a direct relation and a better repartition of the skin probability with regard to the background. This relation is theoretically only valid for the image in which the histograms are calculated. However, this approximation still works for later images when illumination conditions remain stable.

4.2 Tracking using Skin Chrominance

To be able to track the face region, it must be isolated in the image. Face position and surface extent are estimated using moments and tracked using a zeroth order Kalman Filter [61], also called prediction-verification process. The tracking process predicts a region of interest (ROI) that permits processing to be focused on the face region. It also reduces computational cost and improves resistance to distraction by background clutter. In each image, the skin probability

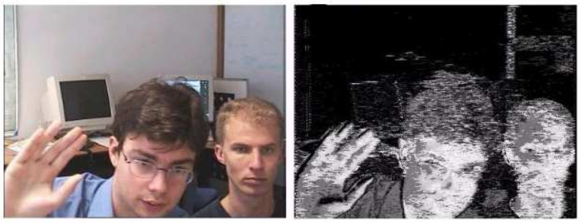

Figure 4.3: Skin probability map of a image

image is calculated within the predicted ROI by the table lookup as described above. This probability map has a centre of gravity $\vec{\mu} = (x_P, y_P)$ and a 2x2 covariance matrix C. Pixels (x, y) within the ROI are then multiplied by the Gaussian $G(x, y, \vec{\mu}, C)$ predicted by tracking. Both the tracking process and face normalization are based on moments. The first moment $\vec{\mu}$, or centre of gravity, provides a robust estimate of face position, while the second moment provides a measure of the width, height and slant of the face. This operation serves to determine the estimated face $Face_{Estimated}$, represented by its moments $(x_E, y_E, sx_E, sy_E, sxy_E)$. The predicted face $Face_{Predicted}$ is determined by (x_P, y_P, sx_P, sy_P). First and second moments of the estimated face are computed with the following formulas:

$$\begin{aligned}
x_E &= \frac{1}{S}\sum P_{skin}(x,y)\cdot x \cdot G(x,y,\vec{\mu},C),\\
y_E &= \frac{1}{S}\sum P_{skin}(x,y)\cdot y \cdot G(x,y,\vec{\mu},C),\\
sx_E &= \frac{1}{S}\sum P_{skin}(x,y)(x-x_P)^2 G(x,y,\vec{\mu},C),\\
sy_E &= \frac{1}{S}\sum P_{skin}(x,y)(y-y_P)^2 G(x,y,\vec{\mu},C),\\
sxy_E &= \frac{1}{S}\sum P_{skin}(x,y)(y-y_P)(x-x_P) G(x,y,\vec{\mu},C)
\end{aligned}$$

where $S = \sum P_{skin}(x,y)\cdot G(x,y,\vec{\mu},C)$.

We estimate the current position and the size of the face within this ROI. The difference between the estimated face at the current frame t and the estimated face at the previous frame $t - \delta t$ represents the variation of the face and serves to predict the ROI in the next frame $t + \delta t$. The centre of the ROI is equal to the centre of the predicted face. The dimensions of the ROI are noted (sx_R, sy_R). For each frame, we have:

- $Face_{Estimated}(t)$: Estimated face at the current frame

- $Face_{Estimated}(t - \delta t)$: Estimated face at the previous frame
- $Face_{Predicted}(t + \delta t)$: Predicted face at the next frame

We define a minimal skin probability P_{min} to eliminate spurious regions. A pixel (x, y) whose skin probability is inferior to this value is set to 0. Another advantage of the minimal probability is that it gives a maximal size for the ROI. By denoting $P'_{skin}(x, y)$ the skin probability of the pixel (x, y) multiplied by the Gaussian based on the predicted face, we have:

$$P'_{skin}(x, y) = P_{skin}(x, y)G(x, y, \vec{\mu}, C)$$

All pixels whose skin probability are inferior to P_{min} are not considered. Such pixels satisfy the condition:

$$\begin{aligned}
P'_{skin}(x, y) &< P_{min} \\
P_{skin}(x, y)G(x, y, \vec{\mu}, C) &< P_{min} \\
P_{skin}(x, y)e^{-(\frac{(x-x_P)^2}{sx_P^2} + \frac{(y-y_P)^2}{sy_P^2})} &< P_{min} \\
e^{-(\frac{(x-x_P)^2}{sx_P^2} + \frac{(y-y_P)^2}{sy_P^2})} &< P_{min} \\
\frac{(x-x_P)^2}{sx_P^2} + \frac{(y-y_P)^2}{sy_P^2} &< -ln(P_{min})
\end{aligned}$$

As $P_{skin}(x, y) \leq 1$. By projecting on the horizontal dimension, the condition becomes:

$$\frac{\|x - x_P\|}{sx_P} < \sqrt{-ln(P_{min})}$$

We want to determine the coefficient c_R which links the dimension of the predicted face sx_P to the dimension of the ROI sx_R:

$$sx_R = c_R \cdot sx_P$$

By expressing the distance $\|x - x_P\|$ as the dimension sx_R, we obtain:

$$c_R = \sqrt{-ln(P_{min})}$$

Idem for y_R. We experimentally chose $P_{min} = 3\%$ in our experiments. We also define an acceleration coefficient c_A to update the dimensions of the predicted face. This coefficient is set to 0.5. The complete prediction-verification step can be described by the formulas:

$$\begin{aligned}
x_P(t+\delta t) &= x_E(t) + (x_E(t) - x_E(t-\delta t)) \\
y_P(t+\delta t) &= y_E(t) + (y_E(t) - y_E(t-\delta t)) \\
sx_P(t+\delta t) &= sx_E(t) + c_A \cdot \|x_E(t) - x_E(t-\delta t)\| \\
sy_P(t+\delta t) &= sy_E(t) + c_A \cdot \|y_E(t) - y_E(t-\delta t)\|
\end{aligned}$$

$$\begin{aligned}
x_R(t+\delta t) &= x_P \\
y_R(t+\delta t) &= y_P \\
sx_R &= c_R \cdot sx_P \\
sy_R &= c_R \cdot sy_P
\end{aligned}$$

The zeroth order Kalman filter process is illustrated on figure 4.4. This step, inspired by robust statistical techniques, improves robustness to background clutter [116]. An example of skin probability map combined with Kalman filtering is presented on figure 4.5. At initialization, the predicted face is either equal to the manual selection on the user onscreen or equal to the whole image. To detect the face on single images, we iterate the Kalman filter until stabilization of the moments is reached. A number of 10 iterations is usually sufficient. Examples of face tracking are presented on figure 4.6.

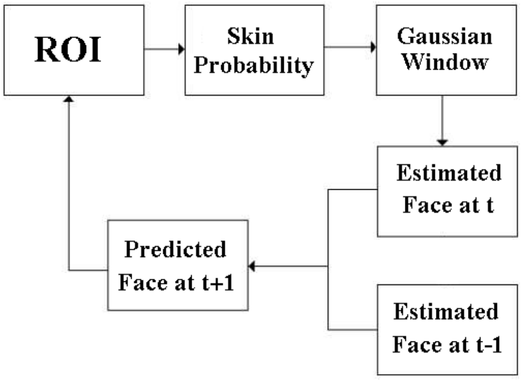

Figure 4.4: Prediction-Verification process. An arrow represent the action "serves to compute".

The discrimination between face and non-face regions is explicited in the next section. A first discrimination of face images is made by considering the ratio between the height and the width of the estimated face region. If this ratio is too high, the region is too thin and cannot correspond to a face. The tracker is then restarted on the whole image.

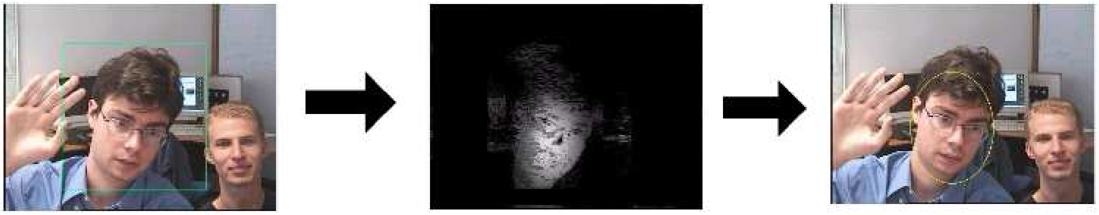

Figure 4.5: From left to right: ROI of a face in the image, Computation of the probability map multiplied by the Gaussian Window, Ellipse delimiting the face region in the image.

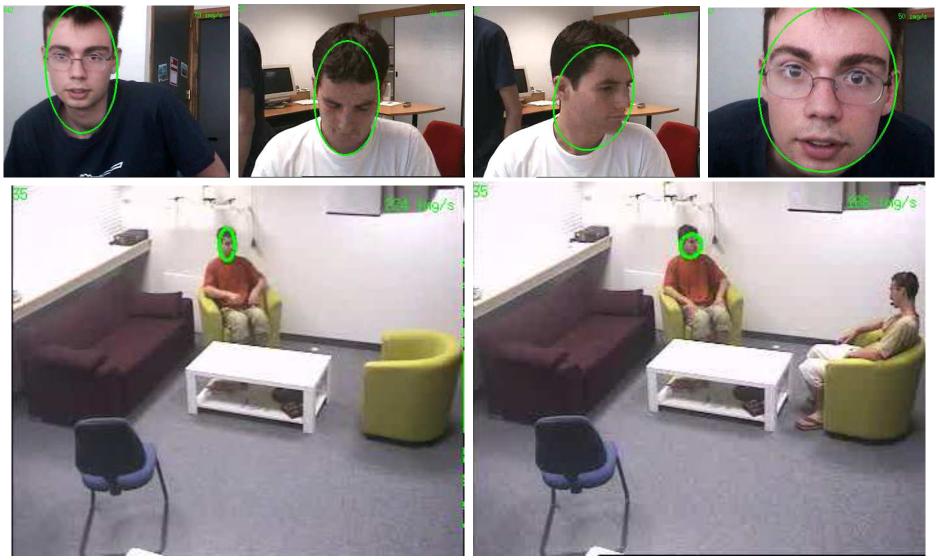

Figure 4.6: Example of face tracking. First and second moments provides an ellipse which delimits the face on the image

4.3 Performance of the Face Tracker

To initialize our face tracker, we employ either the manual selection of the user on the frame, or a generic ratio histogram. The choice of the number of histogram cells used to form the lookup table for skin detection is an important parameter. Histograms with too few cells will not properly discriminate skin from similar colored surfaces such as wood. On the other hand, using too many cells renders the process overly sensitive to minor variations in illumination spectrum as well as skin blemishes. We have empirically observed that (r, g) histograms on the order of ranges 32x32 cells provides a good compromise for face detection. A more thorough analysis is provided by Storing in [139].

The face tracker has been carefully optimized to run at real-time, and can process 384x288

Sequence	Number of images	Eye Detection rate
A	500	99,9 %
B	700	99,8 %
C	580	94,2 %
D	300	93,1 %

Table 4.1: *Eye detection rate. The different video sequences contain the following events: A. Slow Head translation, B. Fast Head translation, C. Head zoom and inclination in the plane, D. Head pitch and yaw*

Pose	Front	Half-profile	Profile
X Center	0,31 %	1,13 %	3,23 %
Y Center	0,64 %	1,05 %	1,58 %
Width	0,55 %	1,08 %	1,38 %
Height	0,64 %	1,14 %	1,38 %

Table 4.2: *Standard deviations of position and dimensions of the detected face ellipse during 20 seconds in different configurations of the face*

pixel images at video-rate on a 800 MHz Pentium processor. Eye detection rate on representative video sequences can be seen in table 4.1. In this case, an error occurs when the computed ellipse does not contain an eye visible in the image.

An important property for a face tracker is jitter. Jitter measures the stability of the tracker. It is computed as the square of the difference in position and size of the detected pixels of the face when the subject is not moving. We have calculated the variance of the moments of the position and size of the detected face region on sequences of 20 seconds taken when the subject's head has a certain pose and is not moving. Results are shown in Table 4.2. We observe that most errors occur when the subject is in profile. In this case, the detection of the neck can modify the detected region.

4.4 Face image normalization

The face tracker delivers the first and second moments of the face region. These values are used to determine an ellipse delimiting the face on the image. From this ellipse, we create a gray scale intensity imagette of dimensions (t_x, t_y) of the face normalized in position, size and slant angle. The intensity, computed as the sum of the color components $R + G + B$, can provide stable salient features based on facial structures and robustness to chrominance changes [119]. An example is shown on figure 4.7. The normalized face imagette is created as follow: for each pixel (x', y') of the imagette, we search its corresponding pixel (x, y) on the original image and take its intensity. The face ellipse is determined by its centre (x_e, y_e), its radius (w, h) and its orientation θ, which represents the slant angle of the face on the image. The transformation of the imagette is a combination of a scaling function S and a rotation matrix R_θ, expressed by:

$$\begin{aligned} S &= \begin{pmatrix} \frac{t_x}{w} & 0 \\ 0 & \frac{t_y}{h} \end{pmatrix} \\ R_\theta &= \begin{pmatrix} cos(\theta) & sin(\theta) \\ -sin(\theta) & cos(\theta) \end{pmatrix} \end{aligned}$$

The centre of the face region corresponds to the centre $(\frac{t_x}{2}, \frac{t_y}{2})$ of the normalized imagette. Thus, the relation between a pixel (x, y) from the original image to its corresponding pixel (x', y') on the normalized imagette is given by:

$$\begin{pmatrix} x' - \frac{t_x}{2} \\ y' - \frac{t_y}{2} \end{pmatrix} = R_\theta \cdot S \cdot \begin{pmatrix} x - x_e \\ y - y_e \end{pmatrix} \tag{4.1}$$

We deduce the inverse relation:

$$\begin{aligned} \begin{pmatrix} x - x_e \\ y - y_e \end{pmatrix} &= S^{-1} \cdot \; R_{-\theta} \cdot \begin{pmatrix} x' - \frac{t_x}{2} \\ y' - \frac{t_y}{2} \end{pmatrix} \\ \begin{pmatrix} x - x_e \\ y - y_e \end{pmatrix} &= \begin{pmatrix} \frac{w}{t_x} & 0 \\ 0 & \frac{h}{t_y} \end{pmatrix} \begin{pmatrix} cos(\theta) & -sin(\theta) \\ sin(\theta) & cos(\theta) \end{pmatrix} \begin{pmatrix} x' - \frac{t_x}{2} \\ y' - \frac{t_y}{2} \end{pmatrix} \end{aligned} \tag{4.2}$$

Which gives us:

$$\begin{aligned} x &= \frac{w}{t_x}(cos(\theta)(x' - \frac{t_x}{2}) - sin(\theta)(y' - \frac{t_y}{2})) + x_e \\ y &= \frac{h}{t_y}(sin(\theta)(x' - \frac{t_x}{2}) + cos(\theta)(y' - \frac{t_y}{2})) + y_e \end{aligned} \tag{4.3}$$

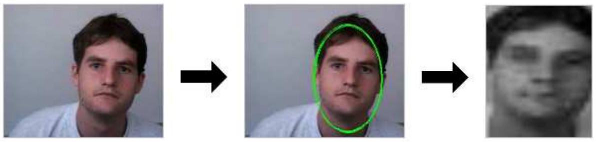

Figure 4.7: Face detection and normalization process

This normalization step offers several advantages. Calculating all pixels this way allows us to restrict processing to a set of positions and scales, thus reducing computation time. This truly provides a fixed number of operations for each face, regardless of its original size onscreen [37]. Furthermore, there is no sampling density problem, because every pixel of the imagette has its match on the source image. Another advantage is that all faces become straight after the normalization step. More precisely, for all faces in a given head pose, the same facials features are expected to be roughly located at the same location on the imagette, as illustrated in figure 4.8. We will use a size of 23x30 pixels for the normalized imagette. All further operations take place within this imagette. This step will be useful for head pose estimation process.

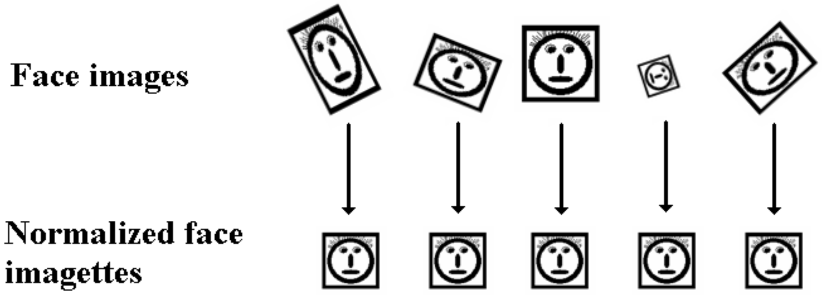

Figure 4.8: The face region normalization make facial features roughly located at the same position

Chapter 5

Head Pose Estimation using linear auto-associative memories

This chapter explains our coarse head pose estimation process. The face tracker described in the previous section isolates a face region within an image. We then project this region of the image into a small fixed-size imagette using a transformation that normalizes size and slant orientation. Normalized face imagettes of the same head pose are used to train an auto-associative memory which acts as a head pose prototype. To enhance the accuracy of the estimation, we use the Widrow-Hoff correction rule to train prototypes. Classification of head poses is obtained by comparing normalized face imagettes with those reconstructed by the auto-associative memory. The head pose whose prototype obtains the highest score is selected.

The first part of this chapter describes the use of linear auto-associative memories. The Widrow-Hoff correction rule is described in the second section. We develop their application to head pose estimation on known and unknown subjects in the third part of the chapter. Performance and comparison with human abilities are discussed in the last section.

5.1 Linear auto-associative memories

Linear auto-associative memories are a particular case of one-layer linear neural networks where input patterns are associated with each other. They were first introduced by Kohonen [70] to save and recall images. Auto-associative memories associate images with their respective class, even when the image has been degraded or partially occluded. With this approach, each cell corresponds to an input pattern. Linear auto-assocative memories allow the creation of prototypes of image classes.

We describe a grey-level input image x' by its normalized vector $x = \frac{x'}{\|x'\|}$. A set of M images composed of N pixels of the same class are stored into a N x M matrix $X = (x_1, x_2, ..., x_M)$. The linear auto-associative memory of a class k is represented by its N x N connection matrix W_k. The number of cells in the memory is equal to the square number of pixels of images x_k. The cost of computing its linear auto-associative memory is $O(N^2)$. The output of a given cell is the sum of its inputs weighted by the connection cells. Thus, the reconstructed image y_k is obtained by computing the product between the source image x and the connection weighted matrix W_k:

$$y_k = W_k \cdot x \tag{5.1}$$

The similarity between the source image and a class k of images is estimated as the cosine between the vectors x and y_k:

$$cos(x, y) = y^T.x = \frac{y'^T.x'}{\|y'^T\|\|x'\|} \tag{5.2}$$

As the vectors x and y are normalized in energy, their cosine delivers a score between 0 and 1, where a similarity of 1 corresponds to a perfect match.

An auto-associative memory must be trained to recognize images of a target class. The steps of the creation of an auto-associative memory are described in figure 5.1. The first learning method for W was proposed by Hebb. This rule consists in increasing the value of a connection cell if its input and its output cells are activated simultaneously. In the case of auto-associative memories, each image x_k of a class k is both its the input and its ouput. The connection matrix W_k of a class k is then initialized by addition of autoassociations of each face vector x_k with itself:

$$W^{t+1} \quad = \quad W^t + \eta \cdot x \cdot x^T \tag{5.3}$$

where η is an adaptation step. This give gives us:

$$W_k = X_k \cdot X_k^T = \sum_{i=1}^{M} x_{ik} \cdot x_{ik}^T \tag{5.4}$$

Reconstructed images with the Hebbian learning are equal to the first eigenface of the image class. Furthermore, the terms of the correction matrix W can have an infinite growth with the number of iterations. To improve recognition abilities of the memory, we learn W with the Widrow-Hoff correction rule.

5.2 The Widrow-Hoff correction rule

The Widrow-Hoff correction rule is a local supervised learning rule aiming at increasing the performance of associators [148]. At each presentation of an image, each cell of the connection

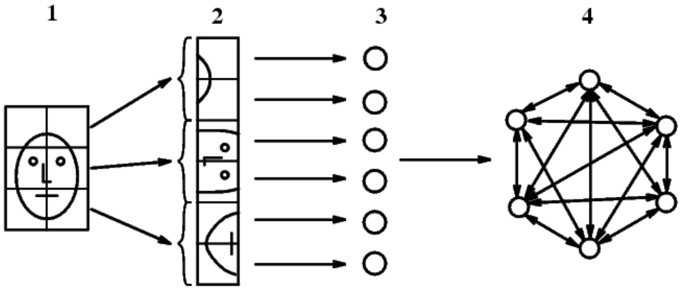

Figure 5.1: Creation of an auto-associative memory from a facial image: 1. Detection of the image, 2. Decomposition into image pixels, 3. Each element serves as an input of the auto-associative memory, 4. Training of the auto-associative memory [148]

matrix modifies its weights from the others by correcting the difference between the response of the system and the desired response. Images X of the same class are presented iteratively with an adaptation step so that the weights in W changes until all images are correctly classified. As a result, the connection matrix W becomes spherically normalized [1]. At each iteration, the weight matrix is updated by injecting the difference into the memory. Adjustments are repeated for all images of the same class until images are perfectly reconstructed.

For linear auto-associative memories, the Widrow-Hoff learning rule is described by:

$$W^{t+1} \quad = \quad W^t + \eta(x - W^t \cdot x)x^T \tag{5.5}$$

where η is the adaptation step and t indicates the current iteration. At each presentation of a class image x, the connection matrix W is corrected with regard to the adaptation step η, the difference between the desired response x and the current response $W^t \cdot x$ and the contribution of the input image x. Rather than expecting the presentation of the whole training data, the matrix is corrected locally for each input data. We consider a class of M images $X = (x_1, x_2, ..., x_M)$. As a positive semi-definite matrix, W can be rewritten by the sum of its eigenvectors:

$$W^0 = X \cdot X^T = \sum_{i=1}^{M} x_i x_i^T = \sum_{r=1}^{R} \lambda_r u_r u_r^T = U \Lambda U^T \tag{5.6}$$

where Λ stands for the diagonal matrix of eigenvalues, U is the orthogonal matrix of eigenvectors, R is the rank of the matrix W and I is the identity matrix. We have $U^T \cdot U = I$. Eigenvectors u are ordered according their corresponding eigenvalue λ. This transformation allows us to rewrite the Widrow-Hoff learning rule as a combination of eigenvectors and eigenvalues. The connection matrix W^t can be expressed as follows:

$$W^t = U \Phi_t U^T \tag{5.7}$$

with

$$\Phi_t = I - (I - \eta\Lambda)^{t+1} \tag{5.8}$$

We recursively obtain the following relation:

$$\begin{aligned}
\Phi_{t+1} &= I - (I - \eta\Lambda)^{t+2} \\
&= I - (I - \eta\Lambda)(I - \eta\Lambda)^{t+1} \\
&= I - (I - \eta\Lambda) + (I - \eta\Lambda) + (I - \eta\Lambda)(I - \eta\Lambda)^{t+1} \\
&= \eta\Lambda + (I - \eta\Lambda)(I - (I - \eta\Lambda)^{t+1}) \\
\Phi_{t+1} &= \eta\Lambda + (I - \eta\Lambda)\Phi_t & (5.9)
\end{aligned}$$

By applying the Widrow-Hoff correction rule, we verify the relation:

$$\begin{aligned}
W_{t+1} &= W^t + \eta(X - W^t X)X^T & (5.10) \\
&= U\Phi_t U^T + \eta X X^T - \eta U\Phi_t U^T U\Lambda U^T \\
&= U\Phi_t U^T + \eta U\Lambda U^T - \eta U\Phi_t \Lambda U^T \\
&= U(\Phi_t U^T + \eta\Lambda - \eta\Lambda\Phi_t)U^T \\
&= U(\eta\Lambda + (I - \eta\Lambda)\Phi_t)U^T \\
W_{t+1} &= U\Phi_{t+1}U^T
\end{aligned}$$

This reformulation exhibits the fact that the correction rule only affects the eigenvalues of the connection matrix W. This process is called eigenvalues equalization or sphericization of the matrix. With a well chosen adaptation weight η , the term $(I - \eta\Lambda)^{t+1}$ tends to 0 at infinite, and the matrix W converges to UU^T. The reconstructed image y_i is then represented by a weigted sum of its eigenvectors:

$$y_i = \sum_{r=1}^{R} u_r u_r^T x_i \tag{5.11}$$

Eigenvectors act as global features of the whole image. That is why linear auto-associative memories are considered as a global approach. The error matrix E is defined as the difference between the source image X and its reconstructed image $W_t X$:

$$E = X - W_t X \tag{5.12}$$

We compute the error function as the quadratic squared sum of the elements of E:

$$\begin{aligned}
Err(W) &= \frac{1}{2}\sum_{j,k} e_{jk}^2 \\
Err(W) &= \frac{1}{2}\sum_j \sum_k (x_{jk} - \sum_i w_{ij}x_{ik})^2 \\
Err(W) &= \frac{1}{2}\sum_j \sum_k (x_{jk} - \sum_i w_{ij}x_{ik})^2 \qquad (5.13)
\end{aligned}$$

The Widrow-Hoff correction rule minimizes the quadratic error due to classification in a least squares sense. The optimal correction term Δw_{ij} of the connection matrix is given by calculating the variation of the error function:

$$\begin{aligned}
\Delta w_{ij} &= -\eta \frac{\delta Err}{\delta w_{ij}} \\
&= \eta \sum_k (x_{jk} - \sum_i w_{ij}x_{ik})x_{ik} \\
\Delta w_{ij} &= \eta \sum_k (x_{jk} - \sum_i (WX)_{ik})x_{ik} \qquad (5.14)
\end{aligned}$$

which corresponds to the Widrow-Hoff correction rule.

The error function increases with the number of images in the training data. Calculating the error allows us to determine a judicious value for the adaptation step η. A good value for η must converge as fast as possible to 0, whereas a bad value will become higher and higher at each iteration. The error matrix E can be expressed as:

$$\begin{aligned}
E &= X - U\Phi_t U^T X \qquad (5.15) \\
&= X - U(I - (I - \eta\Lambda)^{t+1})U^T X \\
&= X - UU^T X + U(I - \eta\Lambda)^{t+1}U^T X \\
&= X - X + U(I - \eta\Lambda)^{t+1}U^T U\Lambda U \\
&= U(\eta\Lambda + (I - \eta\Lambda)^{t+1}U^T \\
E &= U(I - \eta\Lambda)^{t+1}\Lambda U^T \qquad (5.16)
\end{aligned}$$

The error matrix converges to 0 if and only if:

$$\lim_{t \to +\infty} (I - \eta\Lambda)^{t+1} = 0 \qquad (5.17)$$

Which is equivalent to:

$$\forall r \leq R \qquad \lim_{t \to +\infty} (1 - \eta \lambda_r)^{t+1} = 0 \tag{5.18}$$

We can see that the elements of the error matrix are influenced by the terms $(1-\eta\lambda_r)^{t+1}$. As a consequence, and as the current iteration t is a natural number, the error function is influenced by the terms $\hat{e}_t = ((1-\eta\lambda_r)^2)^{t+1}$. We want $\hat{e}_t$ to converge as fast possible to 0. However, this is not possible for real data to obtain for each eigenvalue $\hat{e}_t = 0$. At each iteration, the term $\hat{e}_t$ is multiplied by $(1-\eta\lambda_r)^2$, so this value must be as close to 0 as possible. The adaptation step η must be regulated so that $\eta\lambda_r$ is close to 1. A value of 0 for η leads to a stagnation of the error. If η is too small, the error decreases slowly. If η is too high, the term $(1-\eta\lambda_r)$ becomes greater than 1 and converges to infinite. Thus, there is an optimal value for the adaptation step η. To obtain the convergence of the error function, we must have for each eigenvalue λ_r:

$$\forall r \leq R \qquad 0 < \eta < \frac{2}{\lambda_r} \tag{5.19}$$

By considering the higher eigenvalue λ_{max} , this condition can be reformulated as:

$$0 < \eta < \frac{2}{\lambda_{max}} \tag{5.20}$$

Beyond this value, some of the terms $(1-\eta\lambda_r)^{t+1}$ increase and, as a consequence, the error function raises quickly. The case in which the error is equal to 0 corresponds to an infinite number of iterations and leads to overlearning. Only images trained with the auto-associative memory would be perfectly reconstructed. The system would not learn the image class, but each image part of the class. However, the algorithm must be able to learn intra-class variations. As we want our system to be adaptive and to correctly classified unknown images belonging to the class, it is better to have a fixed number of iterations ι.

Figure 5.2 shows examples of reconstructed images using Hebbian and Widrow-Hoff learning rule. The memory trained with Hebbian rule gives the same response for every image. As a consequence, the cosine between original and reconstructed images is not discriminant enough to classify images while the memory trained with the Widrow-Hoff correction rule provides more discrimination. In-class images are minimally deformed by multiplying with the connection matrix, while extra-class images are more strongly deformed. The reconstruction improves with learning. With a good choice of the adaptation step η and the number of iteration ι, an image of the class can be well reconstructed from the memory, even in cases of partial occlusion. Another advantage of using the Widrow-Hoff learning rule is that outliers are not taken into account during the training phase. By training images of a class made up of a majority of a certain type of images and a minority of outliers, the weights calculated by the correction rule can be optimized to recognize the majoritary type of images, and not outliers. The Widrow-Hoff learning rule has shown good results on classic face analysis problems in the case of images from a single camera, such as face recognition, sex classification and facial type classification.

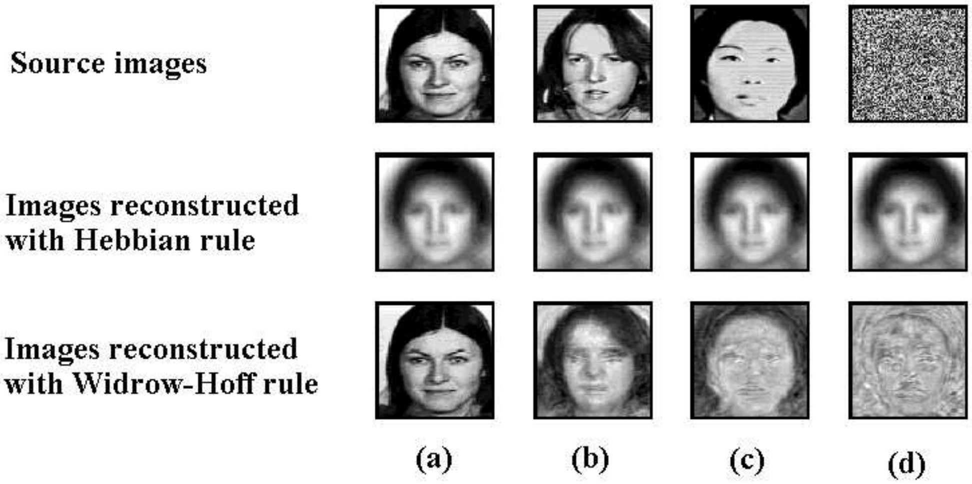
Source images
Images reconstructed with Hebbian rule
Images reconstructed with Widrow-Hoff rule
(a)
(b)
(c)
(d)

Figure 5.2: Reconstruction of images with an auto-associative memory trained either by a standard hebbian learning rule or a Widrow-Hoff correction rule. The memory has been trained with female caucasian facial images. The memory delivers the same reconstructed image for every source image using the Hebbian learning rule. It cannot discriminate caucasian facial images, nor even distinguish a face from a random pattern. We now consider images reconstructed with the Widrow-Hoff learning rule. As belonging to the training data, image (a) is perfectly reconstructed by the memory. A perfect match is obtained. Image (b) does not appear in the training data, but represents a caucasian face. It is a little degraded during reconstruction, but a good match is obtained. As a Japanese face image, image (c) does not belong to the class and is strongly degraded, resulting in a poor match. Image (d) represents a random pattern, its match with the reconstructed image is close to 0 [148].

Linear auto-associative memories trained with the Widrow-Hoff correction rule increases the performance of PCA [1]. The number of principal components does not need to be defined, because all dimensions are used. Contrarly to neural networks, it is not necessary to specify the choice of the structure or the number of cells in hidden layers is not required. Only two parameters, the adaptation step η and the number of iterations ι are required. Furthermore, reconstruction is robust to partial occlusions. Using non-linear memories or neural networks with hidden layers prevents creation and storage of prototypes of image classes. Linear auto-associative memories allows us to create prototypes W_k of image classes that can be saved, recovered and directly reused for other experiments. We apply this approach to the head pose estimation problem.

5.3 Application to head pose estimation

We consider each head pose as a class of images. A linear auto-associative memory W_k is trained for each head pose class k. As for our experiments in Chapter 3, we use the Pointing'04 Head Pose Image Database to measure the performance of auto-associative memories on head pose estimation. There are 13 poses for pan and 9 poses for tilt. To estimate head pose on a given face imagette, a simple winner-takes-all process is employed [40]. For a test image X, the pose k whose memory W_k obtains the best match is selected. We compute the cosine between the source image and the reconstructed images $W_k X$ as indicated in equation 5.21. The computional complexity of the estimation is linear with regard the number of classes N_p. Two experiments are performed using this approach: head poses are trained either separately or together.

$$Pose = argmax_k(cos(X, W_k \cdot X)) \tag{5.21}$$

Concerning the normalization of the face region, we can see that this is a crucial preprocessing step for the use of linear auto-associative memories to head pose estimation. For one thing, all images in the training data must have the same size to enable the creation of the head pose prototype. In addition, normalization allows us to have facial features found at the same location in all of the imagettes for a given head pose, which is appropriate for linear auto-associative memories where all pixels are compared locally.

5.3.1 Learning separate head poses

To train separate head poses, we learn each angle on an axis while varying the angle of the other axis. A pose is represented either by a pan angle, or a tilt angle. Each linear auto-associative memory corresponding to a pan angle is trained with varying tilt angle. Similarly, each memory corresponding to a tilt angle is trained with a varying pan angle. The learning process is explicited in figure 5.3. For P pan angles and T tilt angles, this approach delivers $N_p = P + T$ head poses protoypes. We obtain 13 classifiers for pan angle and 9 classifiers for tilt angle:

$$W_{Pan=-90}, W_{Pan=-75}, W_{Pan=-60}, W_{Pan=-45}, W_{Pan=-30}, W_{Pan=-15}, W_{Pan=0},$$
$$W_{Pan=+15}, W_{Pan=+30}, W_{Pan=+45}, W_{Pan=+60}, W_{Pan=+75}, W_{Pan=+90}$$

$$W_{Tilt=-90}, W_{Tilt=-60}, W_{Tilt=-30}, W_{Tilt=-15}, W_{Tilt=0},$$
$$W_{Tilt=+15}, W_{Tilt=+30}, W_{Tilt=+60}, W_{Tilt=+90}$$

Figure 5.4 shows the variation of the error computed on front pan and tilt poses with regard to the adaptation step. We use an adaptation step η of 0.008 for pan axis and 0.006 for tilt axis for our experiment.

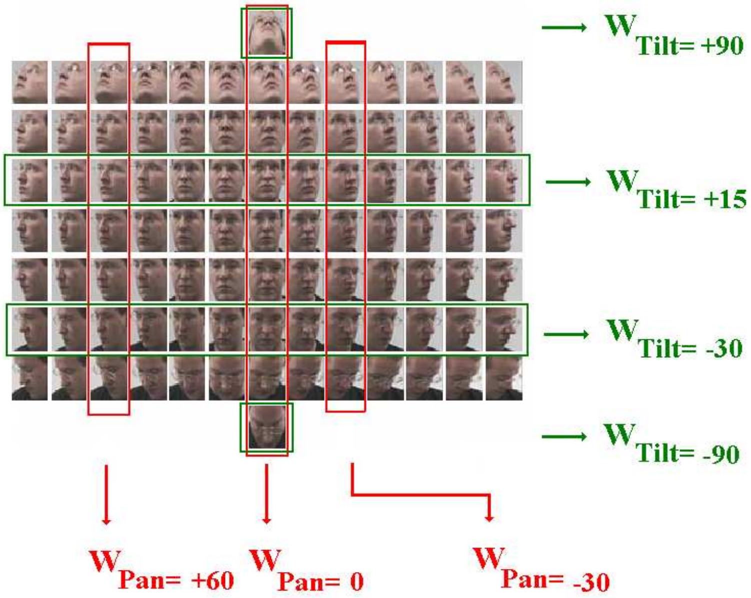

Figure 5.3: Training of linear auto-associative memories on separate head poses

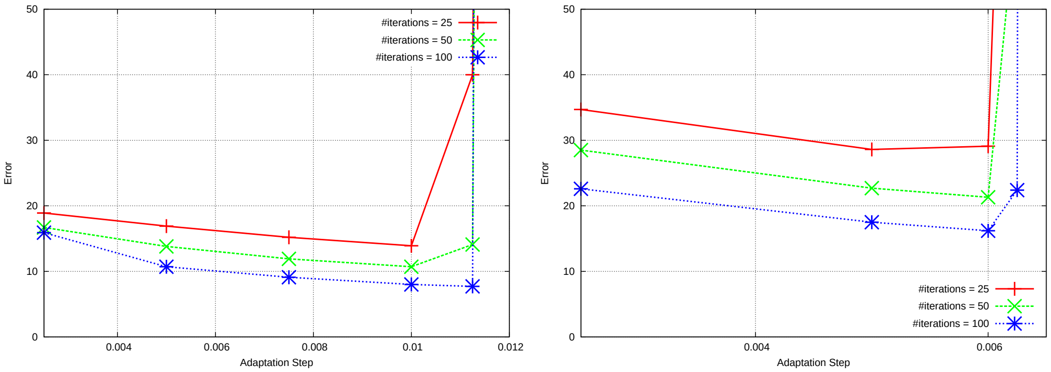

Figure 5.4: Error computation for front separate pan and tilt poses with varying the adaptation step and the number of iterations

5.3.2 Learning grouped head poses

In the grouped head pose experiment, pan and tilt angle are trained together. A pose is represented by a couple of pan and tilt angles. Each linear auto-associative memory is trained from facial images with the same head pose. The learning process is explicited in figure 5.5. This approach delivers $N_p \simeq P \times T$ head poses protoypes. We obtain 93 classifiers:

$$W_{Pan,Tilt=0,-90}$$

$$W_{Pan,Tilt=+90,-60}, W_{Pan,Tilt=+75,-60}, \ldots, W_{Pan,Tilt=-75,-60}, W_{Pan,Tilt=-90,-60}$$

$$W_{Pan,Tilt=+90,-30}, W_{Pan,Tilt=+75,-30}, \ldots, W_{Pan,Tilt=-75,-30}, W_{Pan,Tilt=-90,-30}$$

$$\vdots, \vdots, \vdots, \ldots, \vdots, \vdots, \vdots$$

$$W_{Pan,Tilt=+90,+30}, W_{Pan,Tilt=+75,+30}, \ldots, W_{Pan,Tilt=-75,+30}, W_{Pan,Tilt=-90,+30}$$

$$W_{Pan,Tilt=+90,+60}, W_{Pan,Tilt=+75,+60}, \ldots, W_{Pan,Tilt=-75,+60}, W_{Pan,Tilt=-90,+60}$$

$$W_{Pan,Tilt=0,+90}$$

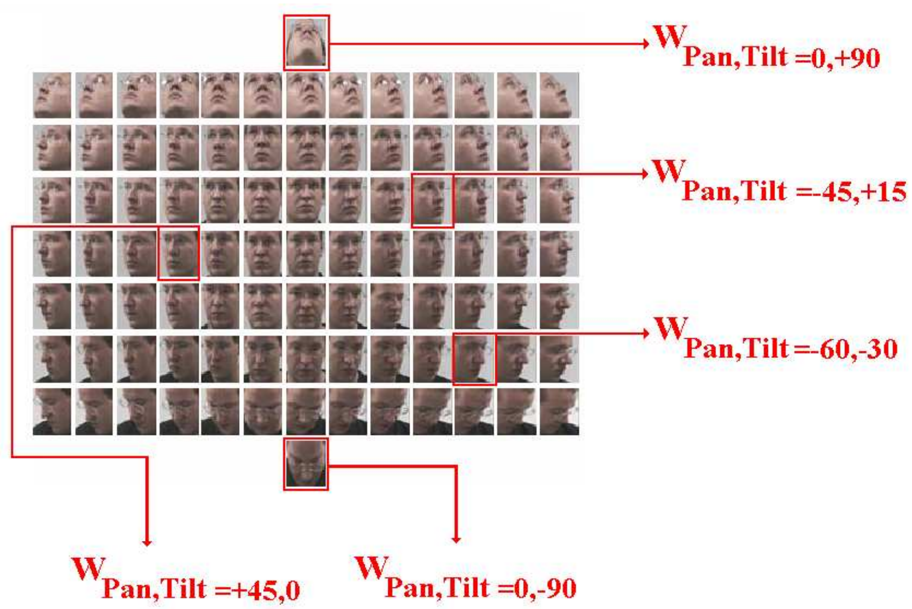

Figure 5.5: Training of linear auto-associative memories on grouped head poses

Figure 5.6 shows the variation of the error computed on front pose with regard to the adaptation step. We use an adaptation step η of 0.07 for this experiment.

5.3.3 Testing on known users

To measure the performance of our system on known users, training and testing using a 2-fold cross-validation on the two sets of the Poitning 2004 database. During the first pass, the first set is used as training data, and the second one as test data. During the second pass, the roles are reversed. This is an exhaustive test method. The number of training images for each pose M is equal to 15. The 2-fold cross-validation algorithm procedure is described below:

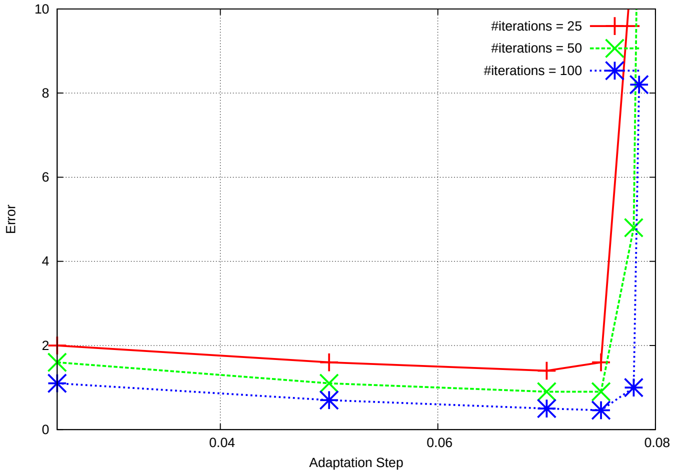

Figure 5.6: Error computation for front pose with variations of the adaptation step and the number of iterations

```
Train the 1st set
Test the 2nd set
Train the 2nd set
Test the 1st set
```

5.3.4 Testing on unknown users

To measure the performance of our system on unknown users, training and testing are performed using the Jack-Knife method, also known as the leave-one-out algorithm. Testing is done only on unknown users, which allows us to see whether linar auto-associative memories really capture the head pose information. This is also an exhaustive test method. The number of training images for each pose M is equal to 28. The Jack-Knife algorithm procedure is described below:

```
For all subjects i
Train all subjects except i
Test subject i
```

5.4 Results and discussion

In this section, we compare results of the two experiments on the images of the Pointing'04 Head Pose image database. Training and testing can be done either on known users or unknown users. To have an idea of the efficiency of our system in man-machine interaction applications, we compare performance of our system with human performance obtained in Chapter 3.

5.4.1 Evaluation Measures

We use the evaluation measures previously defined in section 3.4.1: the mean absolute error, the correct classification rate and average error per pose. We define another measure, the correct pan classification rate with 15 degrees error. Its calculation is explicited by the equation 5.22. An image is correctly classified with 15 degrees if the absolute difference $\|p(k) - p^*(k)\|$ does not exceed 15 degrees. This measure is useful to determine the proportion of images whose head poses can be refined in a later experiment.

$$CorrectClassification15 \quad = \quad \frac{Card\{ImagesCorrectlyClassified15^o\}}{Card\{Images\}} \tag{5.22}$$

The influence of the number of iterations ι with separate and grouped training is shown respectively on figures 5.7 and 5.8. We can see that beyond 70 iterations, the mean average error on pan and tilt axis becomes stagnant. Thus we will use a number of iterations $\iota = 70$ in our experiments.

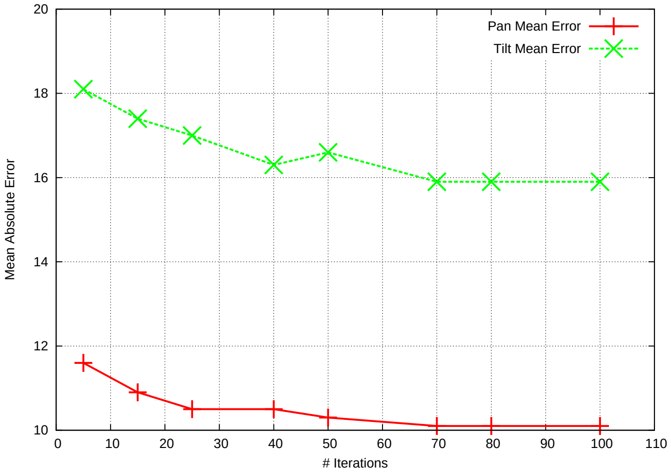

Figure 5.7: Mean average error in pan and tilt with regard to the number of iterations ι with the separate training

5.4.2 Performance

We compare performance of our system with those obtained by some other methods of the state of the art. For testing on known users, we compare our results to those obtained by tensor models, PCA, Locally Embedded Analysis [145] and neural networks [152]. The evaluation measures are calculated with the same data. For testing on unknown users, we compare our results to neural networks developed by Stiefelhagen [137] as well as to closest picture search.

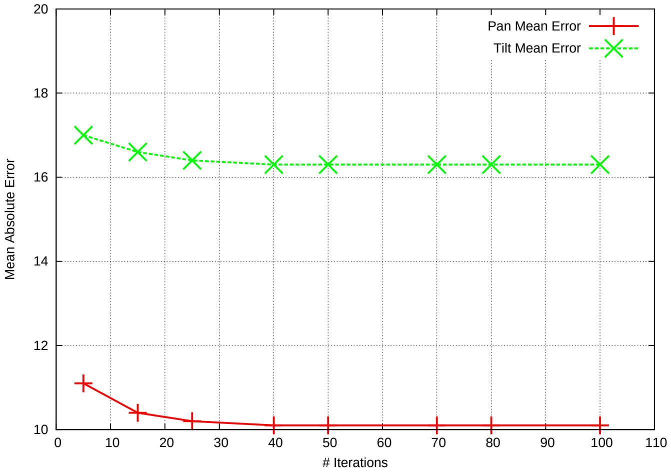

Figure 5.8: Mean average error in pan and tilt with regard to the number of iterations ι with the grouped training

Closest picture algorithm consists in finding the image in the training data which obtains the best match with the image to test. The head pose of the selected image is chosen as the head pose of the test image. The match is done using direct cosine computation. This algorithm can be performed either by estimating pan and tilt angles separately, or by estimating pan and tilt angles together. However, it cannot create head pose prototypes from training images. Furthermore, the closest picture search algorithm has a comoputationnal complexity of $O(MN_p)$ where M is the number of images for each pose in the training data. Closest picture search is computationally more expensive than the linear auto-associative memories, whose complexity is $O(N_p)$, because $MN_p \gg N_p$ and all images of the training data have to be browsed for each test image.

Evaluation results are shown in tables 5.1 and 5.2 . With the separate training for pan and tilt, we can see that pan angle is well recognized with an average error of 7.6 degrees for known users and 10.1 degrees for unknown users. As a comparison, neural networks obtain 12.4 degrees of error for unknown users. Average error is 8.4 degrees for known users and 10.1 degrees for unknown users using the grouped learning. The average tilt error is 11.2 degrees for known users and 15.9 degrees for unknown users using the separate training. Using the grouped learning, the error is 8.9 degrees on known users and 16.3 degrees on unknown users.

Head pose prototypes learned with linear auto-associative perform well for known and unknown users. The comparison to closest image algorithm search shows the utility of gathering images of the same class into a connection matrix.

Average error per pose is shown on figure 5.9. Concerning the pan angle, the average absolute error in pose is relatively stable with both methods. The minimal error can be found at front and profile poses. Separate and grouped learning accommodate well with intermediate tilt angles. Linear auto-associative memories provide better results than searching for the closest image in the training database.

Evaluation measure	Tensor	PCA	LEA	NN	**Sep. LAAM**	**Grp. LAAM**
Pan Average Error	12.9^o	14.1^o	15.9^o	12.3^o	**7.6**^{o}	**8.4**^{o}
Tilt Average Error	17.9^o	14.9^o	17.4^o	12.8^o	**11.2**^{o}	**8.9**^{o}
Pan Classification 0^o	49.3 %	55.2 %	45.2 %	41,8 %	**61.2 %**	**59.4 %**
Tilt Classification 0^o	54.9 %	57.9 %	50.6 %	52.1 %	**54.2 %**	**62.4 %**
Pan Classification 15^o	84.2 %	84.3 %	81.5 %	-	**92.4 %**	**90.8 %**

Table 5.1: *Performance evaluation on known users. NN refers to Neural Networks and LAAM refers to Linear Auto-Associative Memories [40, 145, 152].*

Evaluation measure	Separate CP	Grouped CP	**Separate LAAM**	**Grouped LAAM**
Pan Average Error	14.1^o	13.9^o	**10.1**^{o}	**10.1**^{o}
Tilt Average Error	15.9^o	21.1^o	**15.9**^{o}	**16.3**^{o}
Pan Classification 0^o	40.9 %	40.9 %	**50.3** %	**50.4 %**
Tilt Classification 0^o	41.9 %	41.5 %	**43.9 %**	**45.5 %**
Pan Classification 15^o	80 %	80.1 %	**88.8 %**	**88.1 %**

Table 5.2: *Performance evaluation on unknown users. CP refers to Closest Picture and LAAM refers to Linear Auto-Associative Memories*

We achieve a precise classification rate of 61.2% for pan angle and 54.2% for tilt angle on known users and 50.4% for pan angle and 44% for tilt angle on unknown users with the separate pan and tilt pose training. Using the grouped pose training technique provides a 59.4% classification rate for pan angle and 62.4% for tilt angle for unknown users. Pan angle can be correctly estimated with a precision of 15 degrees in more than 88% of cases with both methods on all subjects. Neural networks used by Stiefelhagen obtain a pan classification rate of 38.8 % with 0 degree precision and 69.1 % with 15 degrees precision.

These results demonstrate that linear auto-associative memories are suitable to head pose estimation with known and unknown subjects. We can see that using the grouped learning technique does not significantly improve results. Furthermore, the system runs faster at 15 images/secs with prototypes trained separately than at 1 image/secs with prototypes trained together. This is due to the fact that $P + T \ll P \times T$. During the selection of the best match, there are only 22 separate prototypes tested versus 93 grouped prototypes. Learning poses and pan and tilt axis separately provide a significant gain of computational time without loss of performance.

Faces are not aligned in the Pointing'04 database. Normalizing face images provides small variations in alignment. Experiments demonstrate that our system can handle alignment problems. Computing a score for each memory allows us to discriminate face and non-face images.

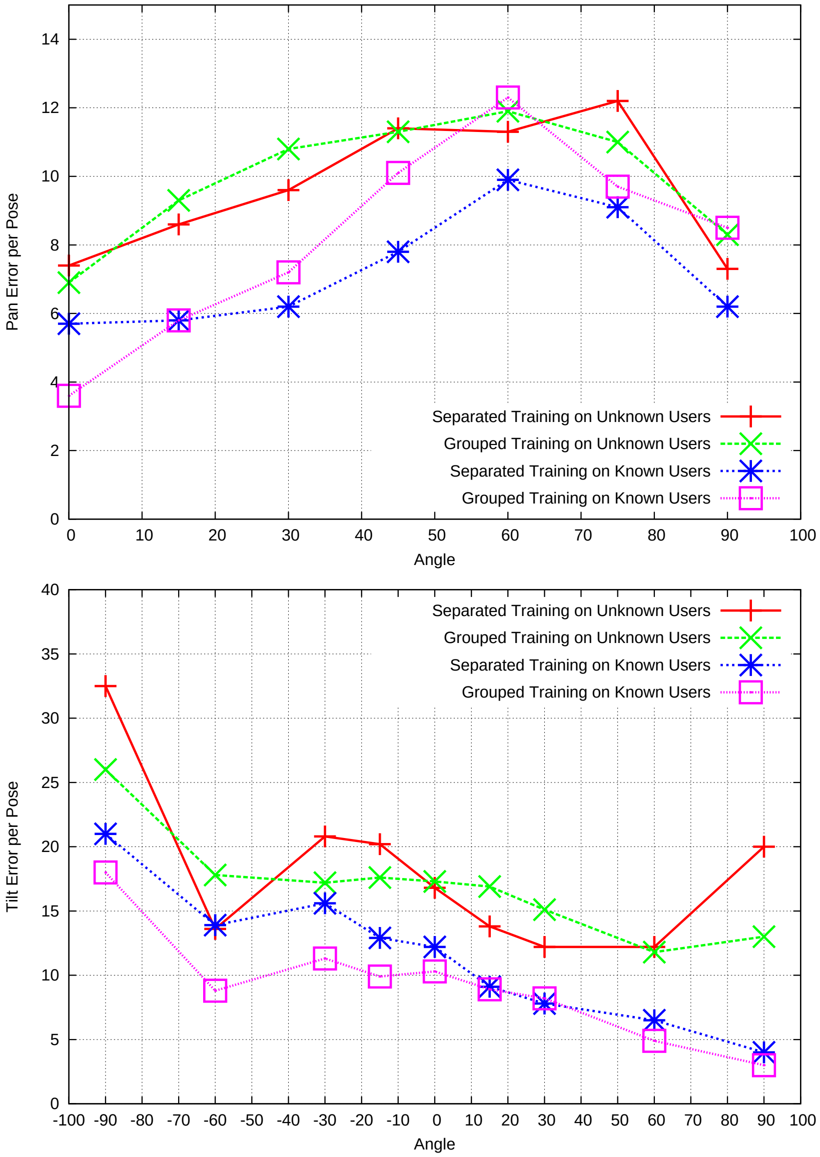

Figure 5.9: Average error per pose on known and unknown subjects on pan and tilt axis

Head detection and pose estimation are done in a single process. The results obtained with Jack-Knife show that our system generalizes well to previous unseen subjects and is robust to identity. As humans estimated angles separately in our experiment, we use the separate prototypes for comparison with human performance.

5.4.3 Comparison with human performance

As we use the same evaluation measures, we can compare performance of our system on unknown users with humans. In our experiment, humans were asked to estimate pan and tilt angles separately, so we will compare their performance to linear auto-associative memories trained separately. Results are shown in table 5.3. As in chapter 3, we use the test of Student-Fisher to determine if the difference of performance betwen two populations is significant.

Evaluation Measure	C Subjects	NC Subjects	S LAAM KU	S LAAM UU
Pan Average Error	11.8^o	11.9^o	7.6^o	10.1^o
Tilt Average Error	9.4^o	12.6^o	11.2^o	15.9^o
Pan Classification 0^o	40.7 %	42.4 %	61.2 %	50.3 %
Tilt Classification 0^o	59 %	48 %	54.2 %	43.9 %

Table 5.3: *Performance comparison between humans and our system. C and NC refer respectively to Calibrated and Non-Calibrated subjects, S LAAM refers to Separate Linear Auto-Associative Memories, and KU and UU refer respectively to Known*

With an average error of respectively 7.6 and 10.1 degrees and a correct classification rate higher than 50% on known and unknown users, our method performs significantly better than humans at estimating pan angle, with an average error of 11.9 degrees. The standard deviation of the average error per pose is low for the system and high for humans. Average error per pose is illustrated on figure 5.10. The system achieves roughly the same precision for front and profile, and higher precision for intermediate poses. As for humans, minimal error can be found at front and profile poses. This means that our algorithm can handle a wide range of head movements.

With an average error of 11.2 degrees in tilt angle angle, our system achieves a comparable performance to humans for known users. However, humans perform significantly better in tilt angle than our system for unknown users. Our method performs well for top poses. This is due to the fact that hair becomes more visible on the image and the face appearance between people changes more when looking down. On the other hand, such changes are less visible for upward poses. Face region normalization also introduces a problem. The height of the neck differs from one person to another. This provides high variations on face imagettes and can disrupt tilt angle estimation.

This chapter proposes a new method to estimate head pose with unconstrained images. Face image are normalized in scale and slant and projected onto an standard size imagette by a robust face detector. Face imagettes containing the same head pose are learned with the Widrow-Hoff correction rule to obtain a linear auto-associative memory. To estimate head pose, we compare source and reconstructed images using their cosine. A simple winner-takes-all process is applied to select the head pose whose memory gives the best match.

We achieved an accuracy comparable to human performance on known users. Our method

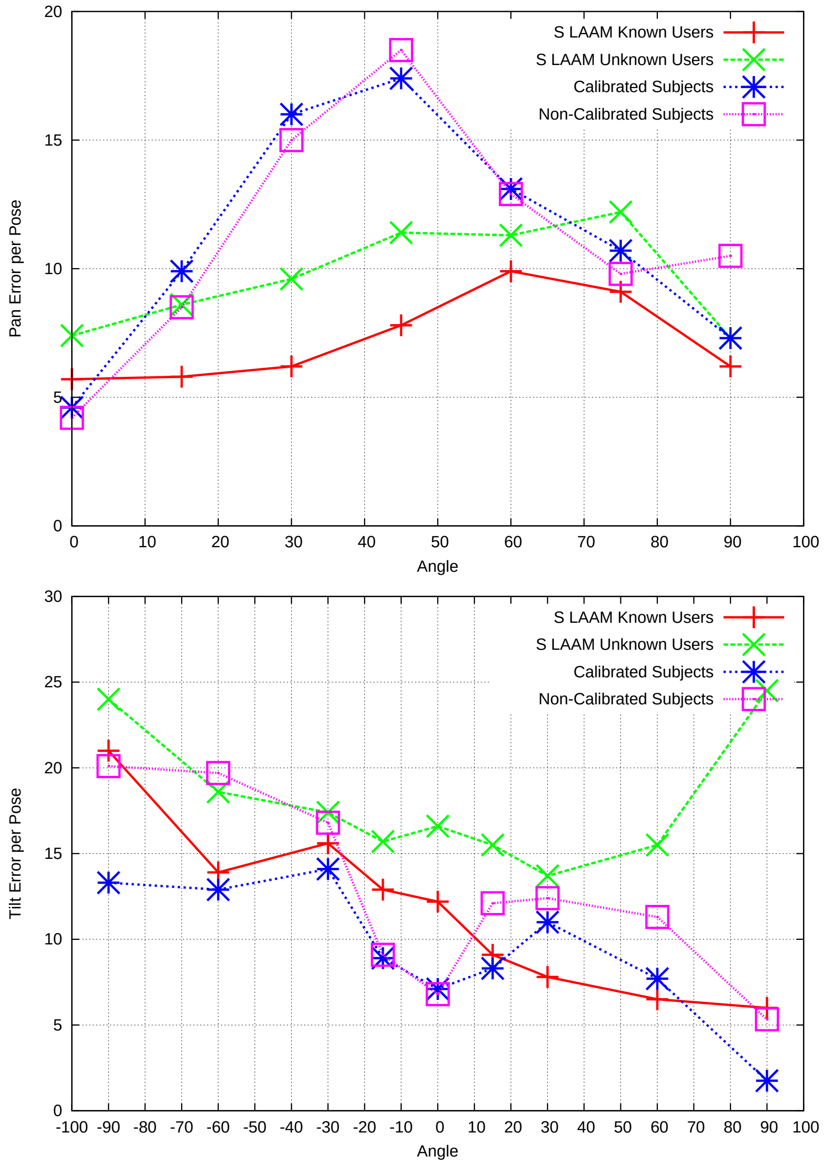

Figure 5.10: Average error per pose on pan and tilt axis

requires very few parameters and can provide good results on very low resolution face images and can handle wide movements, which is particularly adaptated to wide-angle or panoramic camera setups. Furthermore, the system is particularly appropriate for known users, but also generalizes well to unknown users. Our method is robust to alignment and runs at 15 frames/secs. Another advantage of using linear auto-associative memories is the creation of head pose prototypes, which can be saved and restored for other applications. These operations are more difficult with subspaces or neural networks with hidden layers. Our head pose estimation algo-

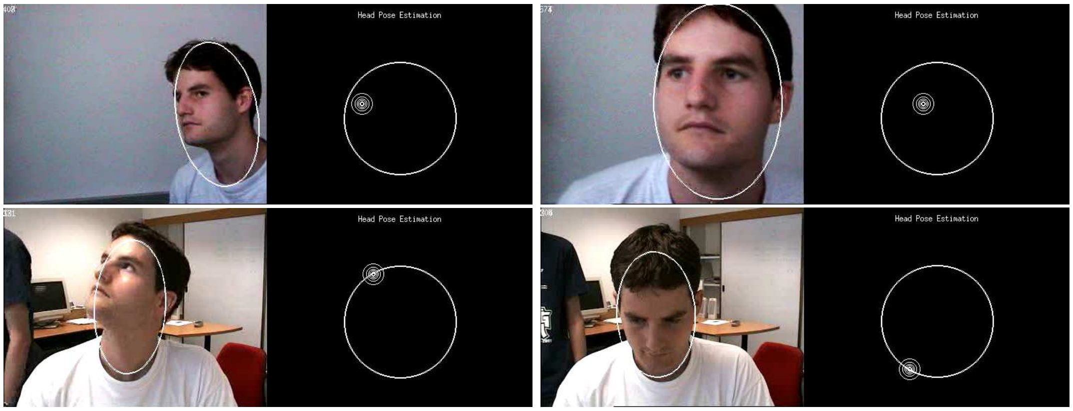

Figure 5.11: Sample video of face and head pose tracking on a known user. Face regions are normalized into 23x30 pixels imagettes. The inner circle represents the estimated head pose.

rithm is reliable and convenient enough for video sequences for applications in man-machine interactions, video surveillance and intelligent environments.

Linear auto-associative memories performs very well for known users, as they were originally designed for exact recognition of images from the training data. Even if a partial occlusion or a partial change occurs, memories can recover images from learned classes and estimate head pose. An example of head pose tracking in real conditions on a subject of the Database is shown on Figure 5.11. Increasing the size of the normalized imagette do not significantly increase the accuracy of the estimation. Results for unknown users can be improved by increasing the size of training images. However, the algorithm has a quadratic complexity with regard to the size of the imagette. We use another face description method based on local information in case of higher resolution images to increase the performance for unknown users.

Below is a summary of our coarse pose estimation algorithm:

Training:

```
For each group of poses $k$:
    Initialize a connection matrix $W_k$
    For each image $X_k \in k$:
        Train $W_k$ using the Widrow-Hoff correction rule:
        $W_k^{t+1} = W_k^t + \eta(X_k - W_k^t \cdot X_k)X_k^T$
```

Testing:

```
Given a test image $Y$,
For each group of poses $k$:
    Compute the reconstructed image $Y_k = W_k \cdot Y$
    Compute the cosine $cos(Y, Y_k) = Y_k^T.Y$
Select the class $k$ which obtains the highest cosine:
$k_{coarse} = argmax_k(cos(Y, Y_k))$
The estimated coarse pose of the image $Y$ is $k_{coarse}$
```

Chapter 6

Face Description using Gaussian Receptive Fields

This chapter describes perception of face images with receptive fields or local linear functions. Gaussian kernels are used to compute the response vectors for these descriptors. When normalized to local intrinsic scale, Gaussian receptive fields appear to be a good detector for salient facial features robust to illumination, pose and identity. The first part of this chapter explains the principles of receptive fields and their properties when computed with Gaussian derivatives. In the second part, the process of automatic scale selection is detailed. The third part of the chapter concerns salient facial feature detection.

6.1 Gaussian receptive fields

Features of intermediate complexity robust to scale, illumination and position changes are used by primates for vision and object recognition [141]. Our objective is to design such local descriptors. Gabor wavelets can be used to detect scale-invariant feature points, as presented in [161] and [73]. However, they have parameters that are difficult to adjust and tend to be computionally expensive. Similar information can be obtained from a vector of Gaussian derivatives, with the advantage that very fast techniques exist for computing scale normalized Gausian derivatives [19]. Gaussian derivatives describe the appearance of neighbourhoods of pixels and are an efficient means of computing scale and illumination robust local features. Furthermore, they have interesting invariance properties.

We describe face images with Gaussian receptive fields. The term ’receptive field" designates a receptor that describes the local patterns of reponses to intensity changes in images. This term comes from sudies of mammalian vision and refers to a pattern of photo-sensitive receptors in the primary visual cortex [54]. Such a structure acts as a weighted region on the retina. Receptive fields in computer vision are used by many researchers under different names. For example, they are used by Koenderink et al. as local measurement of the n^{th} order image structure [69], by Roa and Ballard as iconic feature vector [113], by Schmid to detect natural interest points [118], by Mikolajczyk and Schmid [94] to provide affine invariant invariant descriptions of local appearance and by David Lowe to form the Scale Invariant Feature Transform (SIFT) [82]. We prefer the term receptive field as used by Schiele [114], coming from biological vision. In the following, the expression receptive field refers to local linear fonctions based on Gaussian derivatives of inceasing order.

6.1.1 Mathematical Definition

The response $L_{k,\sigma}$ of a grey level image I to a Gaussian receptive field $G_{k,\sigma}$ of scale σ and of direction k is equal to their convolution $L_{k,\sigma} = I \otimes G_{k,\sigma}$, where $\otimes$ denotes the inner product computed at a sequence of positions. The set of values $L_{k,\sigma}$ forms the feature vector L_σ:

$$L_\sigma = (L_{1,\sigma}, L_{2,\sigma}, ..., L_{n,\sigma})$$

The order and the direction k refers to the type of the derivative of the receptive field and has the form $x^i y^j$. Figure 6.1 shows a description of an image neighbourhood using Gaussian receptive fields. For each pixel (x, y), the Gaussian derivative of scale σ is expressed as:

$$G_{x^i y^j,\sigma}(x, y) = \frac{\partial^i}{\partial x^i} \frac{\partial^j}{\partial y^j} G_\sigma(x, y) \tag{6.1}$$

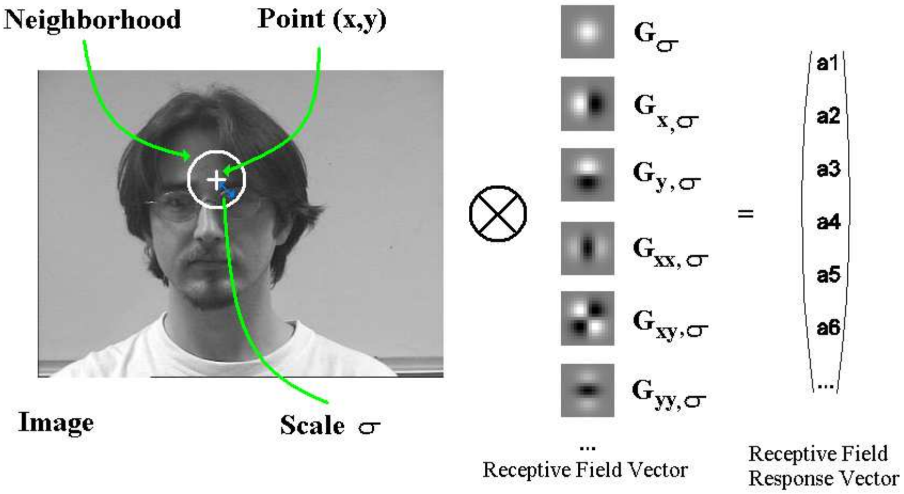

Figure 6.1: Example of neighbourhood description with Gaussian receptive fields

The Gaussian kernel of scale σ is defined in 1D as:

$$G_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}}$$

In 2D, the Gaussian kernel is expressed as follows:

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

A Gaussian receptive field provides a numerical descriptor for local appearance at a particular scale, and position. This descriptor can easily be tuned to local orientation using the steerability property of Gaussian derivatives [27], as well as to affine transformations of local appearance [94]. The space constructed by receptive fields is called the local appearance space or the feature space. Gaussian receptive fields measures the similarity of neighbourhoods of pixels. Two neighborhoods similar in appearance present similar local geometries and are close in the feature space. The similarity of two neighbourhoods of pixels can be measured by computing the distance of Gaussian receptive fields response in the feature space. Furthermore, the Gaussian kernel presents many interesting properties for image description.

6.1.2 Separability

The Gaussian kernel is the unique function that isboth separable and circularly symmetric in Cartesian coordinates:

$$\begin{aligned}
G_\sigma(x, y) &= \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \\
&= \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \cdot \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{y^2}{2\sigma^2}} \\
G_\sigma(x, y) &= G_\sigma(x) \cdot G_\sigma(y) \qquad (6.2) \\
&= \frac{1}{2\pi\sigma^2} e^{-\frac{r^2}{2\sigma^2}} \\
&= \frac{1}{2\pi\sigma^2} e^{-\frac{(rcos\theta)^2+(rsin\theta)^2}{2\sigma^2}} \\
G_\sigma(x, y) &= G_\sigma(r, \theta) \qquad (6.3)
\end{aligned}$$

Where (r, θ) represent the polar coordinates of (x, y). The separability of the Gaussian kernel is an important property in computer vision as it makes it possible to reduce the complexity of computing a multi-dimensional receptive field response. The calculation of the convolution of an image neighbourhood of $n \times n$ pixels with a two dimensonial function requires $O(n^2)$ operations. With the separability of the Gaussian functions, the computationial complexity decreases to $O(2n)$. This property can be extended to n dimensions.

6.1.3 Scalability

Gaussian kernels are self similar over scale and can be easily calculated. They satisfy the following equation:

$$\begin{aligned} G_{t\sigma}(tx) &= \frac{1}{\sqrt{2\pi}t\sigma} e^{-\frac{t^2x^2}{2t^2\sigma^2}} \\ &= \frac{1}{t}\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \\ G_{t\sigma}(tx) &= \frac{1}{t} G_\sigma(x) \end{aligned} \tag{6.4}$$

From this property it is possible to compute an image response to a Gaussian function that does not depend on the scale parameter σ. As stated by Slepian and Pollack [125], the Gaussian is the function which has optimal compactness in frequency and space. Furthermore, the gaussian function is the unique solution to the diffusion equation and is therefore suitable for description of physical images phenomena.

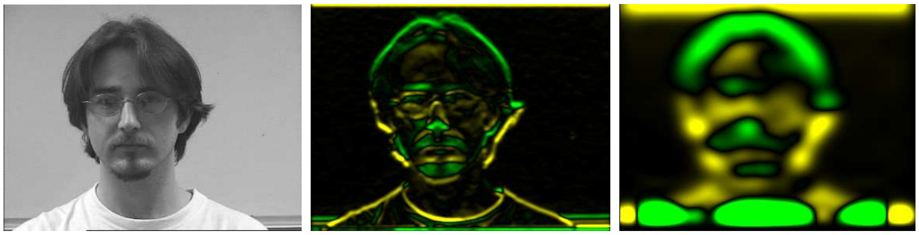

Figure 6.2: Example of receptive fields response to the first gaussian derivative $G_{y,\sigma}$ at different scales σ. The left image is the original image. The middle image is the reponse to the derivative with σ = 2 pixels. The right image is the reponse to the derivative with σ = 10 pixels. Positive values are represented in yellow, negative values are represented in green and zero is represented in black. Original image is 1/4 PAL

6.1.4 Differentiability

The Gaussian function is infinitely differentiable. Any derivative of an image $I \otimes G_\sigma$ blurred by a Gaussian is equal to the convolution of the original image I with the derivative of the Gaussian kernel. Therefore, the image signal can be expressed as Taylor series of Gaussian derivatives:

$$\frac{\partial^n}{\partial x^n}\left[I(x) \otimes G_\sigma(x)\right] = \frac{\partial^n I}{\partial x^n} \otimes G_\sigma(x) = I(x) \otimes G_{n,\sigma}(x) \tag{6.5}$$

The first order derivatives describe the local line orientation in images, whereas line local curvatures are perceived by second order derivatives. We do not take into account zeroth order Gaussian derivatives in order to remain robust to changes in illumination intensity. Derivatives of order strictly superior to 2 have been found to contribute information about appearance only if an important structure is detected in second order terms [68]. For this reason, we take into account derivative terms up to the third order.

We obtain a five dimensional feature vector computed at each pixel by calculating the convolution with the first derivative of a Gaussian in x and y direction (G_x, G_y) and the second derivatives (G_{xx}, G_{xy} and G_{yy}). Our Gaussian receptive field feature vector the image has therefore 5 dimensions: $L_\sigma = (L_{x,\sigma}, L_{y,\sigma}, L_{xx,\sigma}, L_{xy,\sigma}, L_{yy,\sigma})$. The feature vector $L_\sigma(x, y)$ describes the local appearance of the neighbourhood of the pixel (x, y) of scale σ.

As shown on Figure 6.2, large scales describe coarse variations of the image, whereas small scales describe its fine variations. In the following section, we explain how to obtain a reliable value for the best scale parameter σ.

6.2 Automatic scale selection

The notion of scale is one of the most important aspects of computer vision. Observing objects at different scales provides different interpretations. The same image region can be interpreted as an interest feature at a certain scale, and as a spurious region at a different scale. That is why the scale of observation must be specified in image understanding [69]. Many researchers usually describe images at multi-scale [113] or at multi-resolution [82]. Image features are analysed through a set of scales, which provides changing number and appearance of interest features at each scale.

In [78], Lindeberg proposes a method to select appropriate local scales to describe image features. For a given pixel of an image, these relevant scales are called intrinsic scales. A scale profile computed at each pixel provides intrinsic scales[1]. The scale profile of a feature point is obtained by collecting its responses to the normalized Laplacian energy over a range of scales. Local maximas of the scale profile gives maximum responses to the Laplacian and are selected as intrinsic scales. Figure 6.3 shows an example of a feature point and its scale profile. The intrinsic scale is obtained at the zero crossing of the normalized Laplacian energy. The normalized Laplacian operator $\nabla^2 G$ is is invariant to rotation and is defined as:

$$\nabla^2 G_\sigma = \sigma^2 (G_{\sigma,xx} + G_{\sigma,yy}) \tag{6.6}$$

The Laplacian is normalized in amplitude by the term σ^2 in order to detect local maxima in the scale profile. When two images are zoomed, the ratio of intrinsic scales of the same feature

[1] also called characteristic scales

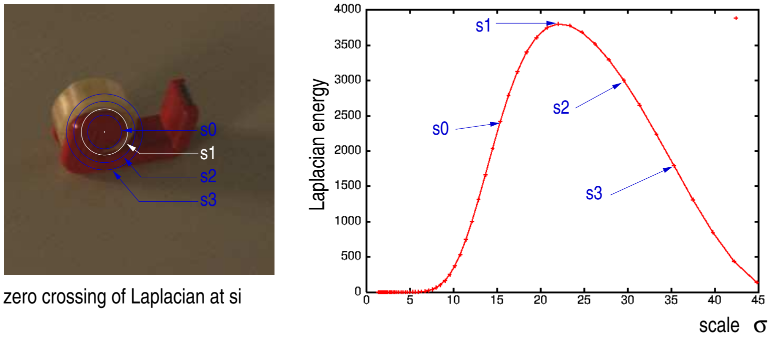

Figure 6.3: Scale profile of an image feature. The intrinsic scale is selected at the local maximum response to the normalized Laplacian energy [41].

in the two images is equal to the zoom ratio. Therefore the Laplacian operator is scale invariant. We have:

$$\begin{aligned} G_{xx,\sigma}(x,y) &= \frac{x^2-\sigma^2}{\sigma^4}G_\sigma(x,y) && (6.7)\\ G_{xx,t\sigma}(tx,ty) &= \frac{t^2x^2-t^2\sigma^2}{t^4\sigma^4}G_{t\sigma}(tx,ty) \\ G_{xx,t\sigma}(tx,ty) &= \frac{1}{t^3}G_{xx,\sigma}(x,y) && (6.8) \end{aligned}$$

We deduct:

$$\begin{aligned} \nabla^2 G_{t\sigma}(tx,ty) &= t^2\sigma^2(G_{t\sigma,xx}(tx,ty)+G_{t\sigma,yy}(tx,ty)) \\ \nabla^2 G_{t\sigma}(tx,ty) &= \frac{1}{t}\nabla^2 G_\sigma(x,y) && (6.9) \end{aligned}$$

Each pixel (x, y) allows at least one value for $\sigma(x, y)$ for which the response to the Laplacian is maximum. However, some pixels can be part of a surimposed feature and can allow two or three local maxima to the normalized Laplacian. We select the smallest of these maxima as a characteristic scale $\sigma_{opt}(x, y)$ for description of the appearance of a face at the pixel (x, y), because such features to describe appearance based on facial structure rather than illumination artifacts. In other domains it can be appropriate to use all of the maxima.

The scale profile can only be computed at a finite range of scales. The denser the sampling of scales, the higher the probability to find a precise value for the intrinsic scale, but the more computationaly expensive it also is. The sampling scales increase geometrically according to $\sigma_{r+1} = (1 + \epsilon)\sigma_r$. We choose $\epsilon = 0.1$, in order to make two consecutive scales grow by

10%. The initial value σ_0 is equal to 0.5 pixels in order to cover a neighbourhood of 1 pixel in diameter. An alternative is cubic interpolation used as pyramids in [19]. Tested scales have therefore the following form:

$$\sigma_r = \sigma_0(1+\epsilon)^r \tag{6.10}$$

6.3 Face image description

Scale invariant receptive fields are obtained by projecting image neighbourhoods of pixels on Gaussian receptive fields vectors normalized with their intrinsic scales. Regions centred on every pixel of the face image are therefore analyzed at an appropriate scale. We describe face images and their salient regions using low dimensionial feature vectors.

6.3.1 Projection into feature space

For each direction k, we compute the corresponding Gaussian receptive field vector at every scale $\sigma r(x,y)$. The normalization of face image into an imagette allows us to reduce the range in which the intrinsic scale is searched [38]. The scale map σ_{opt} of the face image is obtained by computing the intrinsic scale for every pixel of the image. The scale map of a face image is illustrated on image 6.4. For each direction k and pixel (x,y), we obtain a set of responses $(L_{k,\sigma 0}(x,y), L_{k,\sigma 1}(x,y), ..., L_{k,\sigma n}(x,y))$.

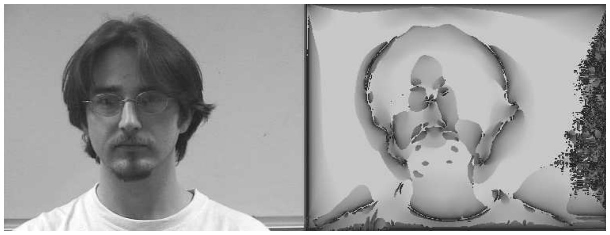

Figure 6.4: Scale map of the face image. Small scales are represented by dark pixels and large scales are represented by light pixels.

By selecting the intrinsic scale in the scale map, we obtain for each pixel the feature vector $L_{k,\sigma_{opt(x,y)}}(x,y)$ invariant to scale changes. The set of intrinsic feature vectors of the whole image in all directions is denoted L_{opt}. An example of face image response to Gaussian receptive fields normalized at intrinsic scales is shown on Figure 6.5.

The Gaussian receptive field reponse vector can be projected into the feature space. The feature space formed by 5 dimensional response vectors to Gaussian receptive fields is dense [41]. Example of cloud points of facial images are shown on figure 6.6. Two neighbourhoods with the same appearance are close in the feature space. To measure the similarity in appearance

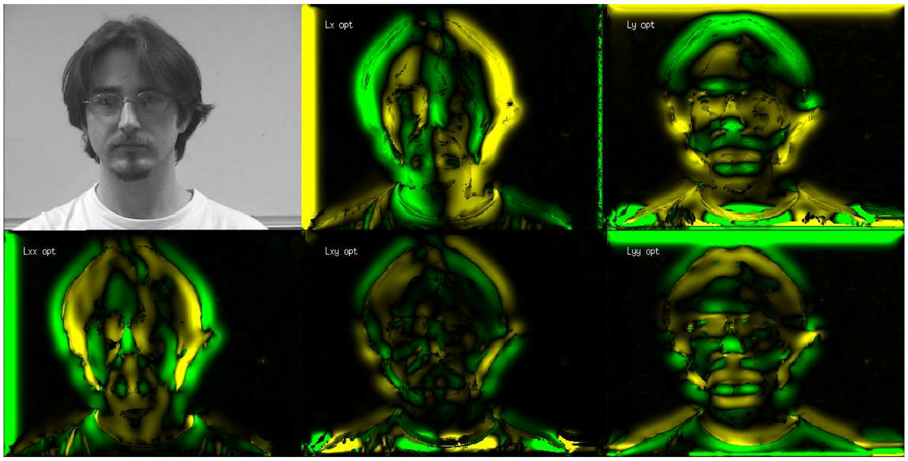

Figure 6.5: From left top to right bottom: the original image, $L_{x,\sigma opt}$, $L_{y,\sigma opt}$, $L_{xx,\sigma opt}$, $L_{xy,\sigma opt}$, $L_{yy,\sigma opt}$

between neighbourhoods of pixels, we compute the covariance normalized distance of their feature vectors, also known as the Mahalonabis distance. Given two vectors X and Y in a feature space, the Mahalonobis distance between X and Y is given by the following formula:

$$d_M(X,Y) = \sqrt{(X-Y)^T C^{-1}(X-Y)} \tag{6.11}$$

where C is the covariance matrix of the cloud points formed by the feature vectors of the image. The Mahalonobis distance takes correlations between variables of different dimensions into account and is more stable than the euclidian distance to describe similarities in multidimensional spaces. The covariance matrix represents axes of the response vectors distribution in the feature space and reflects existing correlations. We will use this distance to determine interesting features on facial images.

6.3.2 Salient facial feature regions

Our objective is to design local descriptors that are robust to changes in to scale, illumination and position to detect salient features in facial images in order to estimate their poses. Determining such local feature points can be performed by partitioning the face image into several regions, using textons as in [89] or finding generic features [93, 118, 79]. Facial features detection can also be performed using eigenfeatures [149], blobs [50] or saddle points and maxima of the luminance distribution [107]. However, such descriptors are sensitive to illumination and provide too many points, which can lead to accumulation errors. Natural interest points defined by Lindeberg [78] are not robust to pose, and are not apppropriate for deformable objects such as the human face, as they describe circular structures and the shape of a structure changes from a pose to another.

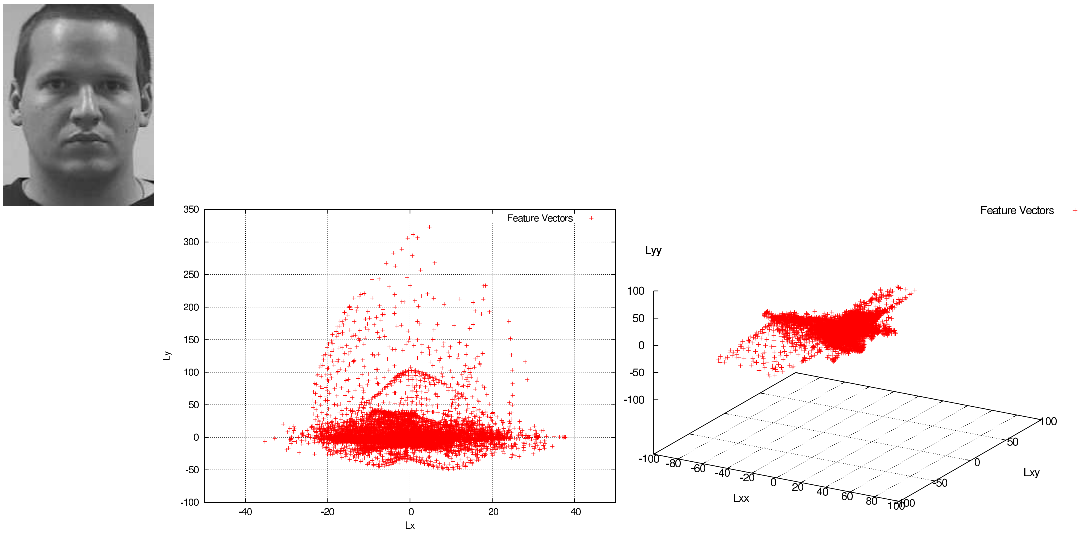

Figure 6.6: Original image and projection of its feature vectors into the feature space.

By considering the notion of saliency in the literature, we have found two definitions. An intuitive definition of salient features are features that draw attention. A mathematical definition is given by Walker et al. in [153] as features isolated in a dense feature space. We saw in the previous section that the feature space formed by Gaussian receptive fields response vectors was dense. However, isolated features may be difficult to determine. Results may depend on clustering algorithms and their parameter used to segment response vectors in the feature space. Most feature points are not an assembly of cloud points, but are composed of one block, which makes them difficult to partition with clustering algorithms. Furthermore, features can be isolated in a feature space without being isolated in the image. Salient features must only cover small regions on the image, otherwise they are not salient. Isolated points may just be outliers.

We propose the following definition for salient regions: A region is salient on an image when its neighbouring pixels share a similar appearance only over a limited radius. When the radius of the neighbourhood is too large, the region is too large and is not salient. When the radius is too small, the region is considered as spurious. There are two parameters in this definition: the size of salient regions δ and the similarity threshold d_S. Two neighbourhoods of pixels are considered different in appearance when their Mahalanobis distance exceeds this threshold.

By considering a pixel (x, y), we compute the 5-dimension normalized receptive field vector response $F(x, y) = L_{\sigma_{opt}(x,y)}(x, y)$ as well as for its neighbours. The pixel (x, y) is chosen as the reference vector. We compute the Mahalanobis distance $d_M(F(x, y), F(x + \iota_x \delta x, y + \iota_y \delta y))$ between the pixel and its neighbours in the eight cardinal directions, as presented on Figure 6.8. Variables (ι_x, ι_y) can have the values $\{-1, 0, 1\}$. If the eight distances are superior to the similarity threshold d_S, the pixel (x, y) is considered as part of a salient region. If most distances are inferior to the threshold, the pixel can either be part of a large region sharing the same appearance, or be a spurious region. When only one or two distances do not exceed the threshold, the pixel can be part of a ridge or a interest line on the image. Appearance similarities of differented facial regions are shown on Figure 6.7. Possible Mahalanobis distances profiles in one direction are presented on Figure 6.9. The effets of varying the parameters are shown on figure 6.11. The condition of saliency of a pixel is summed up below:

$$\forall(\iota_x, \iota_y) \in \{-1, 0, 1\}^2 - (0, 0) \qquad d_M(F(x, y), F(x + \iota\delta x, y + \iota\delta y)) > d_S \tag{6.12}$$

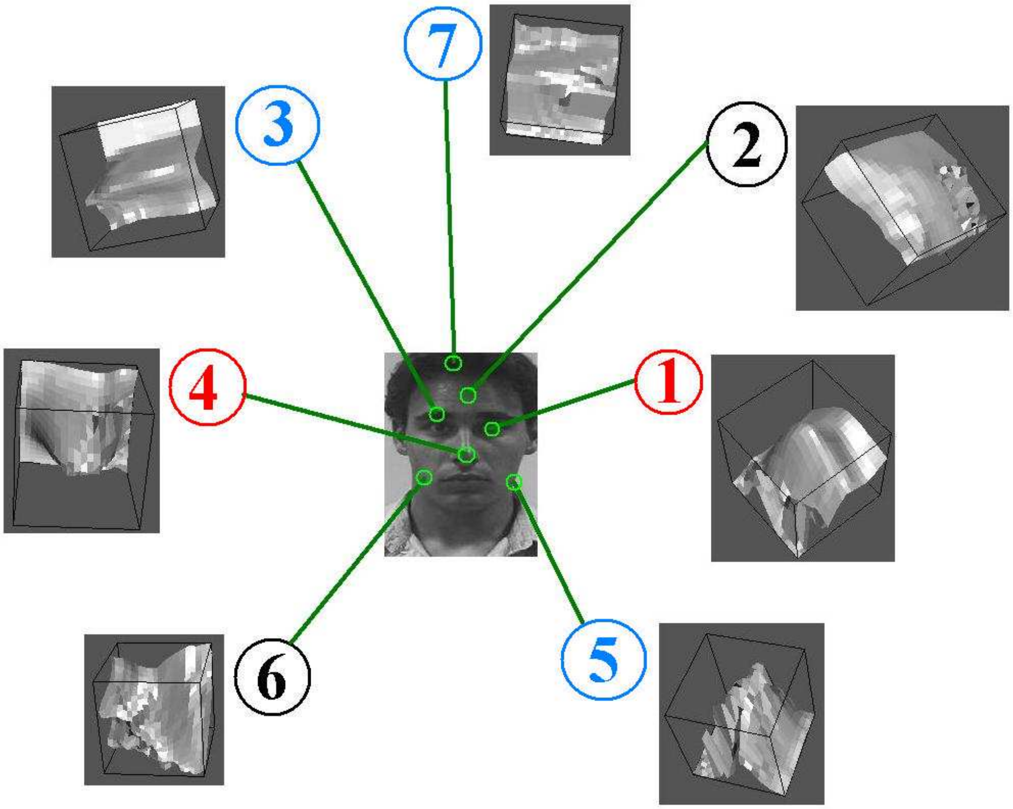

Figure 6.7: Appearance similarities of different facial neighbourhoods: (1) Eye, (2) Forehead, (3) Eyebrow, (4) Nose, (5) Face contour, (6) Cheek, (7) Hair. Regions (1) and (4) appear as blobs and are considered as salient, regions (3) and (5) appear as ridges on the image, other regions do not exhibit such structures and are not considered as salient.

We use a similarity threshold of $d_S = 1$ and a size of $\delta = 10$ pixels for salient region detection on face images. The performance of our detector on face images is compared to other detectors on Figure 6.10. Normalized Gaussian recpetive fields give good results and feature detection appear to be robust to pose and identity.

We found that the salient facial features detected by normalized Gaussian receptive fields correspond to regions covering the eyes, nose, mouth and face contour. These results ressemble

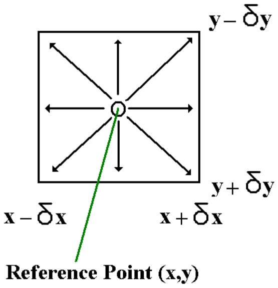

Figure 6.8: Distance profiles of size δ are calculated in the cardinal eight directions from the reference point.

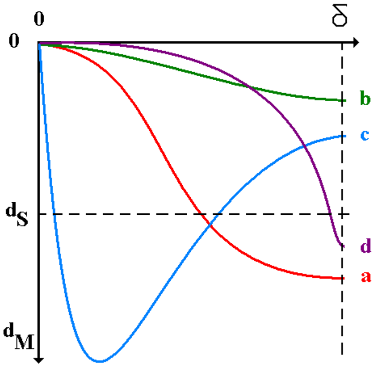

Figure 6.9: Different Mahalanobis distance profiles: (a) Salient region, (b) Region too large in appearance to be salient, (c) Spurious region, (d) Salient region near the maximum size.

those obtained by the studies of psychophysician Yarbus. As shown on Figure 6.12, humans tend to analyse these regions when recognizing people.

Salient facial feature detection and description is efficient using Gaussian receptive fields normalized in scale. Furthermore, the position of the salient features with regard to the position of the face could be a good cue for head pose estimation. We build a structure based on these salient features to refine the coarse estimation obtained in the previous chapter.

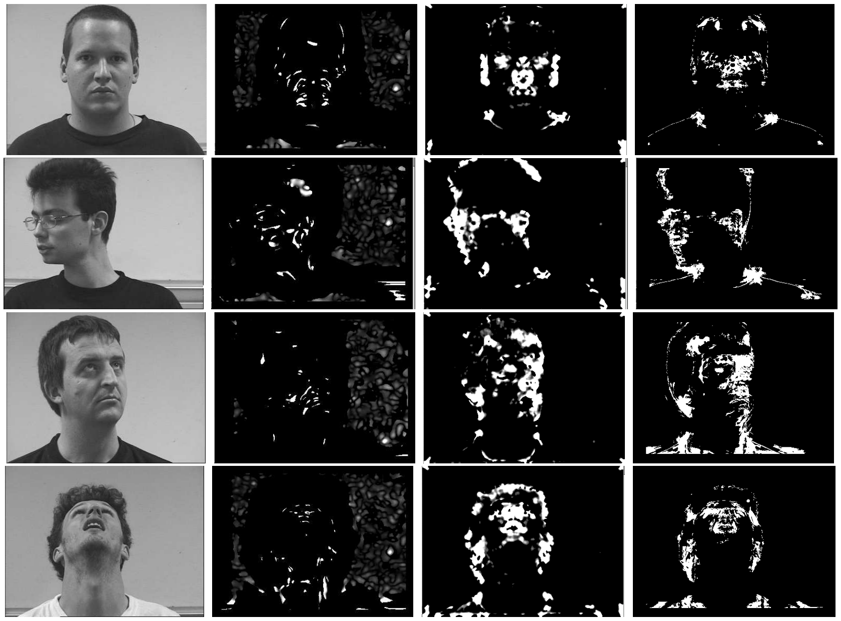

Figure 6.10: Examples of saliency maps obtained. From left to right: Original 1/4 PAL image, Lindeberg natural interest points with a scale of 5 pixels, Harris points [45], Salient feature detection using normalized Gaussian receptive fields.

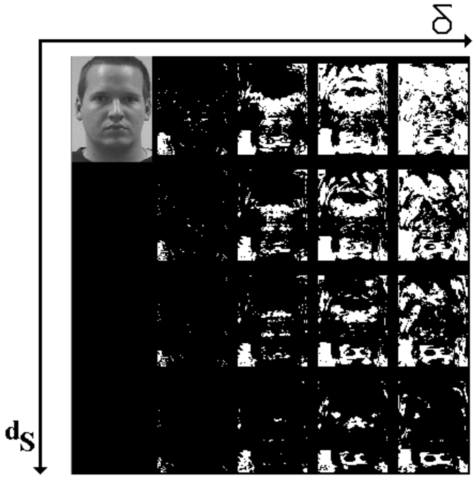

Figure 6.11: Salient facial feature detection by varying the size δ of salient regions and the similarity threshold d_S.

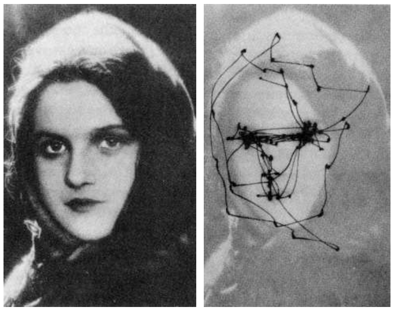

Figure 6.12: The left image is a photography presented to a subject. The right image describes the path followed by the eye gaze of the subject. Eyes, nose, mouth and face contour are the most examinated facial parts [165].

Chapter 7

Salient Gaussian Receptive Field Graphs

This chapter explains the use of salient Gaussian receptive field grid graphs for head pose estimation. This structure has interesting properties for image matching under changing conditions, as its describes both geometrical and textural information present in the image. The first part of this chapter describes node displacement algorithm according to its saliency and its representation by hierarchical clustering of low dimensionality vectors. Head pose computation from salient grid graphs is developped in the second part. We refine the estimate obtained in Chapter 5 by searching for the most similar salient grid graph from its neighbourhing poses. The last two parts of the chapter are dedicated to final results, comparison with human performance and discussion.

7.1 Grid graph structure

The relative position of robust salient facial features found in the previous section with regard to the head may provide useful information about its orientation. However, direct pose estimation from these feature is rendered difficult because of:

- Feature location variation due to changes in identity
- Feature appearance changes due to changes in identity
- Feature location variation due to imperfect alignment of imagettes

To handle these problems, we adapt the "elastic bunch graphs" method proposed by Von der Malsburg et al. [158] to form Gaussian receptive field graphs. This method provides interesting properties for image matching under changing viewing conditions.

Elastic bunch graphs were initially developed for face recognition. A graph G is described as a set of N nodes n_j labelled by their descriptors X_j. In the literature, Gabor Wavelets play the role of such descriptors. They describe both geometrical and textural information in the image. Description with Gaussian derivatives provides information similar to Gabor wavelets at a much lower computational cost.

Head pose estimation has been performed on a varying number of poses using elastic bunch graphs [24, 90, 71, 160]. Nevertheless, such systems require high resolution of the face image. Furthermore, graphs are constructed empirically for each pose. We do not know if the choice of the facial points and of theirs egdes is relevant for head orientation estimation. Training a new person or a new pose requires manually labeling graph nodes and edges on all his face images. As we do not want to use manual annotation in our system, we use graphs whose nodes and egdes are regularly distributed to recover head pose from facial features. Such graphs are called grid graphs.

The graph structure describes both local appearance and the geometric relation of regions in the image. We use the 5 dimmensionnal response vectors composed of first and second order Gaussian receptive fields normalized at intrinsic scales described in the previous chapter as node descriptors. We extend the grid graph structures used in [42] by describing each node n_j by its relative location (x, y) in the face image and a 5 dimensional vector $L_{\sigma opt(x,y)}(x, y)$. The model graph structure takes appearance changes of features due to identity into account by gathering Gaussian receptive fields response vectors on each node. However, although elastic graphs can handle small changes in head movement, they have difficulties with large changes in head orientations [76]. We compute a model graph for each pose $Pose_i$. Each node n_j is labelled by a set of M vectors $\{X_{jk}\}$, where M is the number of images with the head pose $Pose_i$ in the training data. This set of vectors describes possible appearances of the facial feature found at the location (x, y) of the node n_j. The transformation from grid graphs to model graph is shown on Figure 7.1. The model graph structure describes possible variations in location and appearance of facial features for a particular head pose. We extend model graphs to salient grid graphs by allowing local nodes displacements.

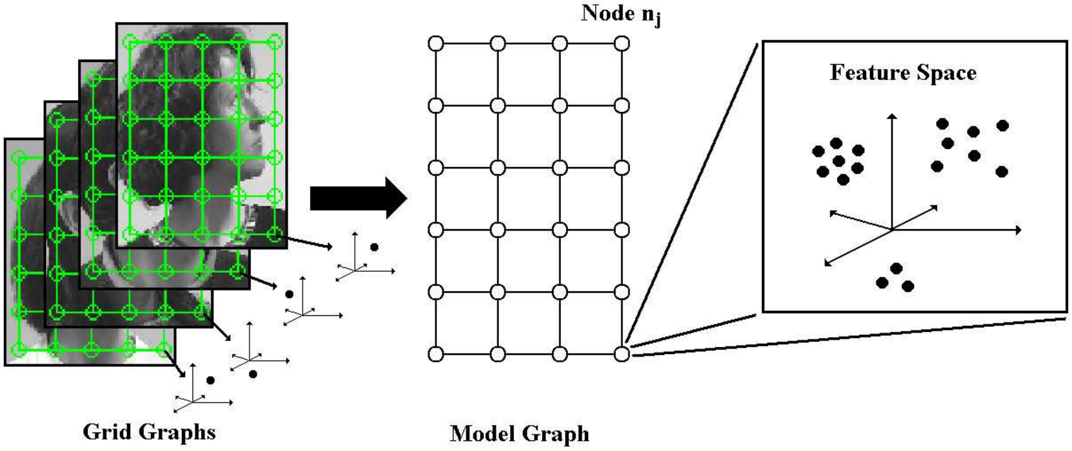

Figure 7.1: Transposition of grid graphs applied to face images of the same head pose to pose model graph.

7.1.1 Node displacement

To handle varations of positions of facial features on the image due to identity changes and imperfect alignment, the model graph can be distorted locally during matching by searching for the most similar label of each node within a small window, as proposed in [109]. The size of the window must not exceed the distance ld_{max} between the nodes, to preserve the order of nodes and to maintain their neighbourhing relations. An example of local displacement on a grid graph is presented in Figure 7.2. The distance between the nodes should be small enough so as to cover relevant facial features and discriminate consecutive head poses.

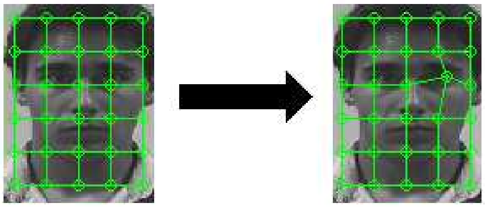

Figure 7.2: Example of local displacement of a node.

Gaussian receptive fields grid graphs are the intuitive extension of salient facial regions developed in the previous section. A region of an image is salient when its neighbouring pixels share a similar appearance only over a limited radius δ. The local displacement of each node of the graph corresponds to the radius δ. The feature located at a certain pixel must be similar only to features located at neighbourhing pixels. We propose to define the maximal displacement of a graph node with regard to its saliency. Salient facial regions can be detected on single images. By computing the sum of salient facial regions of images of the same head pose normalized by the number of images, we obtain a saliency map for each pose, as explicited on Figure 7.3.

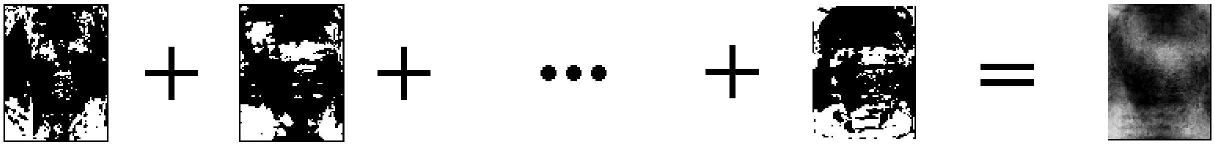

Figure 7.3: Example of salient facial regions detected on single images and their combination to a saliency map of a near frontal head pose. Dark pixel values represent non salient facial regions and light pixel values represent salient facial regions

The pose saliency map gives a direct relation between a pixel (x, y) and its salency $S(x, y)$ comprised between 0 and 1. The more a pixel is salient, the more relevant its location is for the considered pose. By denoting ld_{max} the distance between 2 nodes and (x_j, y_j) the location of the node n_j, we define the maximal local displacement $ld(n_j)$ of the node n_j as follows:

$$ld(n_j) = (1 - S(x_j, y_j)) \cdot ld_{max} \tag{7.1}$$

The rigidity of a node becomes proportional to its saliency. A node placed at a salient fixation represents something relevant for the considered pose and does not need to move too much from its original location. On the other hand, a node placed at a non-salient location does not represent a relevant feature and can be moved with a maximal displacement equal to the distance between 2 nodes, in order to preserve geometric relation. An example of the local displacement of a node based on saliency is shown on Figure 7.4. We refer to such graphs as salient grid graphs. In the next part of this chapter, we explain how to model different features located in the same region.

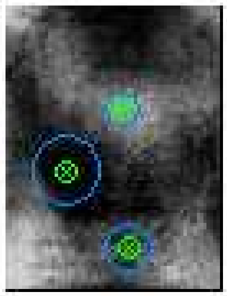

Figure 7.4: Nodes' local displacement according to their saliency. Nodes with little saliency can move with a maximal displacement whereas nodes with high saliency have limited displacement.

7.1.2 Node representation by Hierarchical Clustering

The same facial point can have different aspects with regard to a person. For example, although they can be expected to be roughly found at the same location on the face, eyebrows can have different appearance. They generally tend to be wide for men, and discrete for women. The result is an assembly of clouds of points in the feature space for each node n_j of the graph. To model such different aspects of the same feature, we apply a hierarchical clustering technique to the receptive fields vectors of the same node.

The hierarchical clustering algorithm [60] presents an interesting alternative to other clustering algorithms such as K-Means and EM. The main advantage is that the number of clusters, K, does not need to be arbitrarly choosen, and there are no centroids to initialize. Instead, a series of cluster fusions takes place, which run from n clusters, each containing a single point, to a single cluster containing all of the points. At each step of the algorithm, the method joins the two closest clusters in the feature space together. The distance between two clusters A and B is calculated with the average group linkage method. This is defined by computing the mean distance between all points of the merged cluster $A \cup B$, as shown on Figure 7.5. The average group distance is computed as follow:

$$d(A,B) = \frac{1}{Card(A) + Card(B)} \sum_{i \neq j} d_M(X_i, X_j), \qquad X_i \in A \cup B \tag{7.2}$$

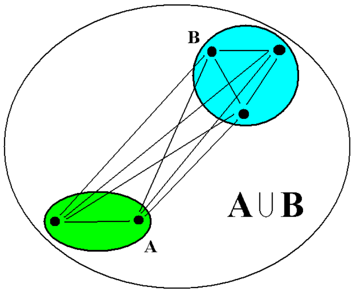

Figure 7.5: Two clusters and their group distances.

The two clusters A and B are merged in such a way that the average pairwise Mahalanobis distance within the newly formed cluster is minimum. The average group linkage method minimizes the information loss associated with each grouping. During each iteration, the union of every possible cluster pair is considered and the two clusters whose fusion results in minimum increase in information loss are merged. The information loss of a partition P of an assembly of points $\{X_i\}$ is defined in terms of a sum of squared criterion distance:

$$Loss(P) = \sum_{i=1}^{K} \sum_{j=1}^{Card(A_i)} d_M(\mu_i, X_{ij})^2, \qquad X_{ij} \in A_i \tag{7.3}$$

Where K represents the number of clusters in the partition P, X_{ij} points of the cluster A_i and μ_i the mean of the cluster A_i. The lower the information loss is, the better the data is represented by the partition. Each cluster A_i is represented by its mean vector μ_i and its covariance matrix C_i.

The convergence criterion of the algorithm can depend on two parameters: the minimum information loss err and the computed distances factor κ. Hierarchical clustering can stop when the information loss goes below the value err. However, depending on the data, this minimal value can sometimes simply not be reached, and the result of the algorithm is a single cluster gathering all points. The factor κ can be used to limit the number of iteration steps in the algorithm. The total number of computed distances between n points is $\frac{n(n-1)}{2}$. Instead of using all distances, the method considers only the κn lower distances. The factor κ must therefore be inferior to $\frac{n-1}{2}$. The hierarchical clustering procedure is summarized below:

```
   Hierarchical Clustering

0. Compute $\kappa n$ distances between the $n$ points and sort them
1. Merge the two clusters whose distance is minimal
2. Update cluster distances
3. Repeat steps 1 and 2 until convergence criterion is reached
```

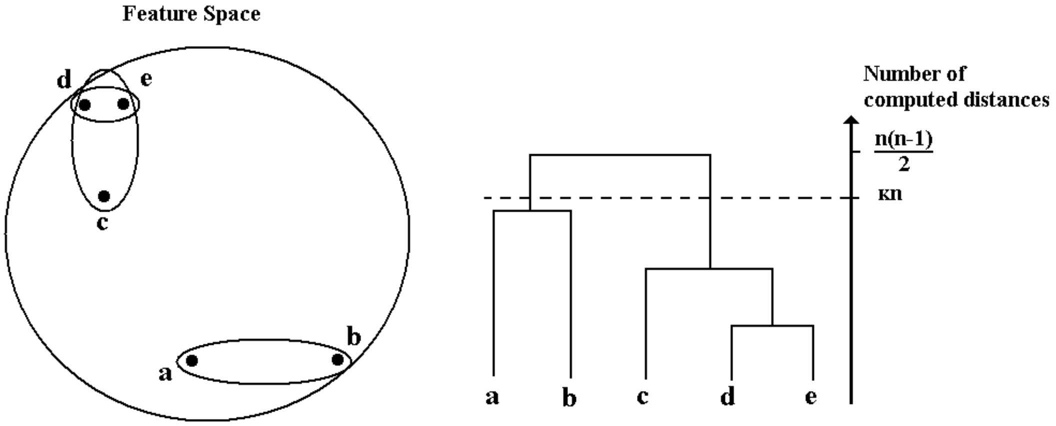

Figure 7.6: Example of hierarchical clustering in a feature space and its dendogram representation. The height of the dendogram stands for the number of iteration steps and the number of computed distances. The higher a newly cluster is formed, the more relevant it is. By limiting the number of computed distances to κn, we obtain a good representation of the data.

A hierarchy of clusters can be represented by a dendogram, as shown on Figure 7.6. In our experiments, we use a minimum information loss of $err = 0.5$ and a computed distance factor of $\kappa = 2.5$. The result of the clustering is a set of K mean vectors and covariance matrix $\{\mu_i, C_i\}$ modelling the changing aspects of features found on the same facial point on different persons. We now have a reliable representation of the appearance changes of faces at every node of the graph.

7.2 Coarse-to-Fine head pose estimation

A salient grid graph is represented by a set of N nodes $\{n_j\}$ that allow a local displacement around their origin. Each node is labelled by a set of K_j clusters $\{A_{jk}\}$ represented by their mean vectors and covariance matrix $\{\mu_{jk}, C_{jk}\}$ and can therefore be considered as a probability density function. During the graph matching, we evaluate the probability $p(Pose_i)$ that the pose of the tested face image is $Pose_i$. Given the law of total probabilities, we have:

$$\begin{aligned} p(Pose_i) &= \sum_{j=1}^{N} p(Pose_i|n_j)p(n_j) \\ p(Pose_i) &= \frac{1}{N}\sum_{j=1}^{N} p(Pose_i|n_j) \end{aligned} \tag{7.4}$$

As the probability $p(n_j)$ for a node to occur is $\frac{1}{N}$. Using Bayes' rule, we obtain for each node:

$$\begin{aligned} p(Pose_i|n_j) &= \frac{p(n_j|Pose_i)p(Pose_i)}{p(n_j)} \\ p(Pose_i|n_j) &= \frac{N}{N_P}p(n_j|Pose_i) \end{aligned} \tag{7.5}$$

By defining the number of possible poses as N_P. Again, the law of total probabilities applied on the K_j clusters $\{A_{jk}\}$ of a node n_j gives:

$$p(n_j|Pose_i) = \sum_{k=1}^{K_j} p(n_j|Pose_i, A_{jk})p(A_{jk}) \tag{7.6}$$

This probability will provide the best location for the node n_j on the tested image. We denote as $X_j(x,y)$ the optimal normalized Gaussian receptive field vector response computed at this node. The prior $p(A_{jk})$ corresponds to the frequency of cluster A_{jk} and is therefore equal to $\frac{1}{Card(A_{jk})}$. The probability of $X_j(x,y)$ to belong to cluster A_{jk} is modeled by a 5 dimensional Gaussian function of mean and covariance (μ_{jk}, C_{jk}). An example of probability density function at a graph node is shown on Figure 7.7. We deduce:

$$p(n_j|Pose_i) = \sum_{k=1}^{K_j} \frac{1}{Card(A_{jk})} \frac{1}{(\sqrt{2\pi}det(C_{jk}))^5} e^{-\frac{1}{2}(X_j(x,y)-\mu_{jk})^T C_{jk}^{-1}(X_j(x,y)-\mu_{jk})} \tag{7.7}$$

The location (x,y) which obtains the highest probablity is selected as the optimal location for the node n_j. The correpending response vector will be denoted X_j. We obtain the complete probability that the face image has a head pose $Pose_i$:

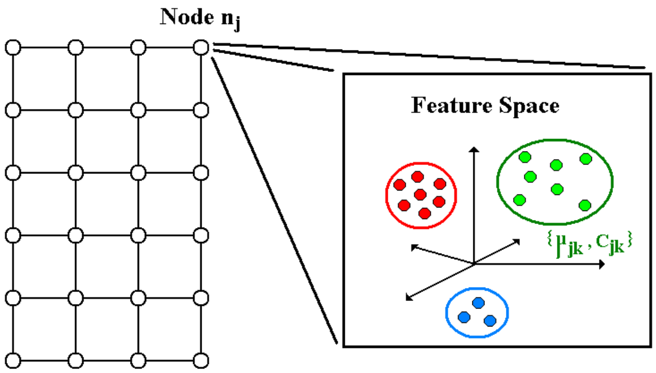

Figure 7.7: Example of salient grid graph and a probability density function at one node.

$$p(Pose_i) = \frac{1}{N_P}\sum_{j=1}^{N}\sum_{k=1}^{K_j}\frac{1}{Card(A_{jk})}\frac{1}{(\sqrt{2\pi}det(C_{jk}))^5}e^{-\frac{1}{2}(X_j-\mu_{jk})^T C_{jk}^{-1}(X_j-\mu_{jk})} \quad (7.8)$$

The pose i whose probability gives the best score is selected as the head pose. The number of nodes N is inferior to the size S of the image. The complexity of Gaussian receptive graphs is therefore linear.

7.3 Performance

The head pose estimation system based on linear auto-associative memories described in Chapter 5 delivers a coarse estimate for the pose. We use separated training of pose prototypes to enhance computation time. The obtained result can be refined by searching the most similar graph from among neighbourhing poses, as illustrated on Figure 7.8. For this experiment, we used graphs composed of 12x15 nodes. Performance evaluation can be seen in Table 7.1. We tested different types of graphs to evaluate our method:

- **LAAM**
  Linear Auto-Associative Memories learned separately as defined in Chapter 5.

- **Salient Grid Graphs**
  Grid Graphs defined in this chapter.

- **1-Clustered Grid Graphs**
  Grid Graphs where nodes' appearance is not clustered hierarchicaly, but represented only by 1 cluster.

- **Oriented Grid Graphs**
  Grid Graphs located only on the region of the face supposed to contain salient features. Examples of Oriented Grid Graphs can be seen on Figure 7.9.

- **Fixed Grid Graphs**
  Grid Graphs where nodes cannot move. This corresponds to the situation where every point on the image is salient.

- **Naive Grid Graphs**
  Grid Graphs where nodes can move with maximal displacement. This corresponds to the situation where no point on the image is salient.

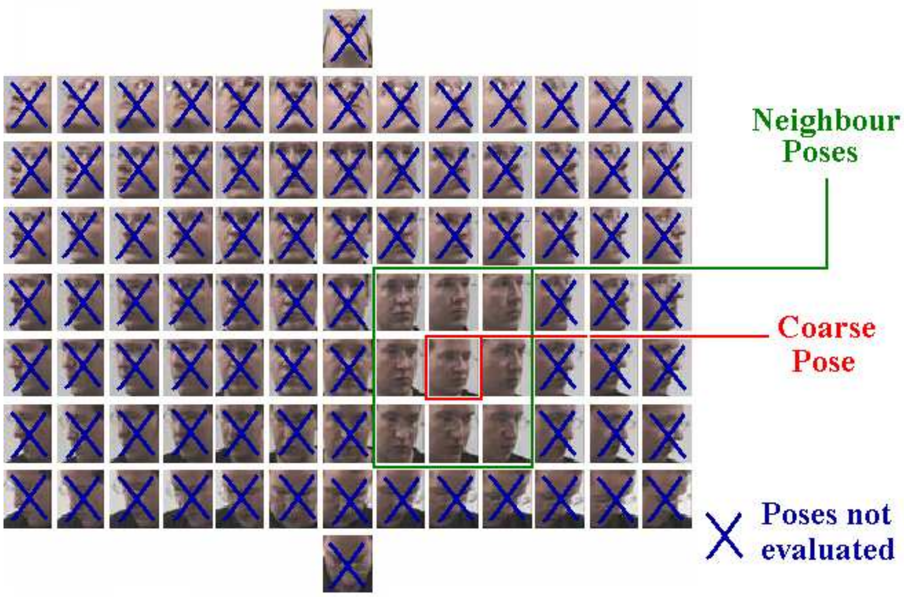

Figure 7.8: Example of neighbour head poses. Other poses are not considered.

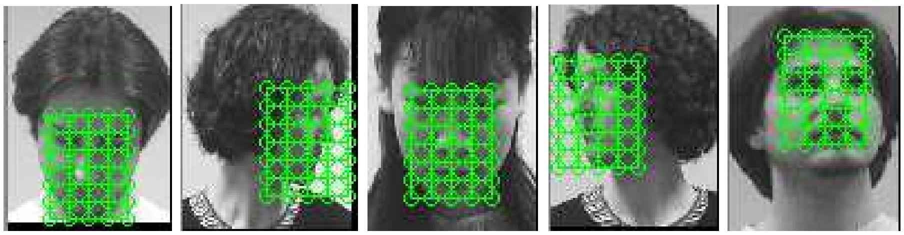

Figure 7.9: Example of oriented grid graphs. Graph centers are calculated with regard to head pose

Method	Pan Error	Tilt Error	Pan Class.	Tilt Class.	Pan Class. 15^o
Salient GG	16.2^o	16.2^o	40.6 %	46.2 %	70.8 %
LAAM	10.1^o	15.9^o	50.3 %	43.9 %	88.8 %
LAAM + 1-Clustered GG	11.5^o	13.5^o	44.7 %	45.9 %	80.1 %
LAAM + Oriented GG	10.8^o	13.5^o	46.8 %	44.8 %	82.1 %
LAAM + Fixed GG	12.7^o	14.9^o	47.1 %	47 %	86.6 %
LAAM + Naive GG	12.2^o	13.5^o	50.4 %	50.4 %	86.9 %
LAAM + Salient GG	**10.1^o**	**12.6^o**	**50.4 %**	**47.3 %**	**88.8 %**

Table 7.1: *Performance evaluation on unknown users with different types of graphs. LAAM and GG refers respectively to Linear Auto-Associative Memories and Grid Graphs. Resolution of images is 75x100 pixels.*

The use of salient grid graphs combined with linear auto-associative memories provides the best results and improves the coarse estimation of head pose. Tilt angle estimation is the most improved. Coarse-to-Fine head pose estimation results can be seen in Table 7.2. Pan and tilt error per pose can be seen on Figure 7.10. This result shows that the combination of the two methods works better than using only any one method individually. When combined, LAAM and Salient Grid Graphs work as a coarse-to-fine process in the sense that a coarse pose estimate is used to initialise a local search for more precise pose.

Evaluation Measure	LAAM	LAAM + SGG
Pan Average Error	10.08^o	10.07^o
Tilt Average Error	15.9^o	12.6^o
Pan Classification 0^o	50.3 %	50.4 %
Tilt Classification 0^o	43.9 %	47.3 %
Pan Classification 15^o	88.8 %	88.8 %

Table 7.2: *Coarse-to-Fine Head Pose Estimation performance. LAAM and SGG refer respectively to Linear Auto-Associative Memories and Salient Grid Graphs*

Salient grid graphs perform better when using linear auto-associative memories as a prior classification step. On the one hand, memories are appropriate for delivering a coarse estimation of the head pose by recognizing global appearance of the face on an imagette. This coarse estimation then allows the salient grid graph matching to be restricted to neighbouring poses, which reduces computational time. Instead of browsing 93 salient grid graphs, no more than 9 salient grid graphs are tested to produce a precise estimate in pose.

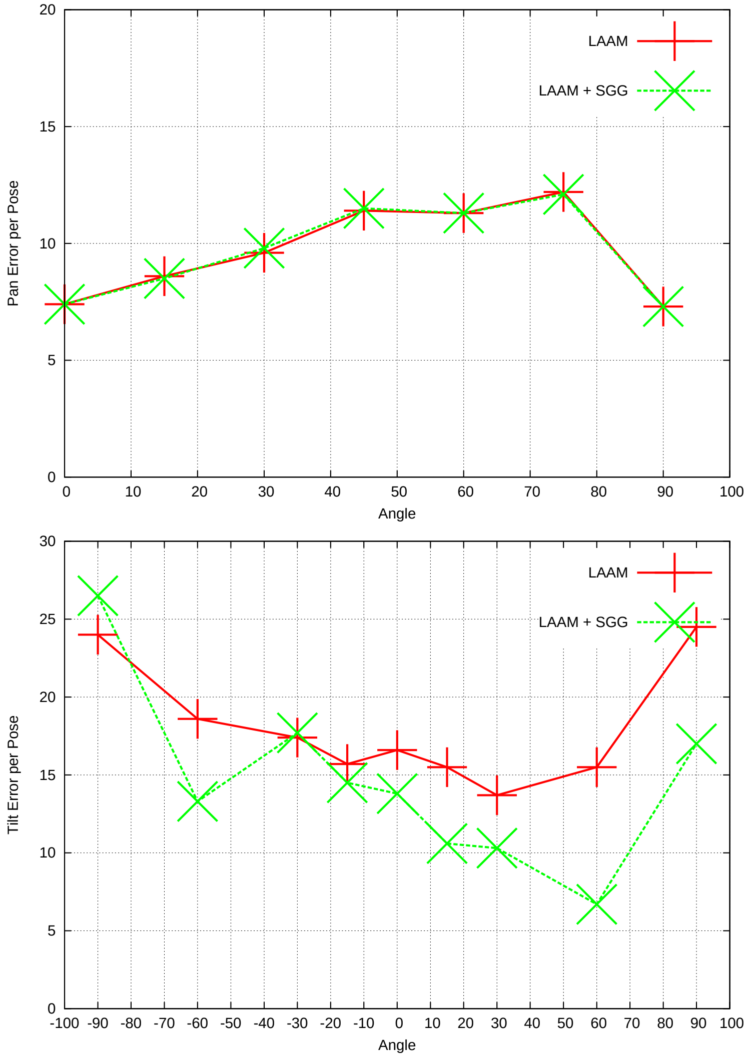

Figure 7.10: Average error per pose on pan and tilt axis

Salient grid graphs perform better than 1-clustered grid graphs. This demonstrates the utility of modeling changing aspects of facial features located at the same place. Hierarchical clustering is an efficient and simple method to obtain a reliable representation of appearance variations of facial features due to identity.

Salient grid graphs perform better than oriented grid graphs. This result shows that the larger the region covered by the graph, the better the discrimination between neighbour poses. By placing the grid graph on only a certain region, the local displacement of nodes can degrade pose classification by replacing the nodes at a neighbor pose, which degrades the final classification result. Covering the whole face image region makes it possible to maintain geometric relations between a certain face region and adjacent regions, which reduces misclassification between neighboring poses.

Salient grid graphs perform better than fixed grid graphs. This demonstrates the importance of local displacement of graph nodes. This displacement is useful to handle feature location variation in the face image due to identity changes and imperfect alignment.

Salient grid graphs perform better than naive grid graphs. This result shows that by limiting the local displacement of graph nodes with regard to their saliency, the matching and discrimination of head poses is enhanced. Furthermore, as the local displacement of nodes are limited, salient grid graph matching is faster than naive grid graph matching. Non-salient facial region location variations are larger than salient facial regions locations, which make salient region more relevant for head pose determination.

7.4 Comparison with human performance

We have compared the performance of our coarse-to-fine system, linear auto-associative memories combined with salient grid graphs, with human performance on unknown faces, as described in section 6.4.3. From these tables we can see that our method achieves accuracy similar to human abilities. Results are shown in Table 7.3. Average error per pose is illustrated on Figure 7.11.

Evaluation Measure	Calibrated Subjects	Non-Calibrated Subjects	C-t-F HPS U
Pan Average Error	11.8^o	11.9^o	10.1^o
Tilt Average Error	9.4^o	12.6^o	12.6^o
Pan Classification 0^o	40.7 %	42.4 %	50.3 %
Tilt Classification 0^o	59 %	48 %	47.2 %

Table 7.3: *Performance comparison between humans and our system. C-t-F HPS U refers to Coarse to Fine head pose estimation system on Unknown users. Calibrated and Non-Calibrated are defined in Chapter 3.*

With an average error of 10.1 degrees and a correct classification rate of 50.4%, our method performs significantly better than humans at estimating pan angle, with an average error of 11.9 degrees. The standard deviation of the average error per pose is low for the system and high for humans. The system achieves roughly the same precision for front and profile, and higher precision for intermediate poses. As for humans, minimal error can be found at front and profile poses.

With an average error of 12.6 degrees in tilt, our method achieves a performance comparable with humans'. The worst tilt angle estimations were obtained at extreme poses: +90 and -90 degrees. The reason is that not every subject in the database was able to raise his head up and down exactly at -90 and +90 degrees. This is due to the variety of shapes of the face and the

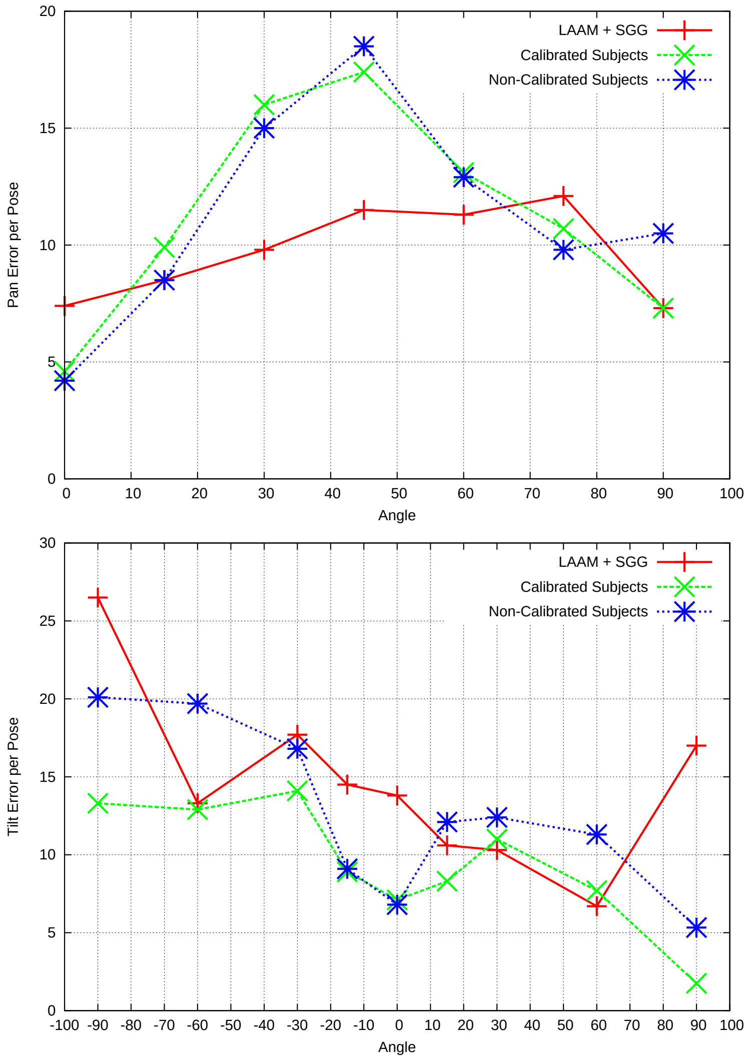

Figure 7.11: Average error per pose on pan and tilt axis.

neck. Face region normalization also introduces a problem. The height of the neck differs from one person to another. This provides large variations on face imagettes and can disrupt tilt angle estimation.

Average error per pose obtained by our system is more homogeneous than the one obtained by humans. The coarse-to-fine approach performs better recognition on intermediate poses, but humans perform better at recognizing front and profile poses. While our algorithm may be confused with two neighbour front or profile poses, humans seem to have an ability to discriminate between extreme, neutral and other poses. This confirms the fact that front and profile poses are used as key poses by our brain.

This chapter has proposed a new coarse-to-fine method to estimate head pose on unconstrained images. Face images are normalized in scale and slant to provide an imagette by a robust face detector. Face imagettes containing the same head pose are learned through a linear auto-associative memory and a salient grid graph. Each node of the graph can be locally displaced according to its saliency on the image and is labelled by a probability density function of normalized Gaussian receptive field vectors clustered hierarchically. The coarse head pose estimation process uses the cosine angle of the source and reconstructed images. A simple winner-takes-all process is applied to select the head pose whose memory gives the best match. The refined estimation process consists in searching the best salient grid graph among the neighbour head poses found by the coarse estimation process.

Salient grid graphs improve the performance obtained by linear auto-associative memories on unknown users. The best improvement occurs for the tilt axis. Pan angle estimation is little improved, which is due to the fact that the pan information is contained in the horizontal asymmetry of the global appearance of the face image. As grid graphs have a linear complexity, and linear auto-associative memories have a quadratic complexity, grid graphs can take over from memories on higher resolution images. Example images are shown on Figure 7.12. Furthermore, Gaussian receptive fields are robust to illumination, which can provide a solution in cases where memories fail. We achieve a fully automatic algorithm for head pose estimation that uses both global and local appearances of low resolution unconstrained single images whose performance is comparable to human performance on known and unknown users. This method does not use any heuristics, manual annotation or prior knowledge on the face and can therefore be adapted to estimate the pose of configuration of other deformable objects or to recognize facial emotions.

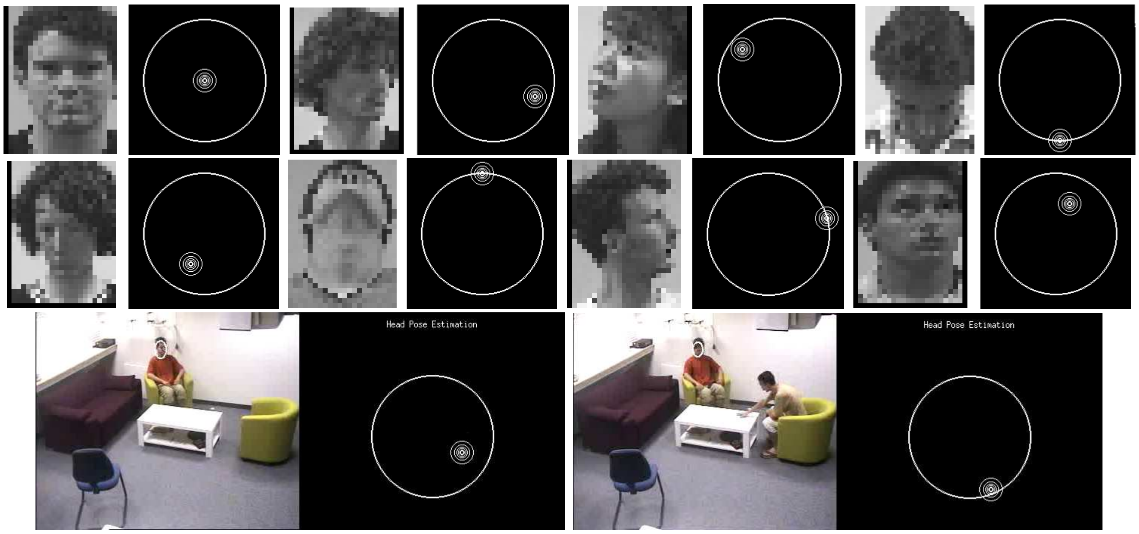

Figure 7.12: Example test imagettes of unknown subjects on the left and their pose representation on the right. A target located at the center of the circle indicates the frontal pose.

Head orientation is often used by humans to estimate visual focus of attention from single images. The pan angle is more relevant for their estimation than the tilt angle. In particular, front and profile poses are particularly well recognized. Abilities degrade for intermediate angles. We develop a new computer vision based system who can deliver performance comparable to human performance on the same data. Furthermore, our algorithm can provide a better discrimination of intermediate angles. Then, the results obtained by our coarse-to-fine approach are sufficiently good and well adpated for head orientation estimation in smart environments, in order to predict human interactions with objects and people.

Below is a summary of our refined pose estimation algorithm:

```
Training:

For each group of poses $k$:

    Initialize a Salient Grid Graph

    For each image $X_k \in k$:

        Compute its Gaussian Receptive Field response vectors $L_{\sigma opt}(X_k) = X_k \otimes G_{\sigma opt}$
        Compute its Saliency Map $S(X_k)$
        Collect its Gaussian Receptive Field response vectors at each graph node $n_j$

    Compute the average Saliency Map of $k$ : $S_k = \frac{1}{Card(k)} \sum_{X_k} S(X_k)$

    For each graph node $n_j$:

        Gather all responses $L_{\sigma opt}(X_k)$ in the feature space
        Do a hierarchical clustering on points formed by the responses $L_{\sigma opt}(X_k)$
```

```
Testing:

Given a test image $Y$,
Estimate its coarse pose $k_{coarse}$

For each group of poses $k$ neighbours to $k_{coarse}$:

    For each node $n_j$ of the Salient Grid Graph of $k$:

        Displace locally the node $n_j$ at the location $(x_j, y_j)$ with a maximal displacement
        oppositely proportional to its saliency: $ld(n_j) = (1 - S_k(x_j, y_j)) \cdot ld_{max}$
        Select the location with the highest probability $p(n_j|k)$ given $L_{\sigma opt}(Y)$

    Compute the score of the Salient Grid Graph of $k$: $\sum_j p(n_j|k)$

Select the class $k$ whose graph obtains the highest score: $k_{refined} = argmax_k(\sum_j p(n_j|k))$
The refined pose of the image $Y$ is $k_{refined}$
```

Chapter 8

Extensions

This chapter presents some extensions of our system. The first part details the use of linear auto-associative memories for people detection in video surveillance systems. Head pose estimation on video sequences is developed in the second part. The third part of the chapter extends the use of head orientation estimation to attentional systems.

8.1 Person Modelisation and Classification

This section presents an application of linear auto-associative memories to person and non-person classification. We propose a simple method working at low resolution that requires very few parameters. Furthermore, this approach inherits strong points of appearance based vision: simplicity and independence from the detection technique. We compare the performance of our system with three other statistical algorithms: a structural ridge-based method, using a set of main human components [144], normalized gradient histograms [118] and a modified version of the SIFT descriptor [81]. To assess the performance of the methods, we use the IST CAVIAR[1] database.

8.1.1 Related Work

A classic public video-surveillance system requires the ability to determine if an image region contains people. Object classification is more difficult because it must accomodate changes in imaging conditions. People detection is much harder due to the high variation of human appearance as well as the small size of human region which prevents face or hand recognition. Numerous efficient appearance-based approaches exist for object recognition [117, 41]. However, such techniques tend to be computationaly expensive.

Video-surveillance systems must run at video-rate and thus require a trade-off between precision and computational time. To speed up the classification, simpler methods have been pro-

[1] http://homepages.inf.ed.ac.uk/rbf/CAVIAR/caviar.htm

posed. In [44], the authors use only compactness measure computed on region of interest to classify car, animal or person. This measure is simple but sensitive to scale and affine transformations. Moreover, this method is highly dependant on segmentation, which remains a fundamental problem. In [5] and [166], the contour is used to model deformable shapes of a person. However, the person must be represented by a closed contour. All these methods strongly depend on contour detection or segmentation techniques.

Whereas local approaches such as ridge extraction use interesting properties of neighboorhoods of pixels, global approaches use the entire appearance of the region of interest. Principal advantages of such approaches are that no landmarks, or model need be computed, only the objects must be detected. Global approaches can also handle very low resolution. A popular method for template matching is PCA, but this approach tends to be sensitive to alignment and the number of dimensions has to be specified. Neural nets have also been used. However, the number of cells in hidden layers is chosen arbitrarily. Linear auto-associative memories appear to be well suited for person and non-person classification.

8.1.2 The IST CAVIAR Data

The CAVIAR video surveillance database consists in 24 video sequences composed of approximately 20000 images of people with hand labeled bounding boxes. Each bounding box is represented by (x, y, w, h, θ), where (x, y) is the center, (w, h) are the width and and the height and θ is the main orientation. Figure 8.1 shows a representation of a main orientation ridge detected in a CAVIAR video sequence. To train non-person regions, we created two sequences of background from where are taken random imagettes. For tests, we use 14 sequences including 12 other sequences in CAVIAR database and 2 background sequences. The sequences contain 9452 people regions and 4990 non-people regions.

8.1.3 Person classification using linear auto-associative memories

We adapted linear auto-associative memories to person classification by using the Widrow-Hoff learning rule [101]. A Bayesian tracker detects the center of gravity and the main orientation for each object in the scene. We use this information to create grey value imagettes normalized in size and orientation as in section 4.4. As shown on Figure 8.2, this normalization step provides robustness to size, chrominance, alignment and orientation.

The problem is to determine the number of persons in a given imagette. We define this problem as a classification problem where the classes are defined according to the number of people. Imagettes of the same class are used for training an auto-associative memory using the Widrow-Hoff correction rule. A connexion matrix W_k is computed for the number of persons k in the imagette, as shown on figure 8.3. The connexion matrix is trained using the Widrow-Hoff correction rule. We obtain two prototypes: one for the 0 person class and one for the $n \geq 1$ persons class. To estimate the number of persons on a given face imagette, a simple winner-takes-all process is employed. We compute the cosine between the source image x and reconstructed

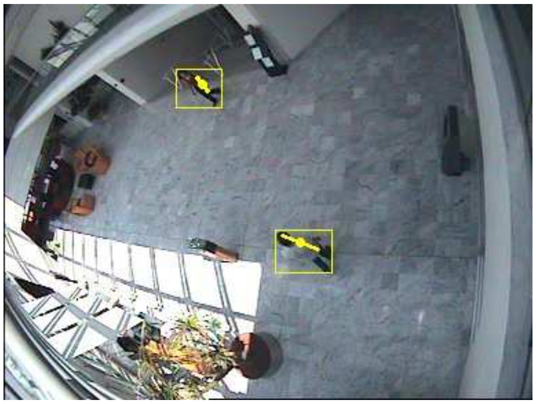

Figure 8.1: Image of a tracking sequence. The bounding boxes represent the regions of interest of tracked persons. The line represents the most significant ridge. The position and the orientation of the region interest is computed using the first and second moments of the difference image

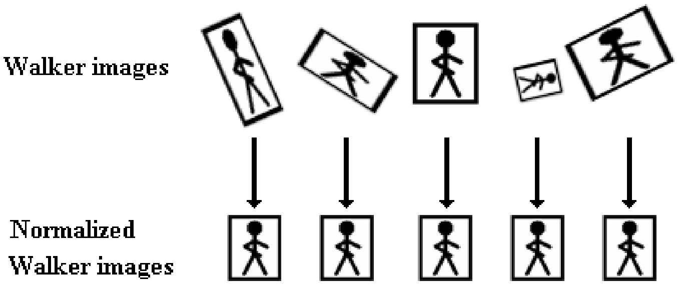

Figure 8.2: The walker image normalization makes features located roughly at the same position

images x'_k. The class whose linear auto-associative memory obtains the best match is selected (8.1).

$$ImageClass = argmax(cos(x, x'_k)) \tag{8.1}$$

We performed 3 experiments to assess our approach on the CAVIAR database. In the first one, we trained an auto-associative memory on the class for 1 person. A threshold value α is used to determine whether the imagette contains a person or not. In the second experiment, we add the 0 persons class for training. In the third experiment, we train 2 auto-associative memories on classes for 0 persons and for $n \geq 1$ persons. We compute recall and precision for each class by varying the size of the imagette.

The recall is defined as the ratio of the number of regions correctly classified and the total number of regions:

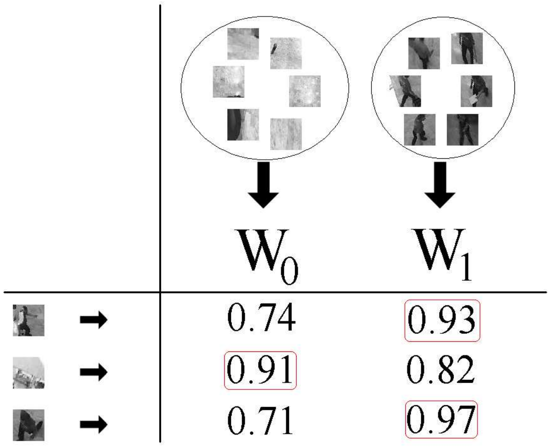

Figure 8.3: Training and test process for classes 0 and 1 person

$$Recall = \frac{Card\{ImagesCorrectlyClassified\}}{Card\{Images\}}$$

The precision is defined as the ratio of the number of regions correctly classified and the sum of number of correct detections and the number of false positives:

$$Precision = \frac{Card\{ImagesCorrectlyClassified\}}{Card\{ImagesCorrectlyClassified\} + Card\{FalsePositives\}}$$

8.1.4 Results and discussion

Results of the first experiment in Figure 8.4 show that training only the class 1 person is not sufficient for reliable classification, even under variations of the threshold value α. This is due to the fact that imagettes which do not contain people present non-uniform variations in appearance. Training the 0 person class improves discrimination between the two classes, as shown in Figures 8.5 and 8.6: 99% correct classification for the 1 person class and 68% for the 0 person class with respectively 95% and 93% precision. By considering the $n \geq 1$ persons class, we obtain comparable results: 99% correct classification for the $n \geq 1$ person class and 70% for the 0 person class with respectively 96% and 90% precision.

The lowest score obtained by the 0 person class can be understood as follows. The 0 persons class is created from randomly chosen imagettes from the background. Some of these imagettes contains some elements whose appearances are similar to persons. Examples of such elements are shown on Figure 8.7: information kiosk, a reception desk, and a pillar. Imagettes containing these elements can easily be misclassified as 1 person imagettes. Therefore the recall for the 0 person class is lower than for the 1 person class. Results also show that varying the size of

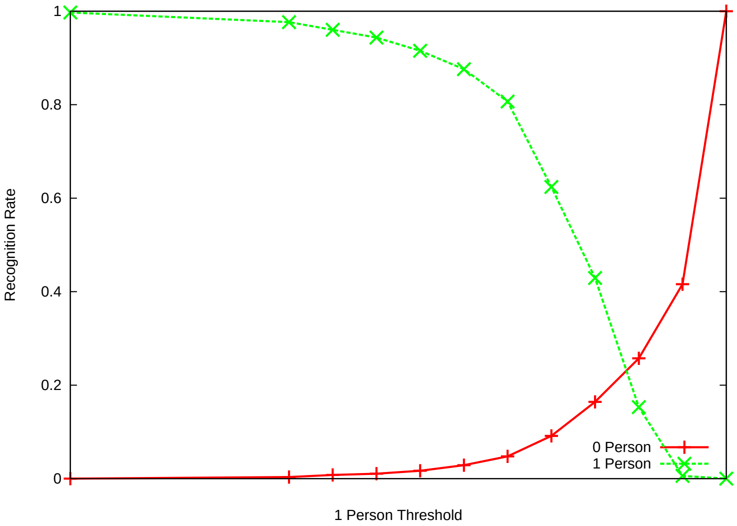

Figure 8.4: Correct classification in the first experience

the normalized imagette does not have much influence on the results. Thus we have elected to maintain a size of 25x25 pixels. Normalization and classification are done at video-rate. We believe that this approach is also well-suited to identity recognition in video sequences as well as to the split and merge problem.

8.1.5 Comparison with three statistical methods

Within the CAVIAR project, the PRIMA research group developed three other classification algorithms for the detection of imagettes containing people. These are the works of A. Negre and H. Tran [100]. The following subsections briefly present their methods along with application to person and non-person classification. We then compare the performance of our system and systems based on other approaches using the same data set.

Ridge extraction

A ridge appears on an image whenever there is a connected sequence of pixels having intensity values which are higher or lower in the sequence than those neighbouring the sequence. With this definition, a ridge can be considered as an approximate medial axis of an oblong object such as a road in a satellite image or a blood vein in a medical image. Given a two-dimensional signal $f(x, y)$, a ridge point is a point at which the signal $f(x, y)$ presents a local extrema in one direction. In case of a maximum, it is a positive ridge point. In case of a minimum, it is a negative ridge point. These two types of points are referred to as *ridge points* because they have the same nature. Geometry shows that at every point of a given surface, there are two main directions corresponding to the largest and smallest curvature of the surface at this point. We

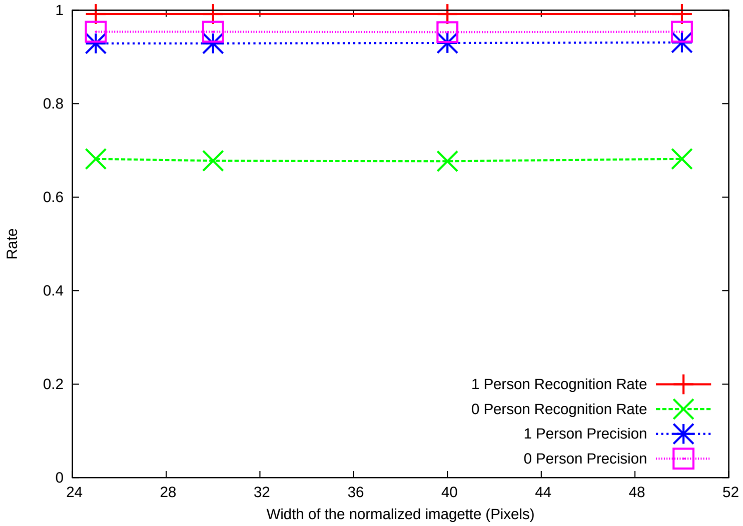

Figure 8.5: Correct classification and precision in the second experience

take the direction corresponding to the largest curvature to determine ridge points.

The definition of ridge is general for any signal. An image is defined by a 2-dimensional function $I(x, y)$. Detecting ridge points in this image consists in detecting ridge points in the surface defined by $z = I(x, y)$. However, the use of the original image signal is limited to detecting only points representing structure of one pixel in size. In addition, the original signal is often noisy. To eliminate noise as well as to have features representing structures of larger than one pixel in size, we need to smooth the image by a Gaussian. Ridges are detected from surfaces defined from the smoothed image $L(x, y; \sigma) = G(x, y; \sigma) * I(x, y)$ at multi-scale [144]. To perform this, the two main directions of the surface at all points (x,y) are calculated with the first and second order derivatives of the smoothed image at a scale σ. The main directions coincide with the two eigenvectors of the Hessian matrix $\nabla\nabla L$. The Hessian matrix is defined as:

$$\nabla\nabla L = \begin{pmatrix} L_{xx} & L_{xy} \\ L_{xy} & L_{yy} \end{pmatrix}$$

We then verify if the normalized Laplacian in the direction of the eigenvector corresponding to the largest curvature admits a local extrema. If so, the point is a ridge point. Once all ridge points are detected, we link neighbour ridge points of the same direction of eigenvector to build ridge lines. In the following, ridge lines will be used to represent human parts.

Ridge structures represent a person on an image in a more structural way, near human perception. Person detection is perfomed by learning different configurations of the human silhouette. Each region containing one or more person will be represented by a descriptor. At a well chosen scale, ridges serve to describe a persons' main axis corresponding to torso and legs. An example can be seen on figure 8.8. We see that ridges well represent oblongated structures

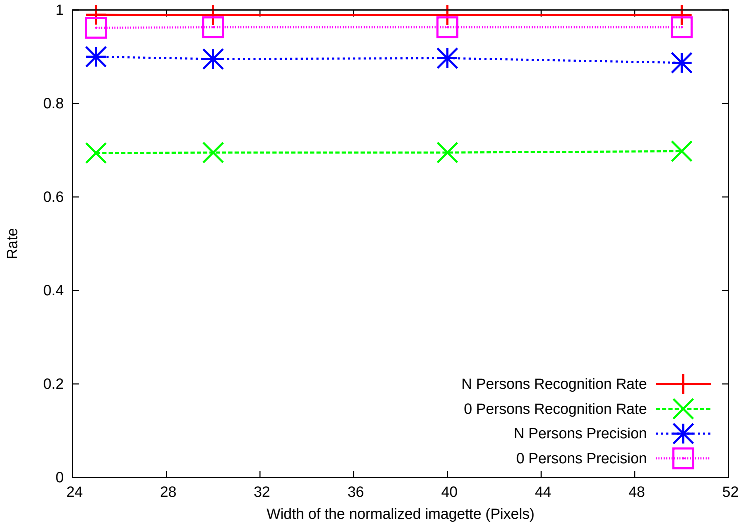

Figure 8.6: Correct classification and precision in the third experience

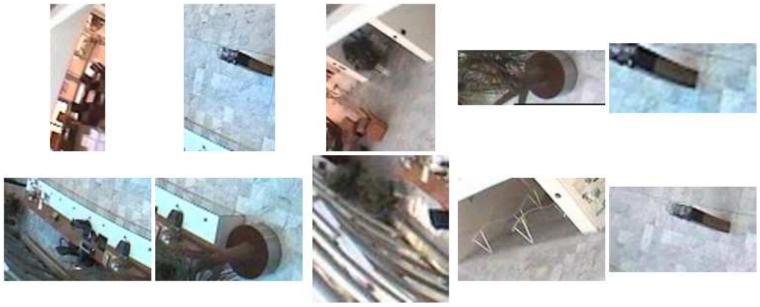

Figure 8.7: Misclassified imagettes

and the topology of a person. As the camera is static in the CAVIAR video sequences, we can compute the orientation of a person, which helps us to quickly determine torso and legs parts.

A person model is described by 3 main ridges corresponding to medial axis of his torso and his legs. There are sometimes no ridges to represent the torso or only one ridge for the leg part. This happens for example when the observed person wears a T-shirt or trousers of the same color as the background.

To test the performance of the method, we use the same sequences for training (12 sequences) and testing (12 sequences). For each region, a model is built and then compared with the 34 person models obtained by K-Means clustering in the database. Each match is caracterised by model identification and the dissimilarity measure. A small value for this measure indicates that the region is similar in appearance to a person. Figure 8.9 shows the result of person recognition varying according to the probability of non-person occurence α.

Classification of person and non-person are optimal for a value of α of 0.9. The correspond-

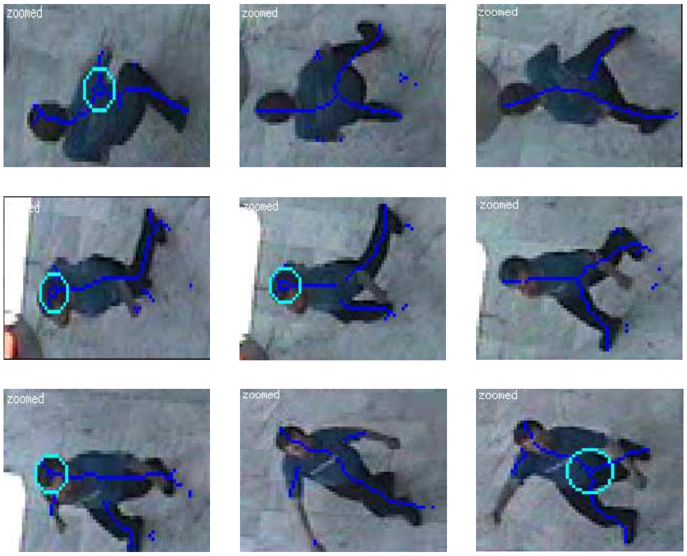

Figure 8.8: Different configurations of a person represented by ridges (blue lines) and blobs (cyan circles) at scale $\sigma = 4\sqrt{2}$. A blob is a local extrema of the Laplacian in 4 directions [101].

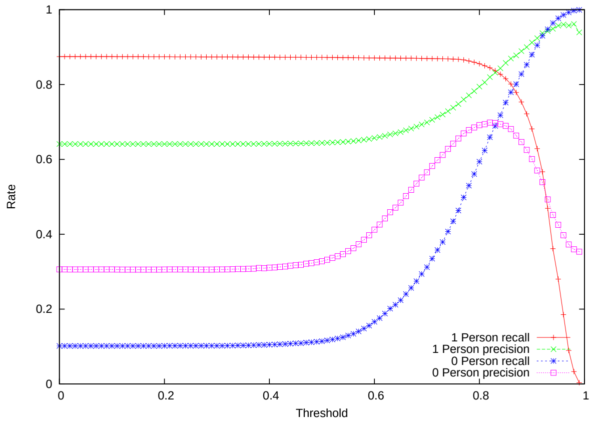

Figure 8.9: Recognition rate and precision for classes 1 person and 0 persons. The alpha value is chosen at the maximum of the average recognition rate.

ing recall is equal to 80%. The use of ridges allows us to detect both the presence of a person and the configuration of this person in an image region.

Ridge normalized gradient histograms

Ridge normalized gradient histograms represent a person by a principal ridge detected in scale space and describe this ridge by histogram of Gradient magnitude and orientation. This approach is similar to those based on Gaussian receptive fields histograms [115] and SIFT descriptors [82]. Person model construction is composed of 2 steps.

In the first step, ridge points are detected in scale space. In order to obtain video-rate performance, a pyramid algorithm is used to compute the Laplacian images [43] needed to compute ridges. Ridges are extracted at each scale level as described in the previous section. Ridge lines are constructed by performing connected component analysis in the (x, y, σ) space. Two ridge points are assimilated to the same ridge line if there are both local minima or maxima and their angle is inferior to a threshold. Each ridge line is caraterised by its centre of gravity μ weighted by the absolute value of the normalized Laplacian, its covariance matrix C_{ij} and its intrinsic scale σ_m.

In the second step, we select the most significant ridge by calculating the mean energy of Laplacian computed at each ridge point. Gradient magnitude and orientation are calculated at each point belonging to the most significant ridge. The magnitude is normalized by the anisotropic Gaussian $G(\sigma_1, \sigma_2)$, where $\sigma_1 = 2\sqrt{\lambda_1}$, λ_1 is the highest eigenvalue of the covariance matrix C_{ij} and $\sigma_2 = \sigma_m$ is the mean intrinsic scale of the ridge line. As a consequence, this normalization gathers information around the central point of the main ridge. Gradient orientation is computed with regard to the main orientation θ of the bounding box. At the construction of the histogram, a four-point-linear interpolation is used to distribute the value of the gradient in adjacent cells. This method is needed to avoid boundary effects. To handle intra-class variations and computational time, person models are clustered using the K-Means algorithm. Comparison between two histograms is performed using the χ^2-divergence distance. Person detection by ridge normalized gradient histograms is evaluated the same way as for ridge extraction. Figure 8.1.5 shows the recall and precision by varying the probability of non-person occurence α.

Classification of an imagette with person and non-person classes are optimal for a value of α of 0.09. The recall is equal to 82%, which is slightly better than the recall obtained by ridges in the previous section. This performance is due to the normalization of the gradient using the second derivatives which are especially well adapted to images of persons walking because of the strong ridge lines. Ridge normalized gradient histograms also have non-person misclassification problems. Non-persons imagettes similar in appearance to people are classified as persons. This method also tends to be sensitive to local illumination changes and partial occlusion.

Performance comparison

Table 8.1 shows the performance of 4 human classification techniques: the three techniques presented in the previous sections and one technique using SIFT descriptor computed at the

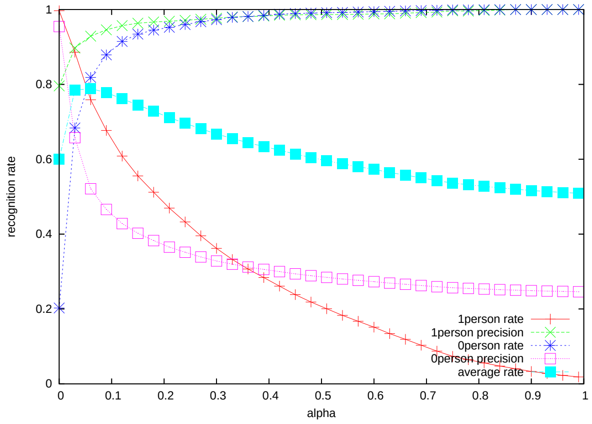

Figure 8.10: Recognition rate and precision for classes 1 person and 0 person. The alpha value is chosen at the maximum of the average recognition rate.

most significant interest points detected in the imagette. This method uses the same technique for learning and testing as the second method. We observe that linear auto-associative memories performs the best when a 0 person class is trained.

The statistical descriptor computed over ridge regions gives better results than the structural descriptor. This is explained by the fact that the first method considers also one ridge as human model. Consequently, all regions containing one ridges are classified as people regions. This method is not good at recognizing non-person regions. The SIFT based method performs worst. The main reason is that interest points are less stable than ridges for representing elongated structures that are typical of images of humans.

Method	Person		Non-Person	
	Recall	Precision	Recall	Precision
Modified SIFT	77 %	90 %	75 %	51 %
Ridge based Structual Model	80 %	90 %	80 %	70 %
Ridge based Normalized Histogram	90 %	93 %	80 %	73 %
Linear Auto-associatives Memories	99 %	96 %	70 %	90 %

Table 8.1: Comparison of recognition methods

Linear auto-associative memories appear to be well suited for person detection. The relatively poor performance obtained for the 0 person class is due to the fact that this class can contain some elements of the background whose appearances are similar to persons. Recognition rate and precision are very high for the 1 person class. This method provides invariance to scale, alignment and orientation. As a global approach, linear auto-associative memories do not need to compute a model for persons and runs at video-rate, but have to learn a 0 person class to be efficient. Ridge-based approaches can be disrupted by neighborhoods of pixels, whereas linear auto-associative memories are robust to partial changes in the imagette.

We believe that linear auto-associative memories can be extended to other vision problems. Ridge configuration models can be useful for movement estimation, but require specific adaptation to other objects. Ridge normalized gradient histograms are well-suited for discrimination of other objects, provided that these objects exhibit a principal ridge. Linear auto-associative memories only require the detection of a region of interest to work. Furthermore, they contain very few parameters to tune and may provide good results for recognition problems, especially for people in video sequences.

8.2 Head Pose estimation on video sequences

In this section, we describe results by evaluating the performance of head pose prototypes on video sequences. Head pose prototypes are created using linear auto-associative memories trained separately in pan and tilt. The use of video sequences introduces a new data to the task: the temporal context.

The temporal context can provide a crucial gain of performance as well as a significant computational time reduction. At a given frame t, we consider that a face has a head pose $P(t)$. The head pose $P(t+1)$ at the next frame is expected to be found in neighbouring poses of $P(t)$. With the use of head pose prototypes, we can restrict research of the current head pose to neighbouring poses, as shown in figure 8.11. Especially, for pan angle, instead of computing the match score for 13 prototypes, we compute only the match score of 5 prototypes, which is less time consuming.

8.2.1 The IST CHIL Data

The IST CHIL database consists of 10 video sequences of people pointing with their heads and their hands. Each sequence contains 1000 frames. All subjects differ from those of the Pointing 2004 database. Head orientation is tracked continously using the head mounted FASTRAK device from Polhemus Inc [56]. Samples of the database are shown on Figure 8.12.

8.2.2 Results and discussion

We trained head pose prototypes separately on the whole Pointing 2004 Database using linear auto-associative memories. We obtained an average error of 22.5 degrees in pan. Our system works at video-rate. Examples of pan angle estimation on the ISL Database can be seen in Figure 8.13.

Head orientations are labelled continously in the ISL Database, which increases the mean error as we have trained discrete head poses. Furthermore, the pan angle is sometimes superior to 90 degrees in both directions. In addition, the face can be occluded by arms in the sequences,

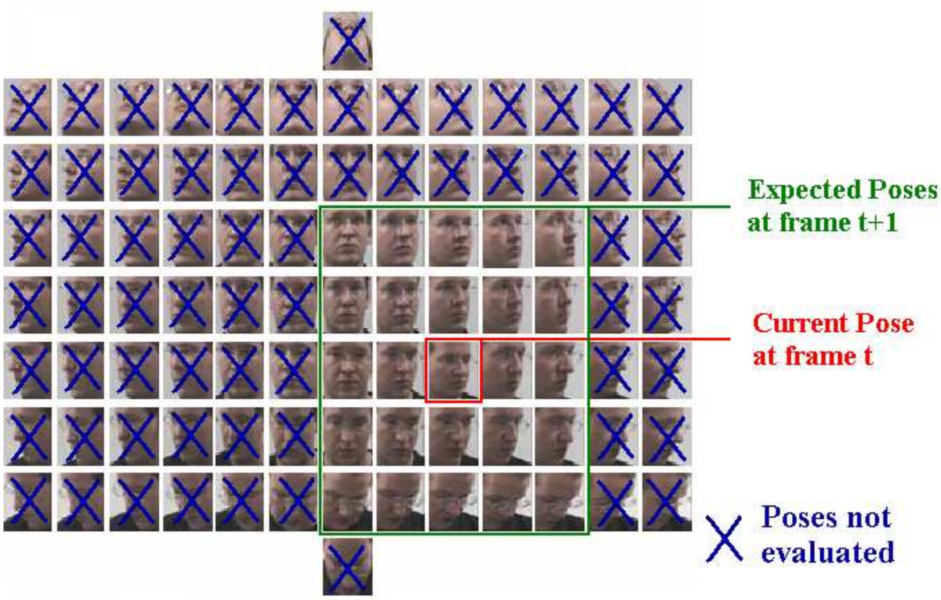

Figure 8.11: Example of expected head poses at next frame. Other poses are not considered.

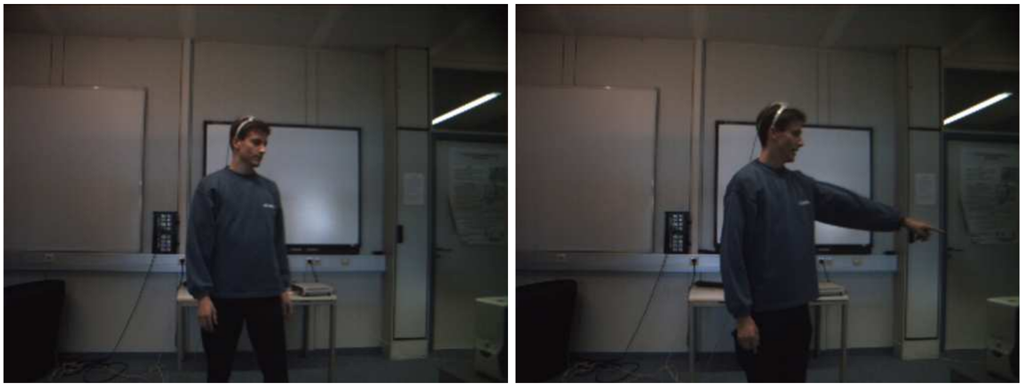

Figure 8.12: Example images of the ISL Pointing Database

and the subject wears a head mounted device, which disrupts face tracking and head pose estimation. Examples of such problematic images are shown on Figure 8.14.

In case of wrong head pose estimation, the head pose tracker may be stuck and may continue to deliver wrong poses in next frames. The score obtained by matching the prototypes with the current pimage can be considered as a confidence factor of the estimation. If the best score is lower than a certain threshold, we consider that the head pose tracker is lost and we reinitialize it at the frontal pose.

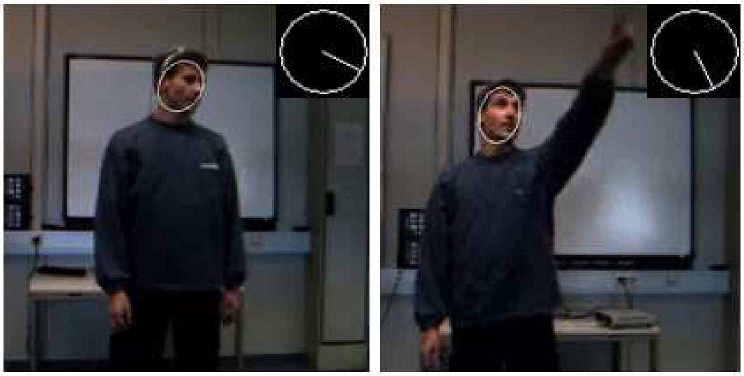

Figure 8.13: Pan angle estimation on example images

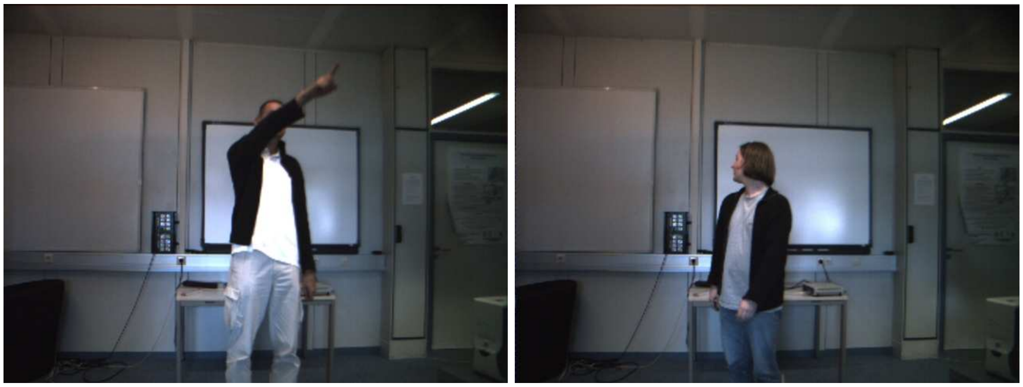

Figure 8.14: Example of problematic images of the ISL Pointing Database

8.3 Attentional Systems

Head pose is only a part of human attention. The concept of attention is generally difficult to define because it comprises visual focus of attention, auditive focus of attention as well as cues about the intention, nature and the implication of the subject in his task. Such cues can be the use of the mouse, the frequency of keyboard strokes and other existing interaction devices. Human attention is also hardly possible to measure precisely because there are no metrics adapted to it, nor does an unified framework exist.

Informatic systems describing the human attention have recently been proposed. These systems are called attentional systems and aim at evaluating people's attention to model social interactions, detecting privacy violation and evaluating the disponibilty of the user. The system proposed by Horvitz [49] models people's attention with ontologies and a set of fixed rules. More recently, Maisonnasse [85] has proposed an attentional system based on a gravitational model that includes interesting concepts which recover attention properties. Any sensor can provide observations for this model, without defining prior knowledge or specific rules. Head pose estimation can serve as an input for this model.

Focus of attention is computed to delimit context boundaries for each user and to detect whether people share the same resources, on the basis of their position and the salience of contextual elements. The focus of a person is defined by the direction of attention which is the combination of its external and internal factors. We can see an example of external and internal factors on figure 8.15.

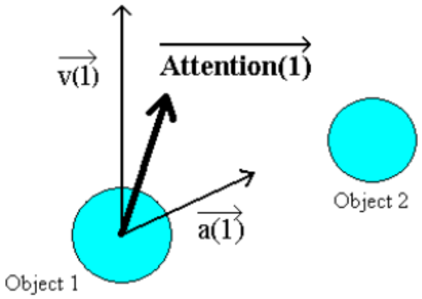

Figure 8.15: The attention vector object 1 $Atten\vec{t}ion(1)$ is a combination of the external factor $a(\vec{1})$ and the internal factor $v(\vec{1})$

The external factor of a person is determined by the attraction coming from other people, objects or artefacts which inhabit the environment. It is based on a gravitational model simulating persons' attraction towards other persons or objects. Salience can be defined on perceptive, social or situation features.

The internal factor, or intentionality of a person, is determined by the person's current goal or current activity, regardless of environment. This factor can be assimilated to the concept of intentionality. Cues of intentionality of a person are for example its current speed, gaze direction, and especially head pose. The internal factor is also represented by a vector that can be perceived as an important directed concentration to an object during a task. Only objects present in the direction of a person's intentionality are considered relevant for the person. We believe head pose estimation could be a good contribution to intentionality representation.

The attentional system can be used to detect when someone pays attention to a device and transgresses privacy. People and objects are tracked with the PRIMA Robust Tracker [12]. The system detects every entity in the environment and converts their positions from image to environment using an homography. An example of real situations and their representations through the attentional model can be seen on figure 8.16. By evaluating people's focus of attention, the system can act on window environment and adapt the services to the situation where the user is [86]. The face tracker described in Chapter 4 could be launched inside the detected body region to delimit the face region. Head pose estimation can provide an indicator of people's attention and privacy violation.

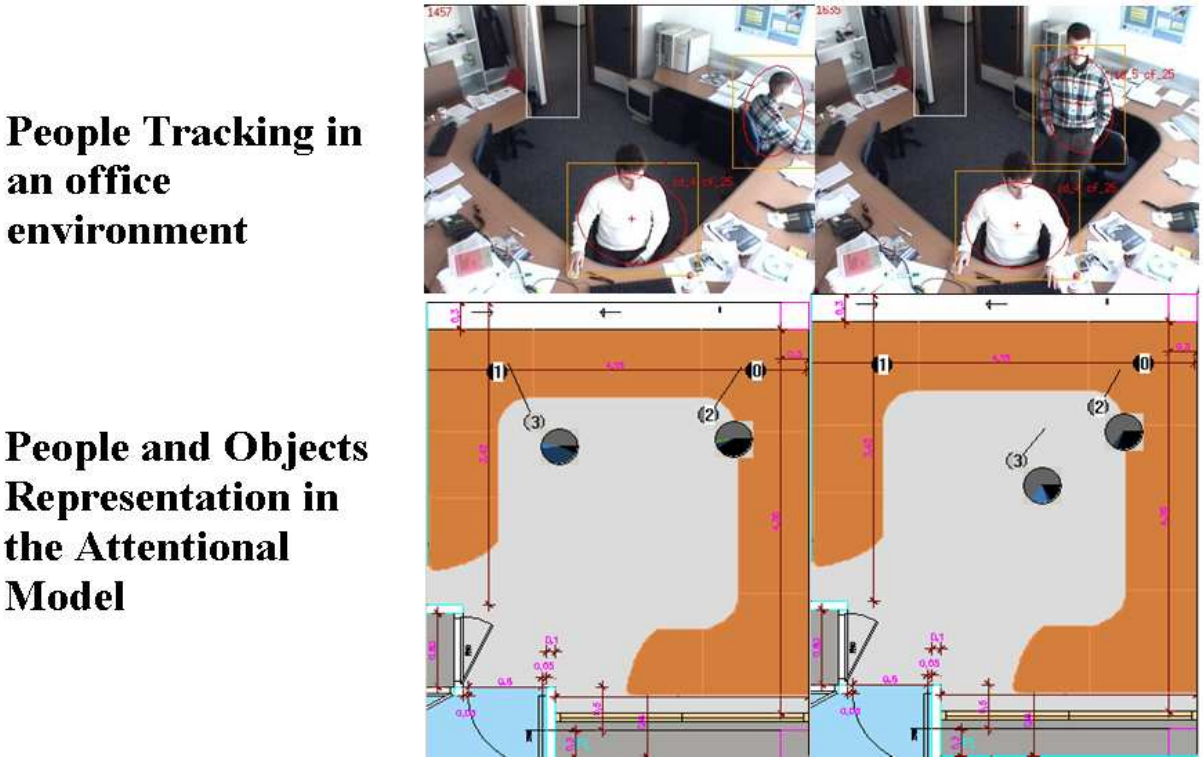
People Tracking in an office environment
People and Objects Representation in the Attentional Model
1457
1655
1
0
2
3

Figure 8.16: Example of privacy violation on the upper right image. Person 3 is gazing at Person 2’s screen [86]

Chapter 9

Conclusions

Inspired by global and local computer vision approaches, we have investigated a two-stage coarse-to-fine head orientation estimation based on linear auto-associative memories and salient Gaussian receptive field graphs. Training head pose prototypes from unconstrained normalized low resolution face images provides a simple, fast and efficient means for recovering coarse head orientation. With this approach, pan and tilt angle can be learned separately. Results can be improved by using grid graphs where each node is represented by Gaussian receptive field vectors. Nodes are displaced locally in a manner that maximizes similarity of appearance while conserving the spatial order relation encoded in the graph. Head pose estimation is refined by searching for the most visually similar model graph within the neighbouring coarse poses. The overall performance is comparable to human performance.

9.1 Principal Results

In our experiments with human abilities for head pose estimation, we observed an average error of 11.85^o in pan and 11.04^o in tilt. We discovered an interesting result for estimating the pan axis. Humans perform well at recognizing front and profile views, but abilities degrade for intermediate views. Pan angle appears to be more natural to estimate. Minimum error in pan is found at 0 degrees, which corresponds to the frontal pose. These results tend to show that the human visual system uses front and profile views as key poses, as suggested in [65]. The age of the subject does not seem to influence human abilities for head pose estimation.

For automatic estimation of head pose, face region images are normalized in position, scale and orientation and saved as low resolution imagettes. Linear auto-associative memories are used to learn prototypes of head pose images. Such memories are very simple to construct, require few parameters, and are thus well suited for head orientation estimation for both known and unknown subjects. Prototypes are trained either separately or together. With an average error of less than 10^o in pan and tilt angles on known faces, the method has better performance than neural networks [152], PCA and tensor model [145]. We achieve an error of 10^o in pan and 16^o

in tilt for unknown subjects. Our method performs well for upward poses. Learning to recognize poses for pan and tilt axis separately provides a significant gain of computational time without loss of performance. Head pose prototypes can be saved and restored for other applications. Our coarse head pose estimation algorithm runs at 15 frames per second, is reliable enough to video sequences for situations such as man-machine interactions, video surveillance and intelligent environments.

Head orientation estimation can be improved by describing face images using Gaussian receptive field responses normalized to intrinsic scale. Gaussian derivatives describe the appearance of neighbourhoods of pixels and are an efficient means to compute scale and illumination robust local features. Furthermore, they have interesting invariance properties. Face images are described using low dimensional feature vectors. Detection of salient facial regions of the face is robust to identity and pose can be recovered by analyzing regions that share the same apperance over a limited region. We have found that the salient facial features detected by normalized Gaussian receptive fields were eyes, nose, mouth and face contour. These results resemble those obtained by humans according to the studies of Yarbus [165].

Gaussian receptive field grid graphs refine the pose obtained from the coarse estimate system. The graph structure describes both neighbourhoods of pixel appearance and their geometric relation within the image. Describing each node at intrinsic scale and using hierarchical clustering gives better results. We also found that graphs covering the whole face image provide better performance than graphs applied to only parts of the image. The larger the region covered by the graph, the more geometric relation information it captures. Furthermore, setting nodes' local maximum displacement according to their saliency provides better results than having a fixed value. A node placed at a salient fixation represents something relevant for the considered pose and does not need to move significantly from its original location. On the other hand, a node placed at a non-salient location does not represent any relevant feature and can be moved with a maximal displacement equal to the distance between 2 nodes, in order to preserve geometric relation. We obtained a coarse-to-fine head pose estimation with 10^o in pan and 12^o in tilt for unknown users. Pan angle estimation appear to be contained in the horizontal asymmetry provided by the global appearance of the face image, whereas tilt angle estimation requires local refinement. Our method does not use any heuristics, manual annotation or prior knowledge on the face, provides results comparable to human abilities and can be adapted to estimate the pose of configuration of other deformable objects or to recognize facial emotions.

Head pose estimation on video sequences has been tested using the IST CHIL Pointing database. The temporal context provides an important gain in performance as well as a significant computational time reduction. The head pose at the next frame is expected to be found in neighbouring poses of the previous pose. We found an average error of 22.5^o in pan. Our method can be used on both single images and video sequences.

9.2 Perspectives

Our two-stage coarse-to-fine head pose estimation system has shown good performance with images and video sequences. The first step of the method is to normalize face images in order to work on imagettes normalized in size and slant angle. As a result, the computational time is independant of the size of the source image, but dependent on the size of the imagette. However, the face tracker can also introduce a problem for face normalization. The height of the neck differs from one person to another. This provides high variations on face imagettes and can disrupt tilt angle estimation. Besides, as the face tracker is based on chrominance detection, it can sometimes track an image region whose chrominance is similar to skin chrominance, but is not the head. It can also include non face skin color regions adjacent to the face, for example when a person has his hands near his face. The raster-scan algorithm developed by Peters [109] can locate the face image region by displacing the whole grid graph without displacing its node locally. Yet, in order to correctly delimit the face region, the size of the face in the image must be known. By enclosing the face tracker and the raster-scan algorithms in a closed loop, image normalization and alignment should be improved. A better alignement can also be obtained by using a Hough transform on the face ellipse. This approach offers the advantage of delimiting the face contour, which could avoid the detection of the neck and of other skin regions.

Following the same idea, salient grid graphs could be used to determine whether a facial feature is occluded or not. By removing the contribution of nodes representing the occluded feature, head pose tracking could be enhanced. Another solution in this case is to keep only the result found by linear auto-associative memories, as they are robust to partial occlusions.

Just as we detect salient facial regions as appearance blobs at intrinsic scales, we could also describe facial ridges as appearance ridges. A new ridge description method based on Laplacian energy has recently been demonstrated [144]. Ridges can serve as edges in salient grid graphs. Combining nodes and edges description may potentially improve face matching and head pose recovery.

We did not perform an exhaustive evaluation of our system on face illumination changes. Linear auto-associative memories are disrupted by global illumination changes but are robust to partial illumination changes. On the other hand, Gaussian receptive fields are robust to global illumination changes but are disrupted by partial illumination changes. By intergrating these two methods in a loop, each one can give feedback to the other about its confidence of the pose estimation. By taking into account this confidence, we should be able to choose the most apppropriate method to use at a certain situation to estimate the head orientation.

Increasing resolution of the normalized face imagette enhances precision and can allow continuous head pose estimation. In our study, only discrete head orientations were trained and tested using a winner-takes-all process. We could compute continuous head pose by interpolating discrete poses. Scores obtained on neighbouring head poses provide a good cue for interpolation.

Recently, a new video sequence benchmark on head pose estimation has appeared [152]. These sequences are taken from seminar recordings of 4 cameras. The speaker's head orientation has been aonnotated manually with eight cardinal directions: north, north-east, east, south-east, south, south-west, west and north-west. Pose estimations from 4 different point of views could be combined to obtain a more precise estimate of the head orientation.

As a conclusion, we should not forget that head orientation is only part of human attention. The eye fixation direction with regard to the eye contributes to gaze direction, but this can only be detected on images of sufficient resolution. However, human attention is also difficult to define because it comprises visual focus of attention, auditive focus of attention as well as cues about the intention, nature and the implication of the subject in his task. Systems for estimating attention are beginning to appear, and head pose estimation can serve as an entry for such systems [85]. These systems can provide important information for man machine interaction and context aware observation of human activity.

Appendix A

Pointing 2004 Head Pose Image Database

To our knowlegde, there are very few public databases providing images annotated with head orientation. We wanted to build a reliable database to assess both machine and human performance at head pose estimation. Such a database has to contain:

- a neutral background
- a wide range of head poses
- a dense sampling of head poses
- images of different people

The Pointing 2004 Database consists in a dense sampling of a half view sphere of head poses from different subjects. It can be downloaded for non-commercial use from the following address:

`http://www-prima.inrialpes.fr/perso/Gourier/Faces/HPDatabase.html`

We used a white board as the background in order not to disrupt the face tracker system nor human subjects during the head pose estimation task. As face tracking is an independant problem which is not the focus of our study, the choice of a white background is legitimate. On the one hand, this allows all faces to be treated equally, on the other hand, a well suited segmentation operation can separate the head region from the background. Training and testing are done using this neutral background, but our system can adapt to an ordinary background using a good face tracker.

To take images of the same subject in a half view sphere of poses, we could think of photographing him using a geodesic dome. However, this approach would consider the human head as a rigid 3D object, which is not consistent with the head pose estimation problem. Indeed, the image of a face taken from a certain view angle is different from the image of the same face oriented with the same angle. An example is shown on figure A.1. The human head is a deformable object. We thus must take images which really capture different head poses from people. There exist head mounted devices, such as FASTRAK [56] from Polhemus Inc., which give the head orientation of a subject with a precision inferior to 3 degrees. The main drawback of such systems is that such devices act like artefacts as they are highly visible on the image and thus can disrupt the pose estimation process.

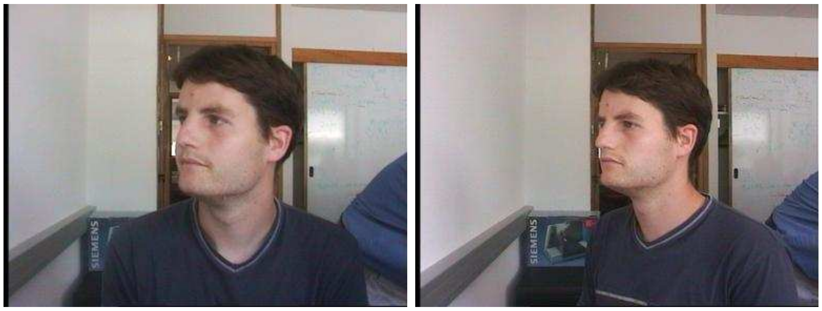

Figure A.1: On the left image, the person looks straight with a head orientation of 45 degrees. On the right image, he looks straightforward and the image is taken under a view angle of 45 degrees. Images are different, especially in the neck region

Images have been taken in the FAME Platform of the PRIMA Team in INRIA Rhone-Alpes using a Sony CCD Camera. To obtain different poses, we put markers in the whole room. Each marker corresponds to a pose (h,v). Post-its are used as markers. The whole set of post-its covers a half-sphere in front of the person, as indicated in figure A.4. Experimental setup is shown on figure A.3. To ensure the face is centered on the image, the person is asked to adjust the chair to see the device in front of him. After this initialization phase, we ask the person to stare successively at the markers, without moving his eyes. This second phase only takes a few minutes. When a subject gazes at a post-it marked (h,v) without moving his eyes, his head orientation corresponds to the pose (h,v). All images of our database are obtained using this method.

The head pose database consists in 15 sets of images. Each set contains of 2 series of 93 images of the same person at different orientations. Images are in PPM format and have a resolution of 384x288 pixels. The pose varies from -90 degrees to +90 degrees in pan and tilt axis. A sample of a serie is shown on figure A.2. There are 13 angles in pan axis and 9 angles in tilt axis. In the case when the tilt angle is -90 or +90, the person is looking at the bottom or the top, and then the pan angle is 0. Each serie therefore contains 7 x 13 + 2 x 1 = 93 images. Here is the sampling of pan and tilt angles used in the Pointing 2004 database:

- Pan: -90^o,-75^o,-60^o,-45^o,-30^o,-15^o,0^o,$+15^o$,$+30^o$,$+45^o$,$+60^o$,$+75^o$,$+90^o$
- Tilt: -90^o,-60^o,-30^o,-15^o,0^o,$+15^o$,$+30^o$,$+60^o$,$+90^o$

	Negative Values	Positive Values
Pan Angle	Left	Right
Tilt Angle	Bottom	Top

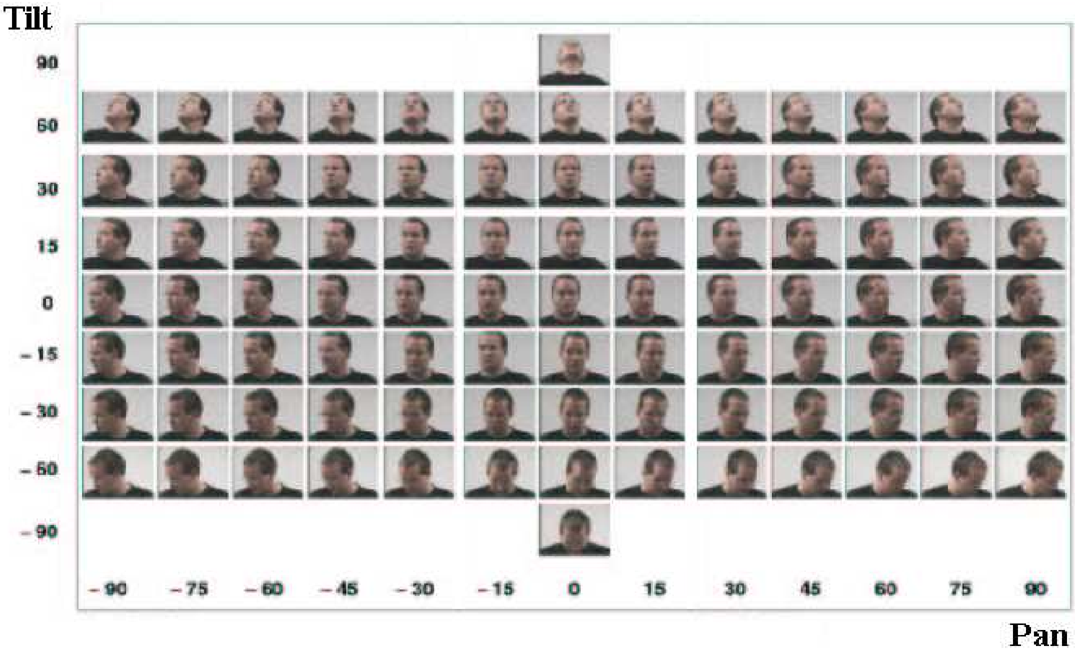

Figure A.2: Example of a serie of the Head Pose Image Database.

Subjects are male or female of different ages, wear glasses or not and have varied skin colors. The Pointing 2004 Head Pose Image database provides a reliable framework to perform head pose estimation.

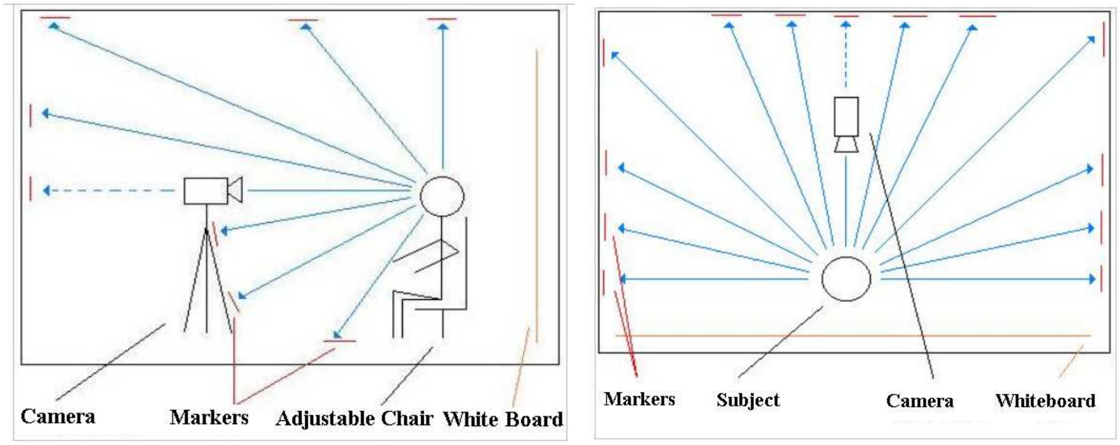

Figure A.3: Side and top view of image acquisition

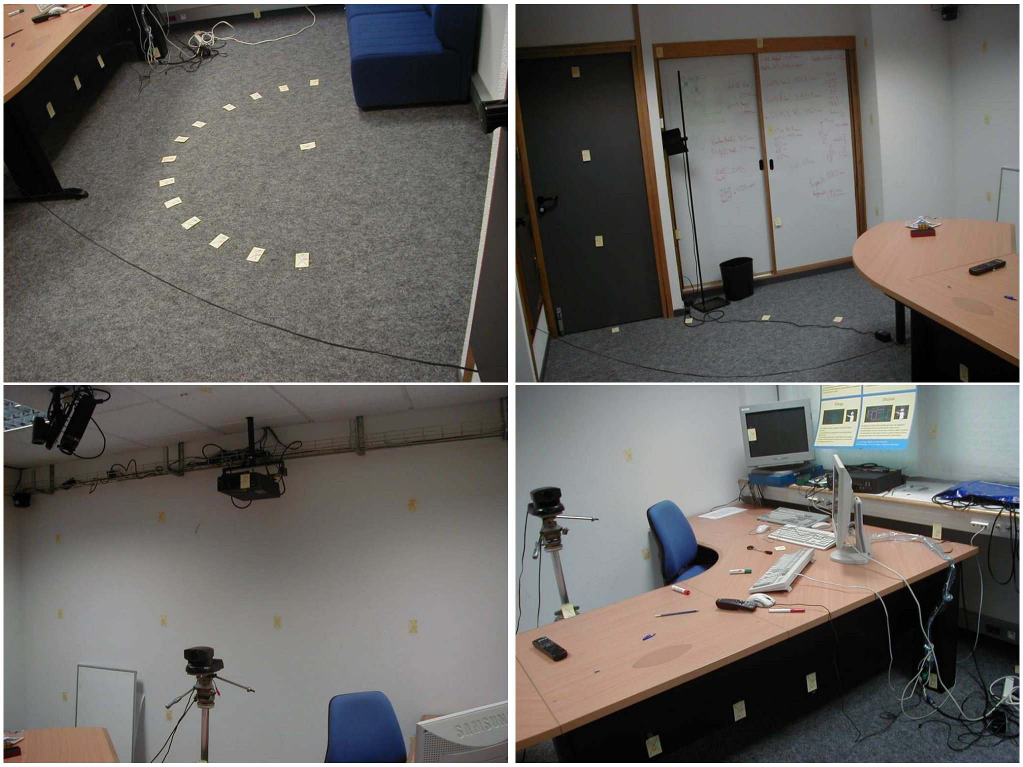

Figure A.4: Images from the FAME Platform of the PRIMA Team in INRIA Rhone-Alpes with the camera and the markers used for image acquisition

Appendix B

Statistical Operations

This section details the statistical operations applied in this thesis. Primary notions such as random variables, expected value, variance and standard variations are mentionned in the first part. The concept of unbiased estimator is explicited in the second section. The third part explains the Test of Student-Fisher. This is a well-known test used to compare performance of groups of population. The use of the correlation coefficient is illustrated in the last section to determine a possible connection between two random variables.

Random Variables

A random variable X is a function that associates an unique value with every outcome of an experiment. The value of a random variable varies from trial to trial as the experiment is repeated. There are two types of random variables: discrete and continuous. A discrete random variable has an associated probability distribution, whereas a continuous random variable has a probability density function. A realisation of X is denoted x_i. Let N be the number of realisations of the variable X. We have:

$$X = (x_1, x_2, ..., x_{N-1}, x_N)$$

The expected value of the random variable X, denoted $E(X)$ or μ_x, is a linear operator which indicates its average or central value. Stating the expected value gives a general impression of the behaviour of some random variable without giving full details of its probability distribution. The expected value of a discrete random variable X is defined by:

$$E(X) = \mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i$$

There are other useful descriptive measures which affect the shape of the distribution, such as the variance. The variance of a random variable X, denoted $Var(X)$ or σ_x^2, is a positive number which gives an idea of how widely spread the values of the random variable are likely to be. The larger the variance is, the more scattered the observations around the average are. Stating the variance gives an impression of how closely concentrated around the expected value the distribution is. The square root σ_x of the variance is called the standard deviation. The variance of a discrete random variable X is given by:

$$Var(X) = \sigma_x^2 = E((X - \mu_x)^2) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu_x)^2$$

The covariance measures the extent to which two random variables with the same number of realisations vary together. The covariance of X and Y is denoted $Cov(X, Y)$ or σ_{xy}. Its calculation begins with pairs of x_i and y_i, takes their differences from their mean values and multiplies these differences together. For instance, if the product is positive, these pairs of data points the values of x_i and y_i will vary together in the same direction from their means. If the product is negative, they will vary in opposite directions. If the covariance is zero, then the cases in which the product was positive were offset by those in which it was negative, and there is no linear relationship between the two random variables. The larger the magnitude of the product, the stronger the connection of the relationship. The covariance is defined as the mean value of this product:

$$Cov(X, Y) = \sigma_{xy} = E((X - \mu_x)(Y - \mu_y)) = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y)$$

Unbiased Estimators

We are interested in an unknown parameter a of the model. A statistic $\hat{a}$ that is used to estimate the parameter is called an estimator of a. The error of the statistic $\hat{a}$ is defined as the difference $\hat{a} - a$ between the estimator and the parameter. The expected value of this error is known as the bias of the estimator:

$$\begin{aligned} Bias(\hat{a}) &= E(\hat{a} - a) \\ &= E(\hat{a}) - E(a) \\ &= E(\hat{a}) - a \end{aligned}$$

The estimator is said to be unbiased if the bias is equal to 0. Tis corresponds to the case in which the expected value of the estimator is the parameter being estimated:

$$E(\hat{a}) = a$$

A natural estimator for the expected value μ_x of the random variable X is the arithmetic average of its realisations x_i: $\hat{\mu}_x = \frac{1}{N}\sum_{i=1}^{N} x_i$. The estimator verifies the condition:

$$
\begin{aligned}
E(\hat{\mu}_x) &= \frac{1}{N}\sum_{i=1}^{N} E(x_i) \\
&= \frac{1}{N} N E(X) \\
&= E(X) \\
&= \mu_x
\end{aligned}
$$

The variance σ_x^2 of X can be reformulated as:

$$
\begin{aligned}
\sigma_x^2 &= \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu_x)^2 \\
&= \frac{1}{N}\sum_{i=1}^{N} x_i^2 - \frac{2}{N}\mu_x \sum_{i=1}^{N} x_i + \mu_x^2 \\
&= \frac{1}{N}\sum_{i=1}^{N} x_i^2 - \frac{2}{N^2}(\sum_{i=1}^{N} x_i)^2 + (\frac{1}{N}\sum_{i=1}^{N} x_i)^2 \\
&= \frac{1}{N}\sum_{i=1}^{N} x_i^2 - \frac{1}{N^2}\sum_{i=1}^{N} x_i^2 - \frac{2}{N^2}\sum_{i<j} x_i \cdot x_j \\
&= \frac{N-1}{N^2}\sum_{i=1}^{N} x_i^2 - \frac{2}{N^2}\sum_{i<j} x_i \cdot x_j
\end{aligned}
$$

We compute the expected value of this quantity. By definition, the variance does not depend on the mean μ_x of the data. Thus the expected value of every quantity x_i^2 is equal to the variance σ_x^2. The terms $x_i \cdot x_j$ take the shape of covariances. However, the experiments are considered independant, so these covariance terms become null. We obtain:

$$
\begin{aligned}
E(\sigma_x^2) &= \frac{N-1}{N^2}\sum_{i=1}^{N} E(x_i^2) \\
&= \frac{N-1}{N^2} \cdot N \cdot \sigma_x^2 \\
&= \frac{N-1}{N}\sigma_x^2
\end{aligned}
$$

This estimator is biased. The unbiased estimator for the variance σ_x^2 of the random variable X is thus:

$$\hat{\sigma}_x^2 = \frac{N}{N-1}\sigma_x^2 = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)^2$$

The unbiased estimator for the covariance σ_{xy} of the random variables X and Y is obtained using the same method:

$$\hat{\sigma}_{xy} = \frac{N}{N-1}\sigma_{xy} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y)$$

Test of Student-Fisher

To determine an interval in which a realisation of a random variable can be found, we use hypothesis tests. With a large number N of realisations, a random variable X follows a normal distribution centered on μ_x. We want to know which is the probability α that the expected value of X is in the interval 2ϵ. This problem can be reformulated as follows:

$$P(\|x - \mu\| < \epsilon) = \alpha$$

The interval 2ϵ is called the trust interval. It is determined by the confidence threshold α. The value 95% for α is generally used for most statistic problems. In the case X follows a normal distribution, the corresponding value for ϵ is $1.96\sigma_x$. There are 95% of chances to find the expected value of X in the interval $[\mu - 1.96\sigma_x, \mu + 1.96\sigma_x]$. In our experiements, we consider that we a have a sufficiently large number of realisations to apply normal distributions.

Let X and Y be two random variables measurable with the same metric. We want to know if the a group $(x_1, x_2, ..., x_{N-1}, x_N)$ of N realisations of X is significantly better than a group $(y_1, y_2, ..., y_{M-1}, y_M)$ of M realisations of Y. The random variable of Student associated to the difference $X - Y$ can be estimated by:

$$T = \frac{\hat{\mu}_x - \hat{\mu}_y}{\sqrt{\frac{\hat{\sigma}_x^2}{N} + \frac{\hat{\sigma}_y^2}{M}}}$$

The group $(x_1, x_2, ..., x_{N-1}, x_N)$ performs better than the group $(y_1, y_2, ..., y_{M-1}, y_M)$ if $T > 1.96$. It signifies that for there are at least 95% of chances for a given realisation of X of being better than a given realisation of Y. We use the Test of Student-Fisher to compare the performance of groups of humans and our system.

Correlation Coefficient

The correlation coefficient $\rho(X, Y)$ is frequently used in statistics to determine a possible link between two random variables X, Y. The covariance $cov(X, Y)$ measures the correlation that may exist between X and Y. However, to be able to compare a set of data with another, we need to normalize the covariance by the product of standard deviations $\sigma_x \cdot \sigma_y$. The two random variables must have the same number of realisations. The correlation coefficient is then comprised between -1 and 1. A score of 0 means that X and Y are completely uncorrelated, whereas a score of ± 1 means that X and Y are completely correlated. The correlation coefficient is defined by:

$$\rho(X,Y) = \frac{cov(X,Y)}{\sqrt{\sigma_x \cdot \sigma_y}} = \frac{\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y)}{\sqrt{\sum_{i=1}^{N}(x_i - \mu_x)^2 \sum_{i=1}^{N}(y_i - \mu_y)^2}}$$

For our experiments, we use the unbiased estimator $\hat{\rho}(X,Y)$ of the correlation coefficient, obtained with the unbiased estimators of the variances σ_x and σ_y and the covariance σ_{xy}:

$$\hat{\rho}(X,Y) = \frac{\hat{\sigma}_{xy}}{\sqrt{\hat{\sigma}_x \cdot \hat{\sigma}_y}}$$

Bibliography

[1] H. Abdi and D. Valentin. Modèles neuronaux, connexionistes et numériques de la reconnaissance des visages. *Psychologie Française*, 39(4):357–392, 1994.

[2] A. Azarbayejani, B. Horowitz, and A. Pentland. Recursive estimation of structure and motion using the relative orientation constraints. *Computer Vision and Pattern Recognition*, 1993.

[3] A. Azarbayejani, T. Starner, B. Horowitz, and A. Pentland. Visually controlled graphics. *IEEE Transactions on PAMI*, 15(6):602–605, 1993.

[4] A.M. Bagci, R. Ansari, A. Khokhar, and E. Cetin. Eye tracking using markov models. In *Proceedings of 17th International Conference on Pattern Recognition*, August 2004.

[5] A. Baumberg. Hierarchical shape fitting using an iterated linear filter. 1996.

[6] M. Bichsel and A. Pentland. Automatic interpretation of human head movements. In *13th International Joint Conference on Artificial Intelligence, Workshop on Looking At People, Chambery France*, 1993.

[7] E. Borovikov. Human head pose estimation by facial features location. Scholarly Paper MD, University of Maryland Institute for Computer Studies College Park, 1998.

[8] T. Brandt, R. Stemmer, and A. Rakotonirainy. Affordable visual driver monitoring system for fatigue and monotony. In *Proceedings of Systems, Man and Cybernetics*, October 2004.

[9] X.L. Brolly, C. Stratelos, and J.B Mulligan. Model-based head pose estimation for air-traffic controllers. *International Consortium for Integrational Programs*, 2003.

[10] L.M. Brown and Y-L. Tian. Comparative study of coarse head pose estimation. *IEEE Workshop on Motion and Video Computing*, December 2002.

[11] G.L. Calhoun and G.R. McMillan. Hands-free input devices for wearable computers. In *Hands-Free Input Devices for Wearable Computers*, 1998.

[12] A. Caporossi, D. Hall, P. Reignier, and J.L. Crowley. Robust visual tracking from dynamic control of processing. *Performance and Evaluation of Tracking and Surveillance*, PETS'04, 2004.

[13] L. Chen, L. Zhang, Y. Hu, M. Li, and H. Zhang. Head pose estimation using fisher manifold learning. *ICCV International Workshop on Analysis and Modeling of Faces and Gesture*, October 2003.

[14] Q. Chen, H. Wu, T. Fukumoto, and M. Yachida. 3d head pose estimation without feature tracking. In *Proceedings of Third International Conference on Automatic Face and Gesture Recognition*, pages 88–93. IEEE Computer Society Press, April 1998.

[15] K.N. Choi, M. Carcassoni, and E.R. Hancock. Recovering facial pose with the em algorithm. *Pattern Recognition*, 35(10):2073–2093, 2002.

[16] C. Collet. *Capture et suivi du regard par un systeme de vision*. PhD thesis, Ecole Normale Superieure de Cachan, 1999.

[17] A. Colmenarez, R. Lopez, and T.S. Huang. 3d model-based head tracking. In *Proceedings of the International Society for Optical Engineering 3024 Serie 1*, pages 426–434, 1997.

[18] C. Colombo and A. Del Bimbo. Head pose estimation for graphic remapping by visual tracking of eye appearance. In *Proceedings of AI*IA Workshop*, 1998.

[19] J.L. Crowley and O. Riff. Fast computation of scale normalised receptive fields. In *International Conference ScaleSpace, Island of Skye*, pages 584–598, 2003.

[20] T. Darrell, B. Moghaddam, and A. Pentland. Active face tracking and pose estimation in an interactive room. *IEEE Conference on Computer Vision and Pattern Recognition*, 1996.

[21] T. Darrell, K. Tollmar, F. Bentley, N. Checka, L-P. Morency, A. Rahimi, and A. Oh. Face-responsive interfaces : From direct manipulation to perceptive presence. *International Conference of Ubiquitous Computing*, 2002.

[22] T. D'Orazio, M. Leo, G. Cicirelli, and A. Distante. An algorithm for real time eye detection in face images. In *Proceedings of 17th International Conference on Pattern Recognition*, August 2004.

[23] F. Dornaika and F. Davoine. Online appearance-based face anf facial feature tracking. In *Proceedings of 17th International Conference on Pattern Recognition*, August 2004.

[24] E. Elagin, J. Steffens, and H.Neven. Automatic pose estimation system for human faces based on bunch graph matching technology. *Automatic Face and Gesture Recognition*, pages 136–141, 1998.

[25] P. Fitzpatrick. Head pose estimation without manual initialization. *Term Paper for MIT Course, Cambridge, MA*, 6.892, 2001.

[26] F. Fleuret and D. Geman. Fast face detection with precise pose estimation. *International Conference on Pattern Recognition*, 1:235–238, 2002.

[27] W. T. Freeman and E. H. Adelson. The design and use of steerable filters. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(9):891–906, 1991.

[28] C. Galev and A.F. Monk. Where am i looking? the accuracy of video-mediated gaze awareness. *Perception and Psychophysics*, 62(3):586–595, 2000.

[29] A. Gee and R. Cipolla. Estimating gaze from a single view of the face. In *Proceedings of 12th International Conference on Pattern Recognition 1*, pages 758–760, 1994.

[30] A. H. Gee and R. Cipolla. Non-intrusive gaze tracking for human-computer interaction. In *Proceedings of Mechatronics and Machine Vision in Practise*, pages 112–117, 1994.

[31] A.H. Gee and R. Cipolla. Determining the gaze of faces in images. Technical Report CUED/FINFENG/TR 174, Cambridge University Departement of Engineering, March 1994.

[32] A.H. Gee and R. Cipolla. Fast visual tracking by temporal consensus. *Image and Vision Computing*, 14(2):105–114, 1996.

[33] J. Gips, P. Olivieri, and J.J. Tecce. Direct control of the computer through electrodes placed around the eyes. In *Human-Computer Interaction: Applications and Case Studies*, pages 630–635. Elsevier, 1993.

[34] S. Gong, S. McKenna, and J.J. Collins. An investigation into face pose distributions. *Automatic Face and Gesture Recognition*, pages 265–270, October 1996.

[35] S. Gong, E.J. Ong, and S. McKenna. Learning to associate faces across views in vector space of similarities of prototypes. *British Machine Vision Conference*, pages 54–63, 1998.

[36] N. Gourier. Extraction de caractéristiques du visage pour estimer la pose. Master's thesis, Institut National Polytechnique de Grenoble, June 2003.

[37] N. Gourier, D. Hall, and J.L. Crowley. Estimating face orientation from robust detection of salient facial features. In *Proceedings of Pointing 2004, ICPR International Workshop on Visual Observation of Deictic Gestures*, pages 17–25, August 2004.

[38] N. Gourier, D. Hall, and J.L. Crowley. Facial feature detection robust to pose, illumination and identity. In *Proceedings of Systems, Man and Cybernetics*, pages 617–622, October 2004.

[39] N. Gourier and J. Letessier. The pointing’04 data sets. In *Proceedings of Pointing 2004, ICPR International Workshop on Visual Observation of Deictic Gestures*, pages 1–4, August 2004.

[40] N. Gourier, J. Maisonnasse, D. Hall, and J. Crowley. Head pose estimation on low resolution images. In *CLEAR Workshop, in Conjunction with Face and Gesture, Southampton, UK*. Springer Verlag, April 2006.

[41] D. Hall. *Viewpoint Independant Object Recognition from Local Appearence*. PhD thesis, Institut National Polytechnique de Grenoble, October 2001.

[42] D. Hall. A system for object class detection. *Cognitive Vision Systems*, 2004.

[43] D. Hall and J. Crowley. Computation of generic features for object classification. *ScaleSpace*, 2003.

[44] I. Haritaoglu, D. Harwood, and L. S. David. Hydra: Multiple people detection and tracking using silhouettes. In *Second IEEE Workshop on Visual Surveillance*, Fort Collins, Colorado, June 1999.

[45] C. Harris and M. Stephens. A combined corner and edge detector. In *Fourth Alvey Vision Conference*, pages 147–151, 1988.

[46] J. Heinzmann and A. Zelinsky. 3-d facial pose and gaze point estimation using a robust real-time tracking paradigm. In *Proceedings of Third International Conference on Automatic Face and Gesture Recognition*. IEEE Computer Society Press, April 1998.

[47] S.Y. Ho and H.L.Huang. An analytic solution for the pose determination of human faces from a monocular image. *Pattern Recognition Letters*, 19:1045–1054, 1998.

[48] T. Horprasert, Y. Yacoob., and L.S. Davisn. Computing 3-d orientation from a monocular image sequence. In *Proceedings of Second International Conference on Automatic Face and Gesture Recognition*, 1996.

[49] E. Horvitz, C. M. Kadie, T. Paek, and D. Hovel. Models of attention in computing and communications: From principles to applications. *Communications of the ACM*, 46(3):52–59, March 2003.

[50] A.J. Howell and H. Buxton. Active vision techniques for visually mediated interaction. *Image and Vision Computing*, 20(12):861–871, 2002.

[51] F.J. Huang, Z. Zhou, H-J. Zang, and T. Chen. Pose invariant face recognition. In *Proceedings of Fourth International Conference on Automatic Face and Gesture Recognition*, pages 245–250. IEEE Computer Society Press, March 2000.

[52] J. Huang, X. Shao, and H. Wechsler. Face pose discrimination using support vector machines (svm). In *Proceedings of 14th International Conference on Pattern Recognition*, pages 154–156, 1998.

[53] K.S. Huang and M.M. Trivedi. Driver head pose and view estimation with single omnidirectionnal video stream. *Third International Conference on Computer Vision Systems*, pages 44–51, April 2003.

[54] D.H. Hubel. *Eye, Brain, And Vision*. Scientific American Library, New York, USA, 1988.

[55] D.P. Huttenlocher and S. Ullman. Recognizing solid objects by alignment with an image. *International Journal of Computer Vision*, 5(2):195–212, 1990.

[56] Polhemus Inc. *FASTRAK*. http://www.polhemus.com.

[57] R. Ishiyama and S. Sakamoto. Fast and accurate facial pose estimation by aligning a 3d appearance model. In *Proceedings of 17th International Conference on Pattern Recognition*, August 2004.

[58] T.S. Jebara and A. Pentland. Parametrized structure from motion to 3d adaptive feedback tracking of faces. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 144–150. IEEE Computer Society Press, 1997.

[59] Q. Ji and R. Hu. 3d face pose estimation and tracking from a monocular camera. *Image Vision Computing*, 20(7):499–511, 2002.

[60] S.C. Johnson. Hierarchical clustering schemes. *Psychometrika*, 32:241–254, 1967.

[61] R.E. Kalman. A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering*, 82(D):35–45, 1960.

[62] J.Y. Kaminski, A. Shavit, D. Knaan, and M. Teicher. Head orientation and gaze detection from a single image. In *Computer Vision Theory and Applications, Setubal, Portugal*, pages 85–92, February 2006.

[63] H. Kawanaka, H. Fujiyoshi, and Y. Iwahori. Human head tracking in three dimensional voxel space. In *Proceedings of ICPR, Hong-Kong*, August 2006.

[64] A. Kendon. Some functions of gaze direction in social interaction. *Acta Psychologica*, 26:22–63, 1967.

[65] D. Kersten, N.F. Troje, and H.H. Bülthoff. Phenomenal competition for poses of the human head. *Perception*, 25:367–368, 1996.

[66] A. Kingstone, C.K. Friesen, and M.S. Gazzaniga. Reflexive joint attention depends on lateralized cortical connections. *Psychological Science*, 11(2):159–166, 2000.

[67] G.J. Klinker, S.A. Shafer, and T. Kanade. A physical approach to color image understanding. *IJVC*, 1990.

[68] J.J. Koenderink and A.J. van Doorn. Representation of local geometry in the visual system. *Biological Cybernetics*, pages 367–375, 1987.

[69] J.J. Koenderink and A.J. van Doorn. Generic neighborhood operators. *PAMI*, 14(6):597–605, June 1992.

[70] T. Kohonen. Associative memory: A system theoretical approach. In *Communication and Cybernetics*. Springer, 1977.

[71] N. Kruger, M. Potzsch, and C. von der Malsburg. Determination of face position and pose with a learned representation based on labeled graphs. *IVC*, 15(8):665–673, 1997.

[72] V. Kruger, S. Bruns, and G. Sommer. Efficient head pose estimation with gabor wavelet networks. *British Machine Vision Conference*, September 2000.

[73] V. Kruger and G. Sommer. Gabor wavelets networks for object representation and face recognition. *Deutsche Arbeitsgemeinschaft fiir Mustererkennung 22 DAGM-Symposium, Kiel*, September 2000.

[74] S.R.H. Langton. The mutual influence of gaze and head orientation in the analysis of social attention direction. *Quarterly Journal of Experimental Psychology: Human Experimental Psychology*, 53A(3):825–845, 2000.

[75] S.R.H. Langton and V. Bruce. Reflexive visual orienting in response to the social attention of others. *Visual Cognition*, 6(5):541–567, 1999.

[76] T.K. Leung, M.C. Burl, and P. Perona. Finding faces in cluttered scenes using random labeled graph matching. *5th International Conference on Computer Vision*, June 1995.

[77] S.Z. Li, Q.D. Fu, L. Gu, B. Scholkopf, Y.M. Cheng, and H.J. Zhang. Kernel machine based learning for multi-view face detection and pose estimation. In *Proceedings of 8th International COnference on Computer Vision*, July 2001.

[78] T. Lindeberg. Feature detection with automatic scale selection. *IJVC*, 30(2):79–116, 1998.

[79] D. Lisin, E. Risemann, and A. Hanson. Extracting salient image features for reliable matching using outlier detection techniques. *Computer Vision Systems Third International Conference*, pages 481–491, April 2003.

[80] R. Lopez and T.S. Huang. 3d head pose computation from 2d images: Templates versus features. *International Conference on Image Processing*, 1995.

[81] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 2004.

[82] D.G. Lowe. Object recognition from local scale-invariant features. *International Conference on Computer Vision*, September 1999.

[83] X. Lu and A.K. Jain. Deformation modeling for robust 3d face matching. In *Computer Vision and Pattern Recognition, New York*, 2006.

[84] B. Ma, W. Zhang, S. Shan, X. Chen, and W. Gao. Robust head pose estimation using lgbp. In *Proceedings of ICPR, Hong-Kong*, August 2006.

[85] J. Maisonnasse, N. Gourier, O. Brdiczka, and P. Reignier. Attentional model for perceiving social context in intelligent environments. In *3rd IFIP Conference on Artificial Intelligence Applications and Innovations, Athens*, June 2006.

[86] J. Maisonnasse, N. Gourier, O. Brdiczka, P. Reignier, and J.L. Crowley. Detecting privacy in attention aware systems. In *Framing the Digital Territories Workshop, in conjunction with Intelligent Environments, Athens*, July 2006.

[87] S. Malassitis and M.G. Strintzis. Real-time head tracking and 3d pose estimation from range data. In *Proceedings of ICIP*, 2003.

[88] M. Malciu and F. Preteux. A robust model-based approach for 3d head tracking in video sequences. In *Proceedings of the Fourth IEEE Int. Conference on Automatic Face and Gesture Recognition*, pages 169–174. IEEE Computer Society Press, 2000.

[89] J. Malik, S. Belongie, and J. Shi T. Leung. Contour and texture analysis for image segmentation. *Proceedings of IJCV*, 43(1):7–27, 2001.

[90] T. Maurer and C. von der Malsburg. Tracking and learning graphs and pose on image sequences of face. In *Proceedings of Face and Gesture*, pages 176–181, 1996.

[91] S. McKenna and S. Gong. Real-time face pose estimation. *International Journal on Real Time Imaging, Special Issue on Real-time Visual Monitoring and Inspection*, 4:333–347, 1998.

[92] S. McKenna, S. Gong, and J.J. Collins. Face tracking and pose representation. *British Machine Vision Conference*, 2:755–764, 1996.

[93] S.J. McKenna, S. Gong, R.P. Wurtz, J. Tanner, and D. Banin. Tracking facial feature points with gabor wavelets and shape models. *Proceedings of the 1st International Conference on Audio- and Videobased Biometric Person Authentication, Lecture Notes in Computer Science*, 1997.

[94] K. Mikolajczyk and C. Schmid. Scale and affine invariant interest point detectors. *International Journal of Computer Vision*, 60(1):63–86, 2004.

[95] D. Miyauchi, A. Sakurai, A. Nakamura, and Y. Kuno. Human-robot eye contact through observations and actions. In *Proceedings of 17th International Conference on Pattern Recognition*, August 2004.

[96] L-P. Morency, P. Sundberg, and T. Darrell. Pose estimation using 3d view-based eigenspaces. In *Proceedings of International Workshop on Analysis and Modeling of Faces and Gestures*. IEEE Computer Society Press, October 2003.

[97] M.C. Motwani. Robust 3d head pose classification using wavelets. Master's thesis, 2003.

[98] M.C. Motwani and Q. Ji. 3d face pose discrimination using wavelets. In *Proceedings of ICIP*, October 2001.

[99] M. A. Nacenta, S. Sallam, B. Champoux, S. Subramanian, and C. Gutwin. Perspective cursor: Perspective-based interaction for multi-display environments. In *CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems*, pages 289–298, New York, NY, USA, 2006. ACM Press.

[100] A. Negre, H. Tran, N.Gourier, D. Hall, A. Lux, and J.L. Crowley. Object recognition invariant to viewpoint. Caviar deliverable, Institut National Polytechnique de Grenoble, 2005.

[101] A. Negre, H. Tran, N.Gourier, D. Hall, A. Lux, and J.L. Crowley. Comparative study of people detection in surveillance scenes. In *Structural and Syntactic Pattern Recognition Workshop, in Cunjunction with ICPR, Hong Kong*. Springer Verlag, August 2006.

[102] J. Ng and S. Gong. Multi-view face detection and pose estimation using a composite support vector machine across the view sphere. In *Proceedings of International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems*, pages 14–21, September 1999.

[103] A. Nikolaidis and I. Pitas. Facial feature extraction and pose determination. *International Conference on Pattern Recognition*, 33:1783–1791, 2000.

[104] S. Niyogi and W.T. Freeman. Example-based head tracking. In *Proceedings of Second International Conference on Automatic Face and Gesture Recognition*, 1996.

[105] S. Ohayon and E. Rivlin. Robust 3d head tracking using camera pose estimation. In *Proceedings of ICPR, Hong-Kong*, August 2006.

[106] E-J. Ong, S.J. McKenna, and S. Gong. Tracking head pose for inferring intention. *European Workshop on Perception of Human Action, Freiburg*, June 1998.

[107] T. Otsuka and J. Ohya. Real-time estimation of head motion using weak perspective epipolar geometry. In *Proceedings of WACV*, October 1998.

[108] C.A. Perez, V.A. Lazcano, P.A. Estevez, and C.M. Held. Real-time iris detection on faces with coronal axis rotation. In *Proceedings of Systems, Man and Cybernetics*, October 2004.

[109] G. Peters. *A system for Object Class Detection*. PhD thesis, Aachen, Germany, 2002.

[110] Claude Prablanc. Cours de neurophysiologie 2003/2004.

[111] F. Preteux and M.Malciu. Model-based head tracking and 3d pose estimation. In *Proceedings of SPIE Conference on Mathematical Modeling and Estimation Techniques in Computer Vision*, pages 94–110, July 1998.

[112] R. Rae and H. Ritter. Recognition of human head orientation based on artificial neural networks. *IEEE Conference on Computer Vision and Pattern Recognition*, 9(2):257–265, 1998.

[113] R.P.N. Rao and D.H. Ballard. An active vision architecture based on iconic representations. *Artificial Intelligence*, 78(1-2):461–505, 1995.

[114] B. Schiele. *Object Recognition using Multidimensional Receptive Field Histograms*. PhD thesis, Institut National Polytechnique de Grenoble, Grenoble, July 1997.

[115] B. Schiele and J. Crowley. Object recognition without correspondence using multidimensionnal receptive fields histograms. *International Journal of Computer Vision*, 36(1):31–50, 2000.

[116] B. Schiele and A. Waibel. Gaze tracking based on face-color. In *Proceedings of International Workshop on Automatic Face and Gesture Recognition*, pages 344–349, June 1995.

[117] C. Schmid. *Appariement d’images par invariants locaux de niveaux de gris*. PhD thesis, Institut National Polytechnique de Grenoble, 1996.

[118] C. Schmid. Constructing models for content-based image retrieval. In *Proceedings of Conference on Computer Vision and Pattern Recognition*. IEEE Computer Society Press, 2001.

[119] K. Schwerdt and J. Crowley. Robust face tracking using color. In *Proceedings of Fourth International Conference on Automatic Face and Gesture Recognition*, pages 90–95. IEEE Computer Society Press, March 2000.

[120] SeeingMachines. *FaceLAB*, 2000.

[121] J. Sherrah and S. Gong. Fusion of perceptual cues using covariance estimation. *British Machine Vision Conference*, 2:564–573, September 1999.

[122] J. Sherrah, S. Gong, and E-J. Ong. Understanding pose discrimination in similarity space. *British Machine Vision Conference*, 1999.

[123] I. Shimizu, Z. Zang, S. Akamatsu, and K. Deguchi. Head pose determination from one image using a generic model. In *Proceedings of Third International Conference on Automatic Face and Gesture Recognition*, pages 100–105. IEEE Computer Society Press, April 1998.

[124] D. Sinley. Laser and led eye hazard : Safety standard. *Optics Photonics News*, pages 32–37, 1997.

[125] D. Slepian and H.O. Pollack. Prolate spheroidal waveforms fourier analysis and uncertainty. *Bell Systems Technical Journal*, 40(1):43–63, January 1961.

[126] P. Smith, M. Shah, and N. da Vitoria Lobo. Determining driver visual attention with one camera. *Transactions on Intelligent Transportation Systems*, 4(4):205–218, 2003.

[127] S. Srinivasan and K. L. Boyer. Head pose estimation using view based eigenspaces. *International Conference on Pattern Recognition*, 4:302–305, August 2002.

[128] B. Steinzor. The spatial factor in face to face discussions. *Journal of Abnormal and Social Psychology*, 45:552–555, 1950.

[129] R. Stiefelhagen. *Tracking and Modeling Focus of Attention in Meetings*. PhD thesis, Universitat Karlsruhe, 2002.

[130] R. Stiefelhagen. Tracking focus of attention in meetings. *International Conference on Multimodal Interfaces*, pages 273–280, October 2002.

[131] R. Stiefelhagen. Estimating head pose with neural networks - results on the pointing04 icpr workshop evaluation data. In *Proceedings of Pointing 2004, ICPR International Workshop on Visual Observation of Deictic Gestures*, August 2004.

[132] R. Stiefelhagen, M. Finke, J. Yang, and A. Waibel. From gaze to focus of attention. *Visual Information and Information Systems*, pages 761–768, 1999.

[133] R. Stiefelhagen, J. Yang, and A. Waibel. A model-based gaze tracking system. *International Journal of Artificial Intelligence Tools*, 6:193–209, 1997.

[134] R. Stiefelhagen, J. Yang, and A. Waibel. Tracking eyes and monitoring eye gaze. In *Proceedings of the Workshop on Perceptual User Interfaces*, October 1997.

[135] R. Stiefelhagen, J. Yang, and A. Waibel. Modeling people's focus of attention. *ACM Multimedia*, October 1999.

[136] R. Stiefelhagen, J. Yang, and A. Waibel. Simultaneous tracking of head poses in a panoramic view. *International Conference on Pattern Recognition*, September 2000.

[137] R. Stiefelhagen, J. Yang, and A. Waibel. Tracking focus of attention for human-robot communication. *IEEE-RAS International Conference on Humanoid Robots Humanoids*, November 2001.

[138] R. Stiefelhagen and J. Zhu. Head orientation and gaze direction in meetings. *Conference on Human Factors in Computing Systems*, April 2002.

[139] M. Storing. *Computer Vision and Human Skin Color*. PhD thesis, Aalborg University, 2004.

[140] K-K. Sung and T. Poggio. Example-based learning for view-based human face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):39–51, 1998.

[141] K. Tanaka. Mechanisms of visual object recognition: Monkey and human studies. *Current opinion in Neurobiology*, 7:523–529, 1997.

[142] K. Toyama. Hands-free cursor control with real-time 3d face tracking. In *Proceedings of Workshop on Perceptual User Interfaces (PUI'98)*. IEEE Computer Society Press, November 1998.

[143] K. Toyama. Prolegomena for robust face tracking. Technical Report MSR-TR-98-65, Presented at the Post-ECCV Workshop on Advances in Facial Image Analysis and Recognition Technology, May 1998.

[144] T.T.H. Tran. *Étude des lignes naturelles pour la représentation d'objets en vision par ordinateur*. PhD thesis, Institut National Polytechnique de Grenoble, 2006.

[145] J. Tu, Y. Fu, Y. Hu, and T. Huang. Evaluation of head pose estimation for studio data. In *CLEAR Workshop, in Conjunction with Face and Gesture, Southampton, UK*. Springer Verlag, April 2006.

[146] M. Turk and A. Pentland. Eigenfaces for recognition. *Cognitive Neuroscience*, 3(1):71–96, 1991.

[147] L. Vacchetti, V. Lepetit, and P. Fua. Stable real-time 3d tracking using online and offline information. *Pattern Analysis and Machine Intelligence*, 26(10):1385–1391, 2004.

[148] D. Valentin, H. Abdi, and A. O'Toole. Categorization and identification of human face images by neural networks: A review of linear auto-associator and principal component approaches. *Journal of Biological Systems*, 2:413–429, 1994.

[149] A.C. Varchmin, R. Rae, and H. Ritter. Image based recognition of gaze direction using adaptative methods. In *Proceedings of International Gesture Workshop*, pages 245–257. Springer Verlag, 1997.

[150] R.C. Verma, C. Schmid, and K. Mikolajczyk. Face detection and tracking in a video by propogating detection probabilities. *Pattern Analysis and Machine Intelligence*, 25(10), October 2003.

[151] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings IEEE Conf. on Computer Vision and Pattern Recognition*, 2001.

[152] M. Voit, K. Nickel, and R. Stiefelhagen. Neural network-based head pose estimation and multi-view fusion. In *CLEAR Workshop, in Conjunction with Face and Gesture, Southampton, UK*. Springer Verlag, April 2006.

[153] K.N. Walker, T.F. Cootes, and C.J Taylor. Automatically building appearance models from image sequences using salient features. *IVC*, 20(5-6):435–440, April 2002.

[154] C. Wang and M. Brandstein. Robust head pose estimation by machine learning. In *Proceedings of ICIP 3*, pages 210–213, 2000.

[155] J.G. Wang and E. Sung. Pose determination of human faces by using vanishing points. *Pattern Recognition*, 34(12):2427–2445, December 2001.

[156] Y. Wang, Y. Liu, L. Tao, and G. Xu. Real-time multi-view face detection and pose estimation in video stream. In *Proceedings of ICPR, Hong-Kong*, August 2006.

[157] Y. Wei, L. Fradet, and T. Tan. Head pose estimation using gabor eigenspace modeling. *IEEE ICIP*, 1:281–284, 2002.

[158] L. Wiskott, J-M. Fellous, N. Kruger, and C. von der Malsburg. Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19:775–779, 1997.

[159] J. Wu, J.M. Pedersen, D. Putthividhya, D. Norgaard, and M.M. Trivedi. A two-level pose estimation framework using majority voting of gabor wavelets and bunch graph analysis. In *Proceedings of Pointing 2004, ICPR International Workshop on Visual Observation of Deictic Gestures*, August 2004.

[160] J. Wu and M. Trivedi. An integrated two-stage framework for robust head pose estimation. In *Proceedings of Analysis and Modeling of Face and Gesture*, pages 321–335, 2005.

[161] Y. Wu and K. Toyama. Wide-range, person- and illumination-insensitive head orientation estimation. In *Proceedings of Fourth International Conference on Automatic Face and Gesture Recognition*, pages 183–188. IEEE Computer Society Press, March 2000.

[162] M. Xu and T. Akatsuka. Detecting head pose from stereo image sequence for active face recognition. In *Proceedings of Third International Conference on Automatic Face and Gesture Recognition*, pages 82–87. IEEE Computer Society Press, April 1998.

[163] J. Yang, R. Stiefelhagen, U. Meier, and A. Waibel. Visual tracking for multimodal human computer interaction. *Human Factors in Computing Systems: CHI*, pages 140–147, April 1998.

[164] P. Yao, G. Evans, and A. Calway. Using affine correspondance to estimate 3-d facial pose. In *Proceedings of 8th ICIP*, pages 919–922, 2001.

[165] A.L. Yarbus. Eye movements during perception of complex objects. *Eye Movements and vision, Plenum Press NYC*, pages 171–196, 1967.

[166] L. Zhao. *Dressed Human Modeling, Detection, and Part Localization*. PhD thesis, The Robotics Institute Carnegie Mellon University, 2001.

[167] Z. Zhu and Q. Ji. 3d face pose tracking from an uncalibrated monocolar camera. In *Proceedings of 17th International Conference on Pattern Recognition*, August 2004.

[168] Z. Zhu and Q. Ji. Robust real-time face pose and facial expression recovery. In *Computer Vision and Pattern Recognition, New York*, 2006.

[169] M. Zobel, A. Gebhard, D. Paulus, J. Denzler, and H. Niemann. Robust facial feature localization by coupled features. In *Proceedings of Fourth International Conference on Automatic Face and Gesture Recognition*, pages 2–7. IEEE Computer Society Press, March 2000.