Universitat Autònoma
de Barcelona

Looking at Faces:

Detection, Tracking and Pose Estimation

A dissertation submitted by **Murad Al Haj** at Universitat Autònoma de Barcelona to fulfill the degree of **Doctor en Informàtica**.

Bellaterra, December 2012

Director	**Dr. Jordi Gonzàlez i Sabaté** Centre de Visió per Computador & Dept. de Ciències de la Computació. Universitat Autònoma de Barcelona.
Co-director	**Dr. F. Xavier Roca i Marva** Centre de Visió per Computador & Dept. de Ciències de la Computació. Universitat Autònoma de Barcelona.

This document was typeset by the author using LaTeX 2_ε.

The research described in this book was carried out at the Computer Vision Center, Universitat Autònoma de Barcelona.

ISBN 978-84-940530-5-4

Printed by Ediciones Gráficas Rey, S.L.

To Mustafa, Souha, Haitham, Noura, Maya
and the memory of Maha.

"Well, that's Philosophy I've read,
And Law and Medicine, and I fear
Theology, too, from A to Z;
Hard studies all, that have cost me dear.
And so I sit, poor silly man
No wiser now than when I began."

Goethe.

Acknowledgments

Starting to write this section, I feel overwhelmed by a sense of appreciation to many great people whose help was instrumental in the completion of this thesis.

I would like to start by expressing my sincere gratitude to my advisers Dr. Jordi Gonzàlez and Dr. Xavier Roca. Working with them has been a very rewarding experience and their continuous support has been essential in my professional growth. I want to also thank Dr. Juan José Villanueva for welcoming me into the research center he established and the group he directed, and Dr. Andrew Bagdanov for supervising some of the work presented in this thesis, in addition to the valued roles he played over the years.

Thanks to Dr. Angel Sappa for accepting to be on the board of examiners and for all his support. Much gratitude is due to him and the rest of thesis committee: Dr. Thomas Moeslund and Dr. Arantxa Villanueva for allocating time in their busy schedules to evaluate this thesis and sit on the committee.

This journey would have been much harder had it not been shared with some amazing friends. Thanks to Ariel for his slightly pessimistic realism that proved valuable many times, for Marco's happy-go-lucky approach to life, and Bhaskar's ability to perceive everyday events as a magical phenomena. I am thankful to them, and Naveen, for being there whenever I needed a friend to talk to, or a place to stay.

I want to also thank the rest of the ISE group members, former and current, for all the knowledge-sharing, discussions and Tuesday lunches. Also, thanks to Dr. Joost van Weijer for many interesting conversations. Equal appreciation goes to Montse and Gigi as well as the rest of the CVC administrative staff for saving me from bureaucratic nightmares.

I also acknowledge the support from AGAUR, Generalitat de Catalunya through an FI predoctoral scholarship and a BE mobility grant, in addition to the support from the Spanish Ministry of Education through another mobility grant.

For the advancement of one's research and the development of his/her ideas, no catalyst is better than visiting great labs with brilliant people, and I have been lucky enough to have my share of those visits. I am sincerely grateful to Dr. Jan-Mark Geusebroek for hosting me at the University of Amsterdam and his influential advice that helped me establish many building blocks for a research career.

I owe much of my professional development to Prof. Larry Davis at the University of Maryland. It has been a privilege and an honor to work with a renowned researcher of his caliber. For all the time he allocated to our discussions and all the mentoring he generously provided, no words can express my gratitude.

Preparing for the next chapter of my life, I am indebted to Doug Carmean, my manager at Intel, who provided me with a unique opportunity to work on a very cool technology. He introduced me to a new world and taught many important lessons: actively by working with him on different parts of the technology, and passively by observing his exceptional ability to solve challenging problems and handle difficult situations. Equal gratitude goes to Carl Marshall for his support and day-to-day mentoring. He was there whenever I needed guidance and his advice was always spot-on. It has been a pleasure to work with both of them over the last year and I am looking forward to rejoining them soon.

If I am to single out a unique person behind my (humble) achievements, it would be my uncle, Dr. Haitham Akkary, to whom I will be eternally grateful. I chose this course following his path, and he supported me in more ways than one can imagine. His intellect and generosity have always been the wind that drives me to better shores.

Many thanks to my father and role model Mustafa, a great man who taught me, by example, that honesty and integrity are not mere words but a sacred path that must be followed with utter disregard to personal gain. Many thanks to my mother Souha, an amazing woman who never hesitated to sacrifice anything she had for the sake of her children. Many thanks to my second mother, my late aunt Maha, a saint who cared for and carried so many people till the weight took the best of her. Many thanks to my dear Noura and Maya, no brother can wish for more awesome sisters. My family endless love is what kept me going through all those years. In a gratitude inversely proportional to the number of words: love you.

Abstract

Humans can effortlessly perceive faces, follow them over space and time, and decode their rich content, such as pose, identity and expression. However, despite many decades of research on automatic facial perception in areas like face detection, expression recognition, pose estimation and face recognition, and despite many successes, a complete solution remains elusive. Automatic facial perception encompasses many important and challenging areas of computer vision and its applications span a very wide range; these applications include video surveillance, human-computer interaction, content-based image retrieval, biometric identification, video coding and age/gender recognition. This thesis is dedicated to three problems in automatic face perception, namely face detection, face tracking and pose estimation.

In face detection, an initial simple model is presented that uses pixel-based heuristics to segment skin locations and hand-crafted rules to return the locations of the faces present in the image. Different colorspaces are studied to judge whether a colorspace transformation can aid skin color detection. Experimental results show that the separability does not increase in other colorspaces when compared to the RGB space. The output of this study is used in the design of a more complex face detector that is able to successfully generalize to different scenarios.

In face tracking, we present a framework that combines estimation and control in a joint scheme to track a face with a single pan-tilt-zoom camera. An extended Kalman filter is used to jointly estimate the object world-coordinates and the camera position. The output of the filter is used to drive a PID controller in order to reactively track a face, taking correct decisions when to zoom-in on the face to maximize the size and when to zoom-out to reduce the risk of losing the target. While this work is mainly motivated by tracking faces, it can be easily applied atop of any detector to track different objects. The applicability of this method is demonstrated on simulated as well as real-life scenarios.

The last and most important part of this thesis is dedicate to monocular head pose estimation. In most prior work on heads pose estimation, the positions of the faces on which the pose is to be estimated are specified manually. Therefore, the results are reported without studying the effect of misalignment. Regression, as well as classification, algorithms are generally sensitive to localization error. If the object is not accurately registered with the learned model, the comparison between the object features and the model features leads to errors. In this chapter, we propose a method based on partial least squares regression to estimate pose and solve the alignment problem simultaneously. The contributions of this part are two-fold: 1) we

show that the proposed method achieves better than state-of-the-art results on the estimation problem and 2) we develop a technique to reduce misalignment based on the learned PLS factors that outperform multiple instance learning (MIL) without the need for any re-training or the inclusion of misaligned samples in the training process, as normally done in MIL.

Resumen

Los seres humanos pueden percibir muy fácilmente las caras, las pueden seguir en el espacio y tiempo, así como decodificar su contenido, como su postura, identidad y expresión. Sin embargo, a pesar de muchas décadas de investigación para desarrollar un sistema con percepción automática de caras, una solución completa sigue siendo difícil de alcanzar en áreas como la detección de caras, el reconocimiento de la expresión facial, la estimación de la posición o el reconocimiento del rostro. Esto es debido a que la percepción facial automática involucra muchas áreas importantes y difíciles de la visión por computador, cuyas aplicaciones finales abarcan una gama muy amplia como la video vigilancia, interacción humano-computadora, la indexación y recuperación del contenido de imágenes, la identificación biométrica, la codificación de vídeo y el reconocimiento de la edad y/o sexo. En particular, esta tesis está dedicada a tres grandes problemas en la percepción automática de caras: la detección de rostros, el seguimiento de caras y la estimación de la posición facial.

En el campo de la detección de rostros, se presenta un modelo que utiliza múltiples heurísticas sencillas ad-hoc basadas en píxeles para detectar las regiones de la imagen correspondientes a piel humana. Además, se han estudiado diferentes espacios de color para determinar si existe alguna transformación de espacio de color que puede mejorar la detección del color de la piel. Los resultados experimentales muestran que la separabilidad no aumenta demasiado en otros espacios de color en comparación con la obtenida en el espacio RGB. A partir del mejor espacio de color, se ha diseñado un detector de caras capaz de generalizar en diferentes escenarios con éxito.

Como segunda aportación, se ha desarrollado un algoritmo para el seguimiento robusto y preciso de la cara, dentro de un marco unificado que combina la estimación de los parámetros faciales con el control de una cámara activa, para el seguimiento de caras mediante una cámara Pan-Tilt-Zoom. Un filtro de Kalman extendido permite estimar conjuntamente las coordenadas mundo de los objetos así como la posición de la cámara. La salida se utiliza para accionar un controlador PID con el fin de realizar un seguimiento reactivo del rostro, generando las acciones de control correctas no solo para mantener un zoom-in en la cara para maximizar el tamaño, sino también para poder alejarse y reducir el riesgo de perder el objetivo. Aunque este trabajo está principalmente motivado para realizar un seguimiento de caras, se puede aplicar fácilmente como ayuda de un detector de objetos para rastrear una escena con una cámara activa. La aplicabilidad del método se ha demostrado tanto en entornos simulados como en escenarios reales.

Se ha dedicado la última y más importante parte de esta tesis a la estimación de la postura de la cabeza. En la mayoría de trabajos previos para la estimación de la posición de la cabeza, se especifica manualmente las caras. Por tanto, los resultados detallados no tienen en cuenta una posible desalineación de la cara, aunque tanto en regresión como en clasificación, los algoritmos son generalmente sensibles a este error en localización: si el objeto no está bien alineado con el modelo aprendido, la comparación entre las características del objeto en la imagen y las del modelo conduce a errores. En este último capítulo, se propone un método basado en regresión por mínimos cuadrados parciales para estimar la postura y además resolver la alineación de la cara simultáneamente. Las contribuciones en esta parte son de dos tipos: 1) se

muestra que el método propuesto alcanza mejores resultados que el estado del arte y 2) se desarrolla una técnica para reducir la desalineación basado en factores PLS que mejoran el aprendizaje basado en múltiples instancias sin la necesidad de re-aprender o tener que incluir muestras mal alineadas, ambos normalmente necesarios en el aprendizaje basado en múltiples instancias.

Resum

Els éssers humans podem percebre molt fàcilment les cares, les podem seguir en l'espai i temps, així com descodificar el seu contingut, com la seva postura, identitat o expressió. No obstant això, tot i moltes dècades d'investigació per desenvolupar un sistema amb percepció automàtica de cares, segueix sent difícil d'aconseguir una solució completa en àrees com la detecció de cares, el reconeixement de l'expressió facial, la estimació de la posició o el reconeixement de la cara. Això és degut a que la percepció facial automàtica abasta moltes àrees importants i difícils de la visió per computador: les aplicacions finals abasten una gamma molt àmplia com la vídeo vigilància, interacció humà-ordinador, la indexació i recuperació del contingut d'imatges, la identificació biomètrica , la codificació de vídeo i el reconeixement de l'edat i / o sexe. En particular, aquesta tesi està dedicada a tres grans problemes en la percepció automàtica de cares: la detecció de rostres, el seguiment de cares i l'estimació de la posició facial.

En el camp de la detecció de rostres, es presenta un model que utilitza múltiples heurístiques senzilles ad-hoc basades en píxels per detectar les regions de la imatge corresponents a pell humana. A més, s'han estudiat diferents espais de color per determinar si hi ha alguna transformació d'espai de color que pugui millorar la detecció del color de la pell. Els resultats experimentals mostren que la separabilitat no augmenta gaire en altres espais de color en comparació amb l'obtinguda en l'espai RGB. A partir del millor espai de color trobat, s'ha dissenyat un detector de cares capaç de generalitzar amb èxit en diferentes escenes.

Com a segona aportació, s'ha desenvolupat un algorisme per al seguiment robust i precís de la cara, dins d'un marc unificat que combina l'estimació dels paràmetres facials amb el control d'una càmera activa, per al seguiment de cares mitjançant una càmera Pa- Tilt-Zoom. Un filtre de Kalman estès permet estimar conjuntament les coordenades món dels objectes i la posició de la càmera. La sortida s'utilitza per accionar un controlador PID per tal de realitzar un seguiment reactiu del rostre, generant les accions de control correctes no només per mantenir un zoom-in a la cara per maximitzar la mida, sinó també per poder allunyar i reduir el risc de perdre l'objectiu. Encara que aquest treball està principalment motivat per fer un seguiment de cares, es pot aplicar fàcilment com ajuda d'un detector d'objectes per rastrejar una escena amb una càmera activa. L'aplicabilitat del mètode s'ha demostrat tant en entorns simulats com a escenaris reals.

S'ha dedicat l'última i més important part d'aquesta tesi a l'estimació de la posició del cap. En la majoria de treballs previs per a l'estimació de la posició del cap, s'especifiquen les cares manualment. Per tant, els resultats detallats no tenen en compte una possible desalineació de la cara, encara que tant en regressió com en classificació, els algoritmes són generalment sensibles a un error en localització: si l'objecte no està ben alineat amb el model après, la comparació entre les característiques de l'objecte en la imatge i les del model condueix a errors. En aquest últim capítol, es proposa un mètode basat en regressió per mínims quadrats parcials per estimar la posició i a més resoldre simultàniament l'alineació de la cara. Les contribucions en aquesta part són de dos tipus: 1) es mostra que el mètode proposat assoleix millors resultats que l'estat de l'art i 2) es desenvolupa una tècnica per reduir la desalineació

basat en factors PLS que milloren l'aprenentatge basat en múltiples instàncies sense la necessitat de tornar a aprendre o d'haver d'incloure mostres mal alineades, ambdós pasos normalment necessaris en l'aprenentatge basat en múltiples instàncies.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

"The serial number
of a human specimen
is the face."

Milan Kundera.

Face perception is perhaps the most highly developed visual sensing skill in humans. Infants are born preprogrammed with primitive facial processing "features". Experiments show that newborn infants, only hours old, are capable of discriminating between faces, showing a clear preference for their mother's. As young as two days old, they can mimic the facial gestures modeled by an adult. In addition, reports show that the newly-born are capable of reproducing mouth opening and tongue protrusion from the first face they ever see. From that age onwards, humans spend more time looking at faces than any other object and they are capable of distinguishing between virtually unlimited number of unique faces [102].

One's face is the fingerprint of his existence, or as Milan Kundra puts it "his serial number". Therefore, face perception is highly coupled with extracting semantic information about the observed individual; the appearance of a familiar face triggers a series of events in the brain, ending with the retrieval of the name. However, recognizing identity is not the main motivation for looking at faces. Face viewing happens within a much broader context of social communications. People, being able to quickly and effortlessly detect faces, use this information to associate themselves with other humans as well as infer understanding of the environment in which they are present. An expression present on one's face provides information about his/her emotional state in addition to the associated social context, i.e. the perception of fear on surrounding faces can alert the observer to an imminent danger. Even further, face perception plays a key role in speech recognition where lip reading improves audio sensing while incoherency between lip motion and audible speech causes hearing errors [42].

Therefore, face processing is different than processing any other visual stimuli and it is served by a dedicated neural system. The existence of a dedicated network explains how certain brain lesions hinder the capacity of recognizing known faces while

leaving the recognition of other objects intact. However, in this network, a distinction exists between the representation of changeable aspects of a face, e.g. expression or head/gaze pose, and the invariant features specifying identity [12]. Without this distinction, a smiling face would be misinterpreted as having a new identity compared to the same face while frowning.

1.1 Motivation

A discovery was reached by Mozi in China in the 5^{th} century B.C., and less than a century later by Aristotle in Greece. They both noted that when light from a scene passes through a small hole into a chamber, it forms an inverted image of the scene on the chamber's opposite wall. These observations where the first reported instances of the pinhole camera; however, the mechanisms behind this effect where not properly described until the 10^{th} century. From 1011 to 1021 AD, the Arabic scholar Ibn al-Haytham, known to western scholars as Alhazen, published Kitab al-Manazir (Book of Optics), in which he correctly described the pinhole phenomenon, among many other important contributions. The book, published in 7 volumes, had a huge impact on the understanding of visual perception and optics [39].

Fast forwarding to the 19^{th} century, a lens replaced the small hole and light sensitive paper substituted for the projection surface and the first "modern-day" camera was created. In the 20^{th} century, electronic light sensors replaced the light sensitive paper resulting in digital cameras. Therefore, it is no surprise that the word camera comes from the Latin "camera obscura" meaning dark chamber.

With the dawn of computer age, it was only natural to attempt at taking this ability of image capturing into the logical next level: automatic analysis of the acquired images and understanding the scenes behind them. Thus, computer vision was born.

Starting in the early 70's, computer vision was considered a first layer towards a more complex human intelligence modeling. Solving the perception layer or "visual input" problem was thought of as a simple preliminary step towards "higher-level reasoning and planning". The difficulty of the problem was so underestimated that MIT's Marvin Minsky told one of his undergraduate students, in 1966, to "spend the summer linking a camera to a computer and getting the computer to describe what it saw" [109]. Till the time of writing of this thesis, and despite huge advances since the 60's, one can safely say that this problem is not yet solved.

The difficulty of the computer vision problem(s) reflects the complexity of the human visual system, of which face perception is perhaps the most central, and equivalently the most complex. Therefore, it is no surprise that the very early work of object recognition by matching pictorial structures in 1973, [28], chose faces as exemplary objects to detect in a "sensed scene". The authors modeled the face as the structure in figure 1.1 and tried to fit this "reference" to the observed images. Automatic face recognition also dates back to the early 70's [57, 56].

Work on automatic localization of faces and detection of their invariant characteristics (such as identity and facial features) alongside their changeable aspects (such as pose and expression) continues to the current day motivated by the diverse and valuable applications that benefit from it. The value of computational face perception

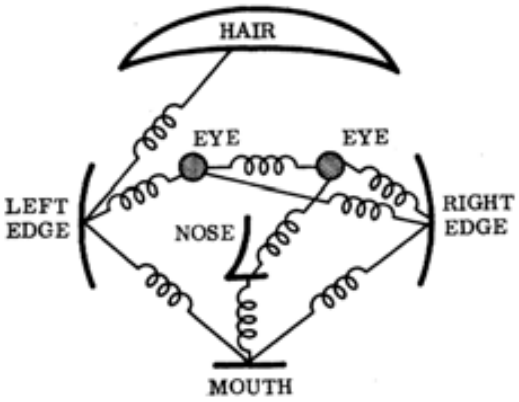


Figure 1.1: Face pictorial structure [28].

in areas like human computer interaction (HCI), video surveillance, computer aided living and virtual reality is direct and obvious.

This thesis is dedicated to faces and their automatic perception. As the title indicates, the research was done along three lines: detection, tracking, and pose estimation. The contributions of this thesis and its outline are highlighted in the next section.

1.2 Contributions

In our work, looking at faces in scenes, the following contributions were achieved:

- **In Face Detection.** A new face detection algorithm was developed. Skin color segmentation reduced the search space and shape features decided whether a given skin blob is a face or not. This resulted in a more efficient search than the traditional sliding window. As a side note, the fact that the human brain treat faces differently than any other object, makes the case for dedicated face detectors that are not necessarily generic object detectors.

- **In Reactive Tracking.** A technique for reactive object tracking using a Pan-Tilt-Zoom (PTZ) camera was implemented, trying to establish a trade-off between the resolution per target and the area of coverage through a joint estimation of the 3D object position and the camera position performed by an extended Kalman filter. While this method was motivated by the importance of tracking a face over a series of consecutive frames, as a preliminary step for any social inference, it can be applied to any object as long as its detector is available.

- **In Head Pose Estimation.** The most prominent part of this thesis is the work

on head pose estimation. We proposed a method based on partial least squares (PLS) regression to estimate pose and simultaneously solve the alignment problem, which normally results from localization errors between an instance of a certain object class and its trained model. In this work, we show that the kernel version of PLS (kPLS) outperforms state-of-the-art methods on the estimation problem and we develop a technique to reduce misalignment based on the learned PLS factors.

1.3 Research Evolution

In his plenary talk in ICPR 2010, Prof. Christopher Bishop gave an overview of the transformation that the machine intelligence field has been undergoing during the last few decades, highlighting three generations. The first generation, which started in the 60's and ended in the 80's, depended on human experts to define hand crafted rules describing the system, of which the pictorial structures approach mentioned earlier is a clear example. The second generation, which started in the 90's and continues up to the present day, abandoned expert rules for solutions learned from data. Statistical models, such as neural networks and support vector machines, are heavily used in this generation. He called those "black-box statistical models" demonstrating that they can not incorporate expert knowledge per se.

Prof. Bishop went on to show the importance of domain knowledge stating that the aim of third generation machine learning is to integrate this knowledge with the statistical learning methods. The three key ideas here are: 1) The use of Bayesian learning to model uncertainties and update them upon the arrival of new knowledge, 2) the use of graphical models that are suitable for representing domain knowledge, giving examples such as principal component analysis, Kalman filters and hidden Markov Models, and 3) Efficient inference methods.

The work in this thesis had followed a similar evolution. In the initial attempt at face detection, hand crafted rules were applied to classify skin vs. non-skin and face vs. non-face, resulting in a first generation face detector. While working very well on some sequences, especially those shot indoor, adapting the model to generalize in different scenarios proved very challenging. Exception after exception needed to be hard coded in the model. At this point, the tyranny of the data prevailed and statistical black-boxes were used to do their magic on the designed features, creating a second generation detector. Following this evolution, concepts belonging to the third generation machine intelligence are embodied through incorporating uncertainty and using Kalman filtering in our reactive tracking. Similarly, the domain specific latent spaces and the efficient regression in our pose estimation are further examples of those third generation concepts.

1.4 Thesis Outline

The remainder of this thesis is organized as follows. Chapter 2 demonstrates our first and second generation face detectors, providing an argument about the effect of colorspace selection on skin color segmentation. Chapter 3 is dedicated to our reactive

tracking method which aims at obtaining a trade-off between high resolution of the tracked object and minimizing the risk of losing it. A joint estimation of the 3D object position and the camera position is performed by an extended Kalman filter and the output is used to drive the PTZ camera. Chapter 4, the most important chapter of this thesis, presents a PLS-based regression method for head pose estimation that significantly reduces sensitivity to misalignment. The method outperforms state-of-the-art methods while simultaneously dealing with misaligned faces. Chapter 5 concludes this thesis summarizing the work present.

Chapter 2

Face Detection: A First and Second Generation Models

"There is nothing worth thinking
but it has been thought before;
we must only try to think it again."

Goethe.

Face detection is an important first step in many applications of computer vision. Of these applications, face recognition, video surveillance, human computer interaction, content-based image retrieval, video coding and expression recognition have attracted much interest lately. The more accurately the detector performs, the less post-processing will be needed and the better the aforementioned applications will function.

However, face detection is an expensive search problem. To properly localize a face in an image, all regions should be searched at varying scales. This is usually done through a sliding window model and naturally produces many more windows corresponding to background objects rather than faces. In such a scan scheme, the ratio of non-face regions to actual face regions can be in the order of 100000:1 [108]. This high ratio calls for a very well trained classifier that will produce a low number of false positives, otherwise the performance of the many applications depending on face detection will normally suffer. Furthermore, within-class variations of non-rigid objects like faces make detection a challenging problem. Those variations are nicely summarized in [133], including the following:

- **Pose variation.** Faces can be frontal, profile, upside-down, etc., depending on the relative position between the camera and the face. This greatly affect the face image where certain features are present in one pose but absent in another, e.g. two eyes appear in frontal poses but only one in profile poses.

This chapter is based on work published in CORES07 [2], ICPR08 [7] and IbPRIA09 [3].

- **Presence or Absence of Structural Components.** These include beards, mustaches and glasses, and they are extremely variable in terms of shape, color and size.

- **Facial expressions.** One's facial expression can highly impact the appearance of his face.

- **Occlusion.** Faces can be partially occluded by different objects present in the scene, including other faces.

- **Imaging conditions.** Face images can be highly impacted by the lighting conditions, in terms of spectra, source distribution and intensity, as well as by the camera specifications, such as lenses and sensors.

Face detection has been extensively studied. Early work, which focused on upright frontal face detection, include that of Sung and Paggio [86] where the difference between the local image pattern and a distribution-based model was computed and used to build a classifier. Papageorgiou used an over-complete wavelet representation of faces to achieve detection [83]. One noteworthy technique in the late 90's is that of Rowley et al. [94] which used trained neural networks to detect frontal faces. However, the most cutting-edge face detection algorithm in the 2000's is that of Viola and Jones [114] which uses Haar-like features followed by Adaboost, making real-time robust face detection possible. Some attempts were made to extend this method to multiview faces as in [53, 47]. However, and while upright frontal faces can be detected with high accuracy and speed using present methods, fast multiview face detection remains an open challenge. This is mainly due to the fact that the techniques aiming at multipose detection are either computationally expensive, such as the previously mentioned paper [47], or produce a large number of false positives [119], or both.

In this chapter, our work towards an efficient and robust multiview face detection method is presented and the evolution of the research is highlighted. Our detection is based on segmenting skin blobs, then shape features are used to judge whether a given blob is a face or not. The rest of the chapter is organized as follows: section 2.1 surveys face detection algorithms, section 2.2 describes our early attempt resulting in a first generation detection model, section 2.3 makes the case for the use of RGB colorspace in skin color segmentation versus other colorspaces that separate intensity/luminance from chromaticity, section 2.4 presents our second generation face detector and shows that it can generalize to unseen scenarios, and section 2.5 ends this chapter with a summary.

2.1 Related Work

An in-depth survey of face detection algorithms up to the year 2001 is presented in [133] where the different methods are classified into four categories. These categories, along with their pros, cons and sample examples, are listed as follows:

- **Knowledge-based methods.** These use human-centric rules to define the formation of a face.

Pros:	Easy to come up with simple rules, e.g. encoding the relative distance between different features. Usually work well for frontal faces in uncluttered environments.
Cons:	Difficult to translate the human knowledge of faces into rules. Hard to generalize to different poses since it is implausible to define a set of rules for every pose.
Sample Method:	Multiresolution rule-based method [132]

- **Feature invariant methods.** Those methods try to detect features that are unique to faces such as facial components (e.g. eyes, nose, mouth, ...), texture, skin color, or a combination of the lateral.

Pros:	These features are normally invariant to pose.
Cons:	Might be difficult to locate such features in complex backgrounds and/or varying illumination.
Sample Methods:	Facial features by grouping of edges [60] Texture [20] Skin color [71] A recent work that combines edge, skin color and texture cues [76]

- **Template matching methods.** Standard patterns are manually set or parametrized to describe a face.

Pros:	Simple to implement.
Cons:	Initialization is needed. As in the case of knowledge-based methods, different templates are needed for different poses.
Sample Method:	Active Shape Model [59]

- **Appearance-based methods.** Unlike template matching methods, the face pattern is learned from data rather than being manually specified.

Pros:	Powerful statistical modeling and machine learning are employed to achieve fast and robust results.
	Can be extended to deal with different poses and orientations.
Cons:	Normally an exhaustive search over space and scale is needed for detection.
	Usually require a long training period with lots of positive and negative examples.
	Do not necessarily generalize to views not in the training set.

Due their empirical successes, those methods have emerged as the standard in face detection since the early 2000's, and much of the work done after the publication of the aforementioned survey falls into this category. Therefore, more discussion is dedicated to those in the rest of this section.

With the increase of compute power and storage capacity, appearance-based methods have been widely adopted in face detection (and similarly in general object recognition). Given the ability to collect a large set of positive and negative examples of faces, as well as to train complex machine learning algorithms, most research in face detection moved in this direction showing experimental success. The main two components of these methods are: 1) features selection and 2) learning algorithms. It can be argued that the work that paved the way for this development is the previously mentioned seminal paper by Viola and Jones [114]. The features they selected were simple Haar-like features and the learning algorithm they chose was Adaboost, trying to create an accurate classifier by combining many weak classifiers. A thorough recent survey of the different features and learning algorithms used in face detection, including various boosting algorithms, can be found in [134]. A summary of the different surveyed features, citing relevant work, is represented in table 2.1, while table 2.2 summarizes various recent learning algorithm, alongside their representative papers.

All the appearance-based method listed above are rooted in the sliding window paradigm, demonstrated in figure 2.1, where window patches are taken at regular grids. As an alternative search technique, recent object detection methods use interest points extracted from the whole image to speed up the detection process [34]. By processing only a set of patches around interest points, the search space is reduced, usually without much impact on the detection rate. This relatively new paradigm, exemplified through the bag-of-words method in figure 2.2, is being applied lately to faces [105, 108]. Details on the bag-of-words approach can be found in [26]. Local interest points detectors include Harris [41], Harris-Laplace [73], and Harris-Affine [74], in addition to others. Features extracted from those points, rather from the grid windows, are used in the classification process.

Feature Type	Representative Work
Variations of Haar-like features	Rotated Haar-like features [67] Rectangular features with structure [63, 53] Haar-like features on motion filtered image [115]
Pixel-based features	Pixel Pairs [15] Control point set [1]
Binarized features	Modified census transform [29] LBP features [52, 137] Locally assembled binary feature [131]
Generic linear features	Anisotropic Gaussian filters [72] LNMF [19] Generic Features with KL boosting [70] RNDA [121]
Statistics-based features	Edge orientation histograms [61, 21] Spectral histogram [124] Spatial histogram [135] HoG and LBP [123] Region covariance [112]
Composite features	Joint Haar-like features [75] Sparse feature set [46]
Shape features	Boundary/Contour fragments [82, 101] Edgelet [129] Shapelet [95]

Table 2.1
Summary of various features used in face/object detection. Table source: [134].

Learning Method	Representative Work
Boosting Techniques	Adaboost [114] RealBoost [128, 75] GentleBoost [66, 18] FloatBoost [63]
Bayesian	Bayesian discriminating features method [69]
SVM (speed up)	Reduced set vectors and approximation [92, 90] Resolution based SVM cascade [44]
SVM (multiview face detection)	SVR-based detection [64] SVR fusion of multiple SVMs [130] Cascade and bagging [120] Local and global kernels [45]
Neural Networks	Constrained generative model [27] Convolutional neural network [30, 81]
Part-based approaches	Wavelet localized parts [97, 96] SVM component detectors adaptively trained [43] Overlapping part detectors [74]

Table 2.2
Summary of recent learning algorithms used in face/object detection. Table source: [134].

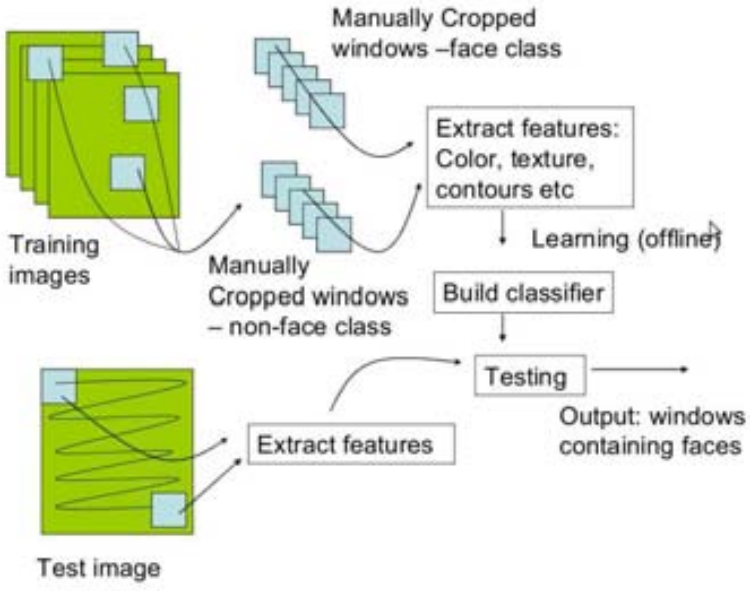


Figure 2.1: Sliding window detection paradigm. Image source: [34].

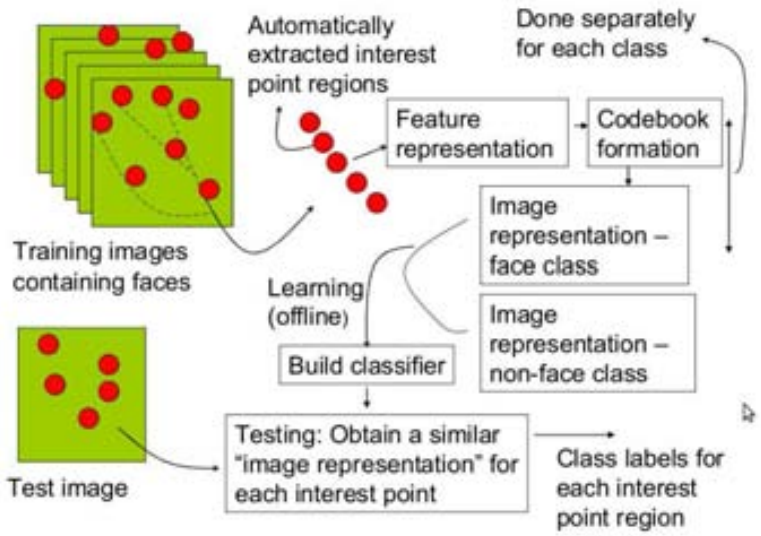


Figure 2.2: Interest points detection paradigm. Image source: [34].

2.2 A First Generation Model

Our early attempt was motivated by the fact that skin is an effective and robust cue for face detection. Skin color is highly invariant to geometric variations of the face and it allows fast processing. Surveys on skin color detection can be found in [113, 55]. In most of the work on skin color, it has been assumed that colorspaces separating luminance from chromaticity, e.g. YCbCr or HSV, perform better in detection than RGB due to the assumption that transforming the colorspace will reduce the overlap between skin and non-skin pixels [55]. Moreover, dropping the luminance component is appealing because it changes the classification space from 3D to 2D. Due to this, colorspace transformation has been the dominant trend in skin color detection. However, some work surfaced doubting any real effect of colorspace transformation [9, 100]. In this section, we will use RGB for skin color detection and in the next section we will present a detailed analysis of the performance of RGB vs. other colorspaces.

The face detection method shown in this section belongs to first generation pattern recognition paradigm where no machine learning is employed, but rather the rules are set by human observations. Skin color segmentation is applied to separate the skin areas from non-skin. Edge detection with dilation is then implemented to separate face candidates from any background or foreground blobs. Connected components are later analyzed using simple shape features to decide whether a skin blob is a face or not. The complete system is shown in figure 2.3.

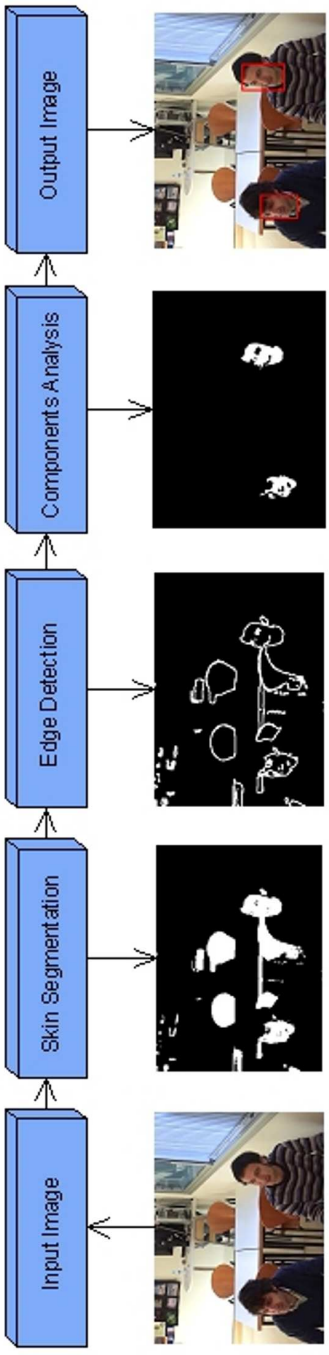


Figure 2.3: An overview of our face detection system.

2.2.1 Skin Segmentation

Trying to model skin color, around 1,000,000 skin pixels were sampled from the UCFI database which contains images of people with different ethnicities and under various lighting conditions [99]. The quality of the images also covers a wide spectrum with certain images obtained by digital cameras while others scanned using photo-scanners. Studying the distribution of these pixels in the RGB space, we concluded that a pixel with (R,G,B) value should be classified as skin if:

$\mathbf{R > 75}$ **&**
$\mathbf{20 < R - G < 90}$ **&**
$\mathbf{R/G < 2.5}$

The advantages of this method are its simplicity and computational efficiency, especially that no colorspace transformation is needed, which motivated its application for smart phones [54]. The distributions of R, $R-G$, and R/G are shown in figure 2.4. As demonstrated in the lateral figure, our experiments revealed that 96.9% of the skin pixels have their R value greater than 75, 94.6% of them have the difference between their red and green values, $R-G$, ranging from 20 to 90, which supports the observation in [8], and 98.7% of them have their R/G values less than 2.5.

After skin pixels are detected and used to mask the original image, an approximation of the sobel edge detector is applied in an attempt to separate any face blob from a foreground or background blob that might have similar color. In the sobel approximation, the gradient magnitude, $|G|$, is computed as the sum of x-direction gradient magnitude, $|G_x|$, and the y-direction gradient magnitude, $|G_y|$, i.e. $|G| = |G_x| + |G_y|$ instead of $|G| = \sqrt{(G_x)^2 + (G_y)^2}$. A dilation process is then employed to further separate the edges. An example of this process is shown in figure 2.5.

2.2.2 Shape Features

The resulting segmented image is searched for connected components and each of the components is then analyzed to determine whether it is a face or not. In our initial attempt and in order to select the features for the face vs. non-face classification, we started looking at few sample images from recorded sequences in which we needed to detect the faces present (more details on those sequences are available in the next section). Given the fact that the skin color segmentation greatly reduced the search space leaving only the faces and few other components that are significantly different, we noticed that simple features can do a very good job in the classification. Therefore, we defined scale and pose invariant shape features to detect the face blobs, setting the values based on the observed sample images and the general anatomy of the face. Those features and their corresponding rules are listed below. Each of the rules set on one of the features can be thought of as a very weak classifier, and their combination over skin segmented blobs leads to a strong classification.

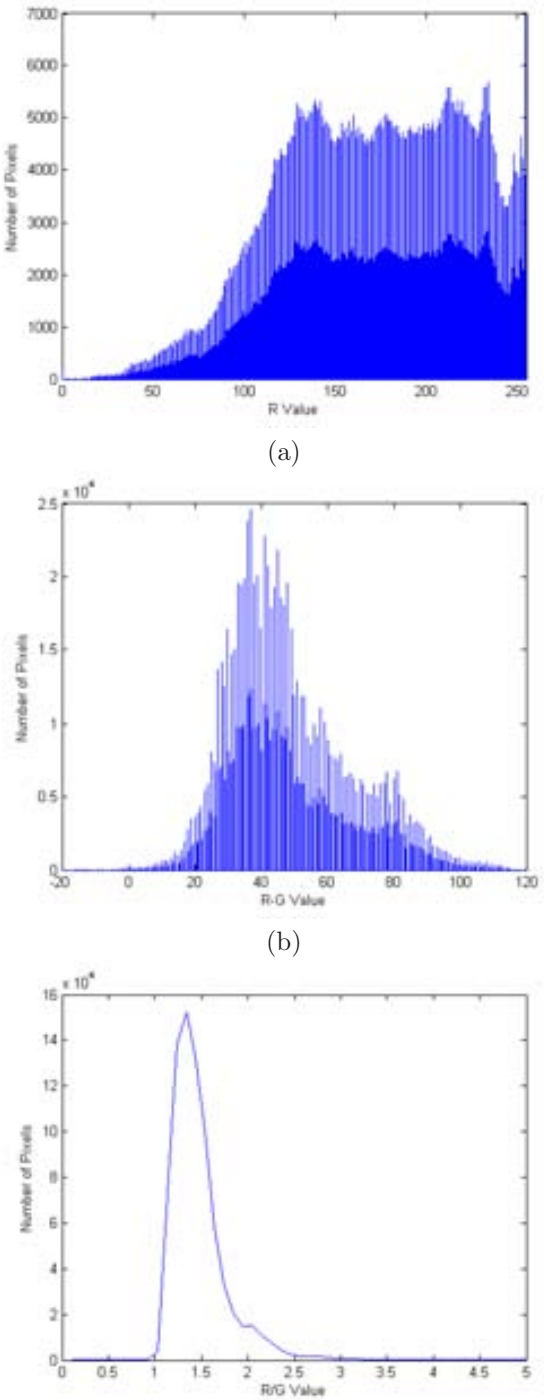
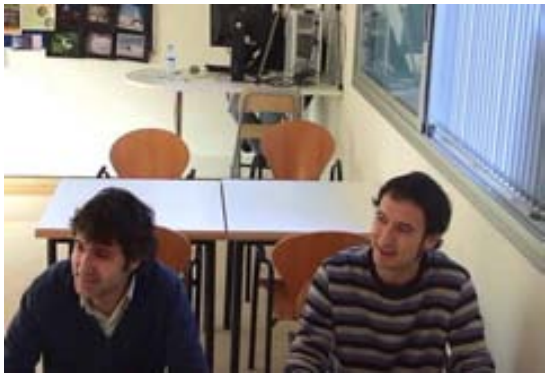
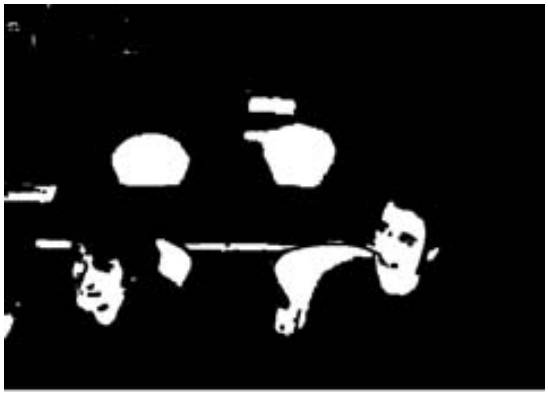


Figure 2.4: RGB distributions of skin pixels in: (a) R, (b) $R - G$, and (c) R/G.



(a)

(b)



(c)

Figure 2.5: An example of the skin segmentation process: (a) original image, (b) skin color mask, and (c) segmented image.

Aspect Ratio: Defined as the ratio of the bounding box longer side to the shorter. The ratio of a face height to its width is around 1.4 on average, with some variations from one person to another and between different poses. In our experiments, any region whose aspect ratio is greater than 1.8 is classified as a non-face.

Euler Number: Defined as the number of objects in the region minus the number of holes in those objects. Given the geometry of the face, it is expected to see at least one hole in the region that corresponds to a face. Thus, any region with no holes is classified as a non-face.

Extent: Defined as the area of the blob divided by the area of the bounding box. Given the elliptical form of the face and its skin distribution, our experiments revealed that the extent for a face is between 0.3 and 0.8. Thus, any region whose extent is not in this range is classified as a non-face.

Orientation: Measured as the absolute angle between the horizontal axis and the major axis of the ellipse having the same second-moments as the blob. Given the limit on the pan and tilt of one's head, this value is expected to range from 20° to 160° for natural poses. Any blob whose orientation is outside this range is classified as a non-face.

Centroid Position: Represented by two features encoding the distance between the centroid of the blob and the center of its bounding box in both x and y directions. The face is evenly distributed in the region where it is located. Therefore, the centroid of a face region should be found in a small window centered in the middle of the bounding box. The dimensions of this window were experimentally set to be 15% of the dimensions of the bounding box. Any region whose centroid is outside this 15% window corresponds to a blob that is not evenly distributed and therefore it is not a face.

2.2.3 Experimental Results

At the time this method was developed, most results were reported on databases with gray-scale images, such as FERET face recognition database and CMU face detection database [136]. Therefore, we have tested our method on sequences that we generated and made publicly available. Few sample images from those sequences were used to tune the rules of the shape features as indicated earlier. The recorded scenario consisted of one person entering a cafeteria, to be later joined by two of his friends. The three of them sit together and chat for a while before leaving. The advantage of the sequences, captured from different angles, is that they provided us with color images containing different faces that vary in pose, size, position, and expression.

Table 2.3
QUANTITATIVE RESULTS OF THE FIRST GENERATION MODEL ON IMAGES FROM RECORDED SEQUENCES.

Number of Faces	True Positives	False Positives	Detection Rate	Precision
266	238	22	89.45%	91.5%

The sequences are available at http://iselab.cvc.uab.es/indoor_hermes_cam3 and http://iselab.cvc.uab.es/indoor_hermes_cam4. Out of those sequences, and since many frames contained similar face appearances, we collected 211 different images that summarize the variation in pose, size, position and expression. Those images contained 266 faces. Our method, despite its simplicity, was able to correctly detect 238 of those faces with 22 false positives, resulting in a detection rate of 89.5%. Those quantitative results are summarized in table 2.3, while qualitative results are shown in figure 2.6.

2.3 Skin Color Modeling

After our first generation model was implemented, we wanted to build a more complex system that is capable of generalizing to different scenarios. We started by examining whether there is a better colorspace, than RGB, for skin segmentation. This section is dedicated to our findings regarding the colorspace selection. The main aim of this part is to refute a common practice fallacy, which is the assumption that colorspaces separating the intensity/luminance component from chromaticity, e.g. rg-chromaticity, HSV and YCbCr, improve skin classification. Colorspace transformation has been a dominant trend in skin color detection [55].

It is important to note here the work by Albiol el al. where the authors proved that for every colorspace there exist an optimum skin detector scheme such that the performance of all these optimum detectors is the same [9]. Their main argument can be summarized by the following: as long as the transformation T mapping colorspace C_i to colorspace C_j is invertible (i.e. $T(C_i) \rightarrow C_j$ & $T^{-1}(C_j) \rightarrow C_i$), any hyperplane producing an optimum separation in one of the colorspaces can be transformed, either by T or T^{-1}, to produce the same optimum result in the second. However, when researchers speak about the separability of skin vs. non-skin colors, they are usually referring to the ease of the separation, or, better put, the accuracy that is achieved by simple classification. The reason why researchers prefer one colorspace over another is mainly due to the assumption that skin colors will be easier to separate in that space. There is a common belief that skin colors "share almost a common region in the chromatic space" [33] which makes chromatic colorspaces a more natural choice for skin detection. Our experiments reveal that there is no advantage of any colorspace over the RGB, especially when using a very simple linear classifier as will be shown later.

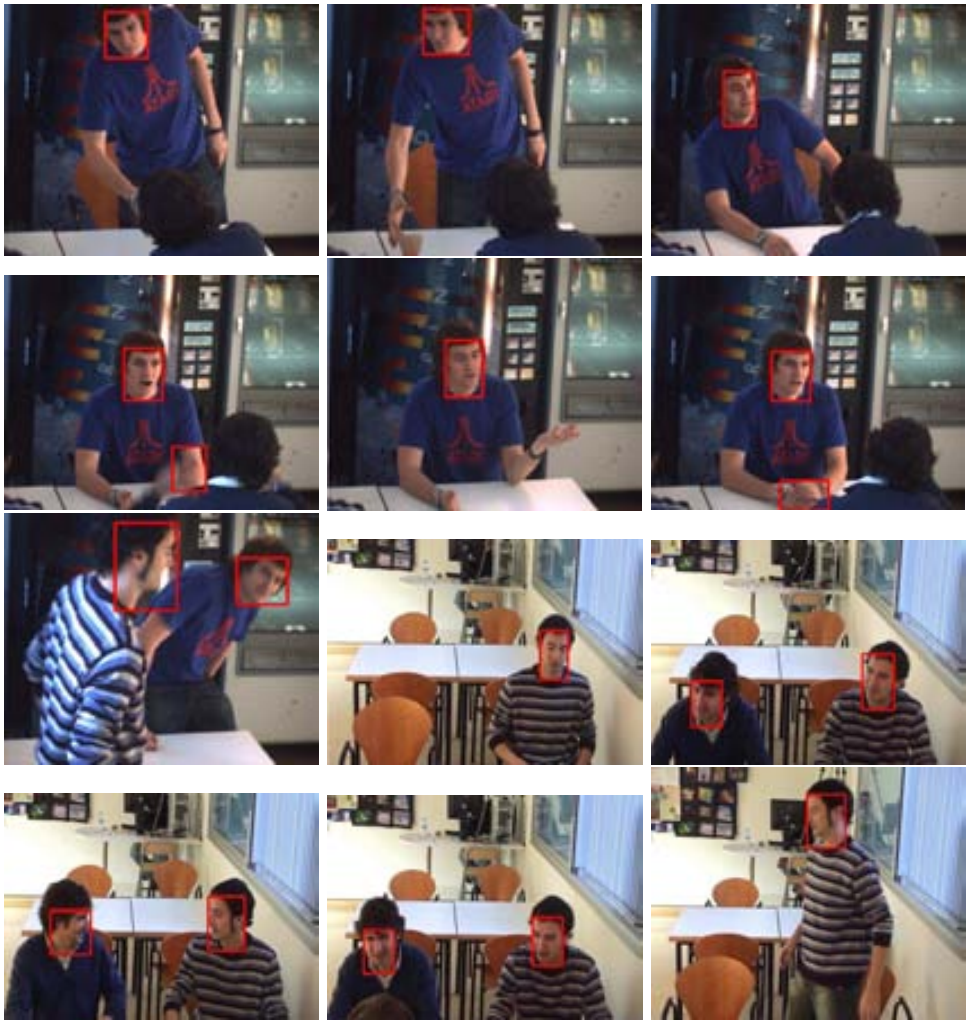


Figure 2.6: Qualitative results of the first generation model on images from recorded sequences.

2.3.1 Skin vs. Non-skin Pixels

To model skin color, we used the 1,000,000 skin pixels that were sampled from the UCFI database. Acquiring the skin pixels, although laborious, is a straightforward task. We know exactly what skin is and where it exists, but what about non-skin tones? Should we include pixels from a wooden door as negatives? what about a silver clock or a brownish cardboard? The answer is not obvious and the choices made at this stage will inevitably affect the accuracy of our classifier. Taking only the pixels which are very distinct from skin (i.e. greenish or bluish) will result in too many false positives while taking too many "skin-looking" pixels will result in many false negatives.

To work around this issue, we decided to randomly select an equal number of pixels, i.e. 1,000,000, from images that do not contain skin. The intuition is the following: taking a certain number of pixels to represent skin and the exact same number to represent non-skin, we are guaranteed to have the skin colors forming an overpopulated subregion(s) compared to the non-skin, irrespective of the colorspace. However, if it is true that a certain colorspace renders the skin pixels more separable, then, for that colorspace, we would expect to have less overlap between skin and non-skin regions, increasing the accuracy of the classification.

To get non-skin pixels, we used Caltech 101 database after removing the two categories: Faces and Faceseasy; henceforth, we will refer to it as Caltech 99. These two categories were removed since the pixels coming from them have a much higher probability of being skin pixels than non-skin. We randomly selected 145 pixels with unique colors from each and every image in Caltech 99 to form the 1,000,000 non-skin pixels.

2.3.2 Knn Classifier

Given the setup of the problem, i.e. the fact that the skin space might overlap with non-skin space but with higher mode, a k nearest neighborhood (knn) classifier is one of the best classifier to be used in this scenario. In regions which are "skin-like" we will have more pixels that are skin than pixels which are non-skin, while the opposite is true for the non-skin regions.

For training and testing, we divided our dataset in half. So, 1,000,000 pixels (half of them are skin and the other half are non-skin) were used for training and the remaining 1,000,000 pixels were used for testing. We investigated the accuracy in 6 colorspaces: RGB, HSV, YCbCr, rg-chromaticity, HS and CbCr. The number of neighbors was determined as the number minimizing the leave-one-out error on the training set. The overall accuracy of the classification is shown in table 2.4.

Repeating the experiment multiple times, it was clear that the accuracy in the 3 colorspaces: RGB, HSV and YCbCr was the same and any small difference in performance had no statistical significance. It was also clear that removing the luminance/intensity component worsened the results, as can be seen in the colorspaces: rg-chromaticity, HS, and CbCr.

For qualitative analysis, we used a new dataset called FDDB [51]. The aim behind using a new dataset is to test the generalizability of our model. Some sample results are shown in figure 2.7. For those images, we subsampled the training set taking only

Colorspace	Accuracy (knn)
RGB	**88.40%**
HSV	**88.43%**
YCbCr	**88.41%**
rg-chromaticity	86.85%
HS	86.58%
CbCr	86.52%

Table 2.4
QUANTITATIVE KNN CLASSIFICATION RESULTS ON SKIN VS. NON-SKIN IN DIFFERENT COLORSPACES.

2.5% of its data to reduce the time complexity of classification. This lowered the time necessary for classification by more than 40 folds without much decrease in accuracy. The results in Figure 2.7 verify the conclusions drawn from the quantitative analysis. We can see that there is a very little difference between the 3 colorspaces: RGB, HSV and YCbCr, and the results in those spaces are slightly better than the rest.

2.3.3 Naive Bayes Classifier

We repeated the same experiment with a linear naive Bayes classifier. We wanted to see how this classifier will perform in the different colorspaces and if it will favor any colorspace over the RGB. Although the knn classifier is optimal in this scheme, the naive Bayes is a much faster classifier which makes it worth exploring.

Given the two classes, Class = {skin, non-skin}, and the feature vector, $\mathbf{X} = \{x_1, .., x_n\}$, which represents a pixel color values in a certain colorspace (clearly X is three dimensional in RGB, HSV, YCbCr and two dimensional in rg, HS, CbCr colorspaces), and using Bayes' rule, we can write the following for a class $Class_j$ and a feature vector $\mathbf{X}$:

$$P(Class_j|\mathbf{X}) \propto p(Class_j) \prod_{i=1}^{n} p(x_i|Class_j).$$

In our case, we assume equal prior probabilities, i.e.

$$P(Class_{skin}) = P(Class_{non-skin}) = 0.5,$$

and the probability of each feature is given by a normal distribution:

$$p(x_i|Class_j) = \frac{1}{\sigma_{ij}\sqrt{2\pi}} e^{\frac{-(x_i-\mu_{ij})^2}{2\sigma_{ij}}}$$

where μ_{ij} is the mean of feature i in colorspace j and σ_{ij} is the variance of feature i in colorspace j, obtained from the training set.

Classification is done via maximum a posteriori decision rule, i.e. a feature vector $\{x_1, .., x_n\}$ is classified as class c which maximizes the product of the class prior and

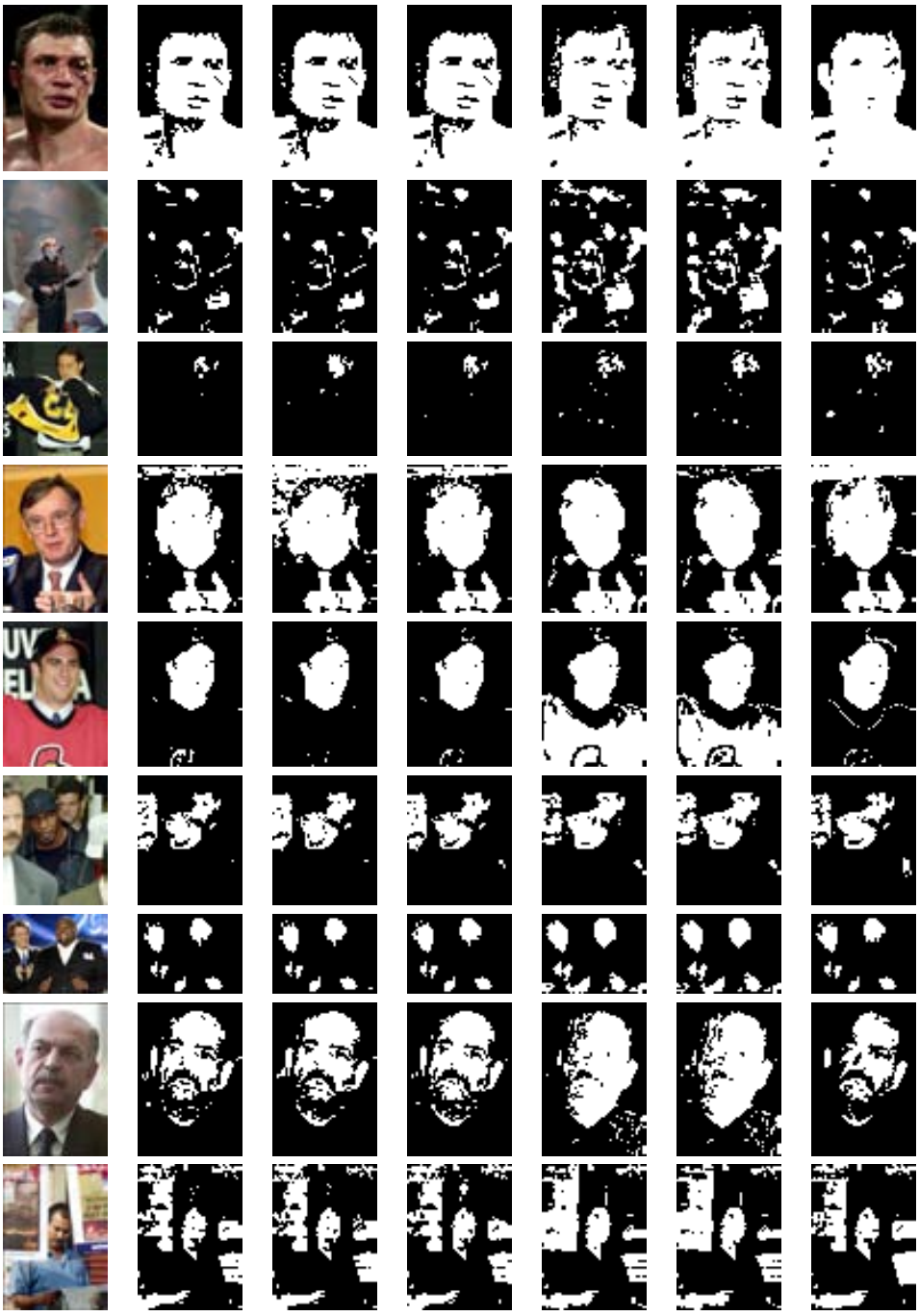


Figure 2.7: Qualitative knn classification results on skin vs. non-skin in different colorspaces: (a) original image, (b) result in RGB, (c) in HSV, (d) in YCbCr, (e) in rg, (f) in HS, and (g) in CbCr.

Colorspace	Accuracy (knn)	Accuracy (naive Bayes)
RGB	**88.40%**	**83.44%**
HSV	**88.43%**	72.39%
YCbCr	**88.41%**	**83.44%**
rg-chromaticity	86.85%	76.05%
HS	86.58%	77.35%
CbCr	86.52%	83.29%

Table 2.5
Quantitative linear naive Bayes and knn classification results on skin vs. non-skin in different colorspaces.

the class density, i.e.

$$\arg\max_c p(Class = c) \prod_{i=1}^{n} p(x_i|Class = c).$$

We follow the same training and validation scheme as in the knn case (50% for training and 50% for testing). The results for the naive Bayes along side the knn results are show in table 2.5. It can be seen in the lateral table that the performance in the HSV colorspace dropped drastically when using Bayes linear classifier, and a better result could be obtained in the HS space than HSV. However, still the best performance was obtained in the RGB colorspace along with the YCbCr, while CbCr shows very similar results. Qualitative results of the linear naive Bayes classification on the same images, on which knn was evaluated, are presented in Figure 2.8.

From the work presented in this section, it can be seen that there is no advantage of HSV, YCbCr and rg-chromaticity spaces over RGB in skin detection accuracy. Also, a robust detector could be designed in the RGB colorspace using simple classification techniques when having enough representative samples to model the skin tones and the non-skin colors.

2.4 A Second Generation Model

Using the skin classifiers obtained in the previous section, we wanted to develop the face detector presented in section 2.2 further to generalize to different scenarios. As any first generation model, we found that editing the rules was not a proper way to proceed, especially that an exception after an exception needed to be added. Therefore, we followed an evolution similar to the pattern recognition field where the tyranny of the data prevails over hand-crafted rules, and statistical models are used to automatically learn the discriminative functions from the data. Although our knn classifier is more accurate in skin classification, we used the naive Bayes classifier trained in the previous section to select skin regions due to its faster performance. Additional shape features are added and different classifiers are tested to make the classification process more robust.

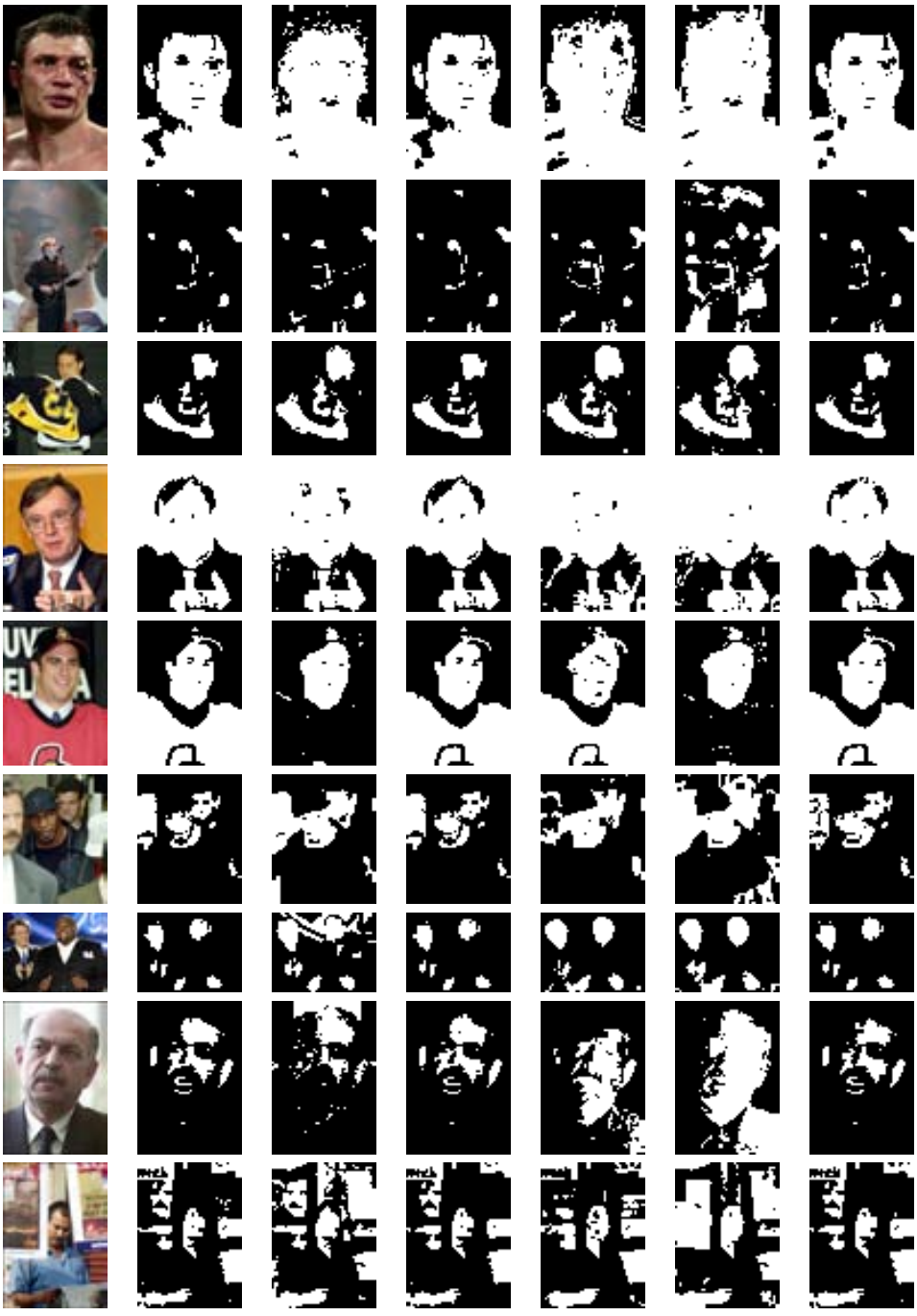


Figure 2.8: Qualitative naive Bayes classification results on skin vs. non-skin in different colorspaces: (a) original image, (b) result in RGB, (c) in HSV, (d) in YCbCr, (e) in rg, (f) in HS, and (g) in CbCr.

2.4.1 Additional Shape Features

In the addition to the features presented in section 2.2.2, more view-invariant scale-independent features are extracted, resulting in a feature vector made of 16 shape features. These additional features are listed below.

Eccentricity: Fitting the whole blob under an ellipse which has the same second-moments as the blob, eccentricity is defined as the distance between the foci of this ellipse and its major axis.

Solidity: The proportion of the pixels in the convex hull that are also in the region.

Roundness: The ratio of the minor axis of the ellipse to its major axis.

Hu Moments: Hu moments are very useful for our problem since they are invariant under translation, rotation, and changes in scale. They are computed from normalized centralized moments up to the third order as shown below:

$$
\begin{aligned}
I_1 &= \eta_{20} + \eta_{02} \\
I_2 &= (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \\
I_3 &= (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\
I_4 &= (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \\
I_5 &= (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + \\
&\quad (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\
I_6 &= (\eta_{20} - \eta_{02})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2 \\
&\quad + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})] \\
I_7 &= (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + \\
&\quad (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]
\end{aligned}
$$

2.4.2 Experimental Results

Having generated the corresponding features of each skin-color blob in a certain image, what remains is to classify which of these blobs correspond to a face. For that purpose several different classifiers have been evaluated. The first part of this section describes the training process where three classifiers were trained on faces from the CVL database, while the second part shows the classification results on one of our independent sequences, where the face varies under pose and expression.

2.4.2.1 Results on CVL Database

The face images used in this section have been provided by the Computer Vision Laboratory (CVL), University of Ljubljana, Slovenia [84, 103]. The CVL database contains color images of people under seven different poses with varying expressions. The database contains around 800 faces that we used as positive examples for the face class. For the non-face class, many images that do not contain any faces but do contain skin-colored regions have been collected from various sources. A total

of around 2100 non-face skin-colored blobs were detected in those collected images. Examples of faces blobs and non-face blobs are shown in Fig. 2.9, where it can be noted that the negative examples contained body parts as well as non-body regions. We generated the invariant features on the face and the non-face blobs and used them to train several classifiers. The classifiers that were tested are the following:

- **Linear Bayes Classifier:** Bayes classifier assuming normally distributed classes and equal covariance matrices.
- **Decision Tree Classifier:** hierarchically based classifier which compares the data with a range of automatically selected features. The selection of features is determined from an assessment of the spectral distributions or separability of the classes.
- **K Nearest Neighbor:** classifies objects based on closest training examples in the feature space. The number of neighbors was determined as the number minimizing the leave-one-out error on the training set.

The learning curves for the different classifiers are shown in Fig. 2.10. The learning curves were computed by varying the size of the training set. For each training set size, the classification error was estimated by averaging over 10 trials (i.e. 10 randomly selected training sets for each size). It can be noted from the lateral figure that the three classifiers can achieve good results with small training sets. The ROC curves are shown in Fig. 2.11 where half of the data was used for training while the other half was used for testing. It can be seen that a high true positive rate can be achieved despite low false positive rate, with the decision tree classifier performing the best in spite of an initial slight lag behind the knn classifier.

2.4.2.2 Classification on a Video Sequence

We tested those classifiers trained in the previous section on images from one of our recorded sequences http://iselab.cvc.uab.es/indoor_hermes_cam4. Skin color segmentation is done using the Bayes skin classifier since it is much faster than the knn classifier, and small blobs whose area is less than 2% of the image area are dropped. The three classifiers, that were trained on the faces from the CVL database and the non-faces we collected, are tested on 541 images containing 541 faces. The results of the various classifiers, in addition to the Viola and Jones baseline, are shown in table 2.6. As expected, the decision tree classifier achieved the highest detection rate. Moreover, in this particular scenario, the linear Bayes classifier outperformed the knn classifier achieving the highest precision.

It is important to mention that we were able to achieve a high classification rate despite the fact that the testing sequence is completely unrelated to the training samples. All our classifiers outperformed the Viola and Jones detector having a higher detection rate with a much lower false positive rate. A sample result of the Viola and Jones detector on one of our images is shown in figure 2.12. Some experimental results of the decision tree classifier are shown in figure 2.13, of the knn classifier in figure 2.14, and of the linear Bayes Classifier in figure 2.15.

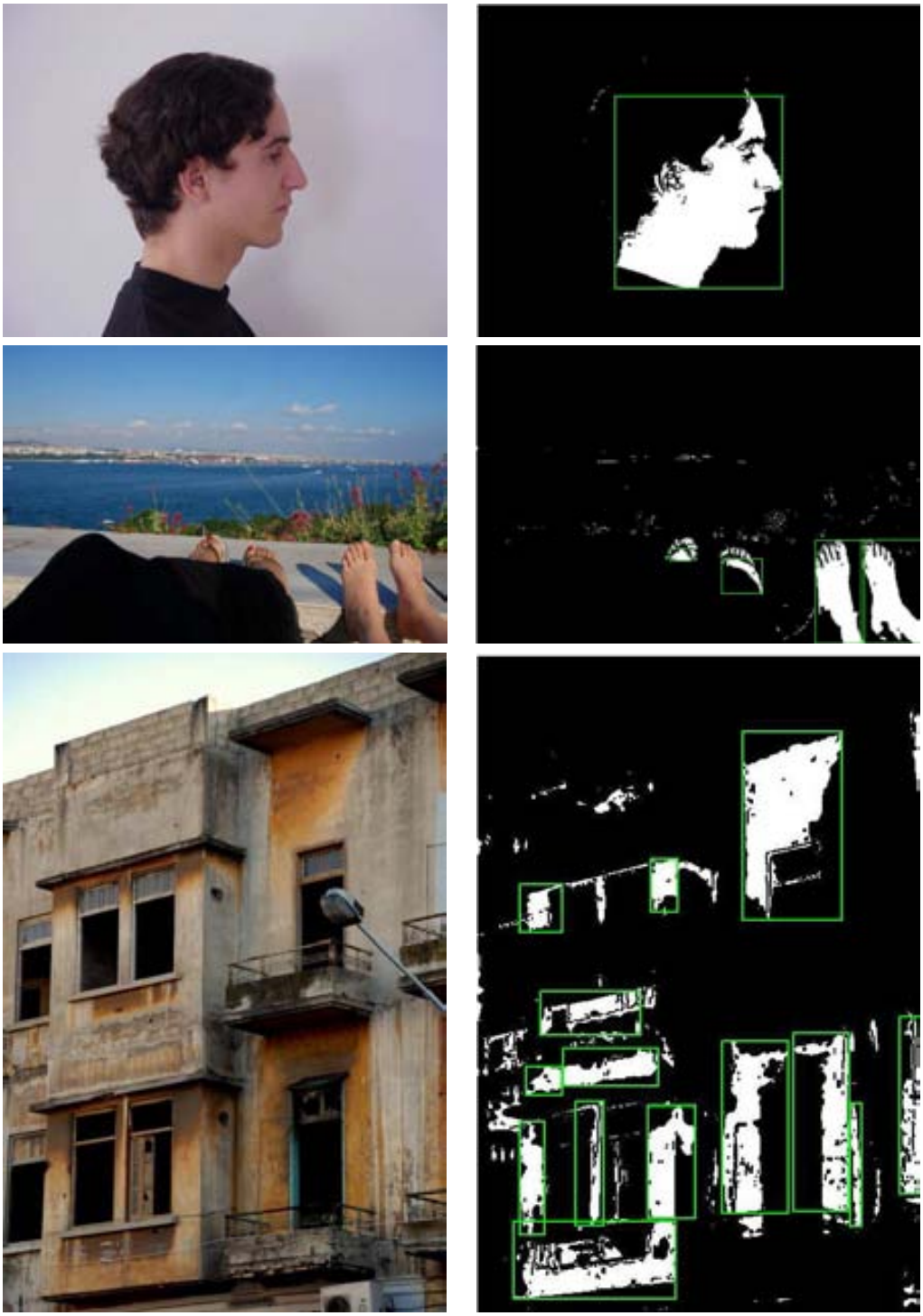


Figure 2.9: Positive and negative face examples, used to train classifiers, shown in green boxes with their original images.

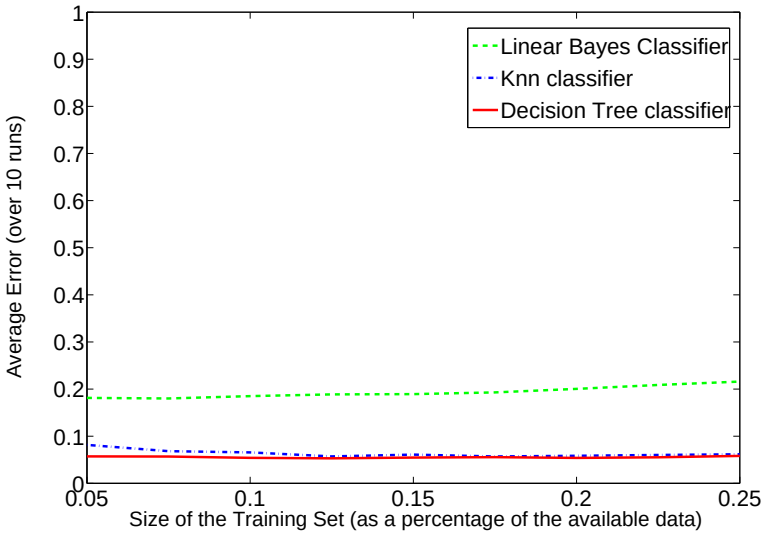


Figure 2.10: Learning curves of the different second generation classifiers.

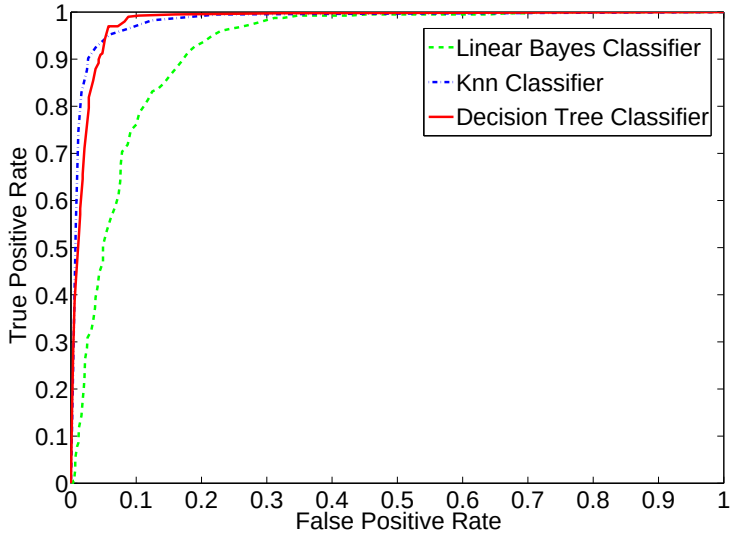


Figure 2.11: ROC curves of the different second generation classifiers.

Table 2.6
Results of the Second Generation Detectors.

	Number of Faces	True Positives	False Positives	Detection Rate	Precision
Viola and Jones	541	386	470	71.35%	45.09%
Linear Bayes (ours)	541	462	9	85.40%	98.09%
Knn (ours)	541	400	73	73.94%	84.57%
Decision Tree (ours)	541	476	54	87.99%	89.81%

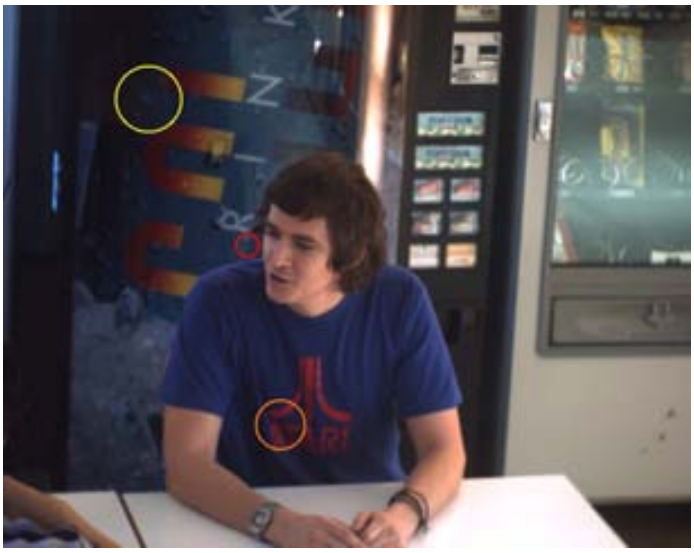


Figure 2.12: Sample output of the Viola and Jones detector on one of our images.

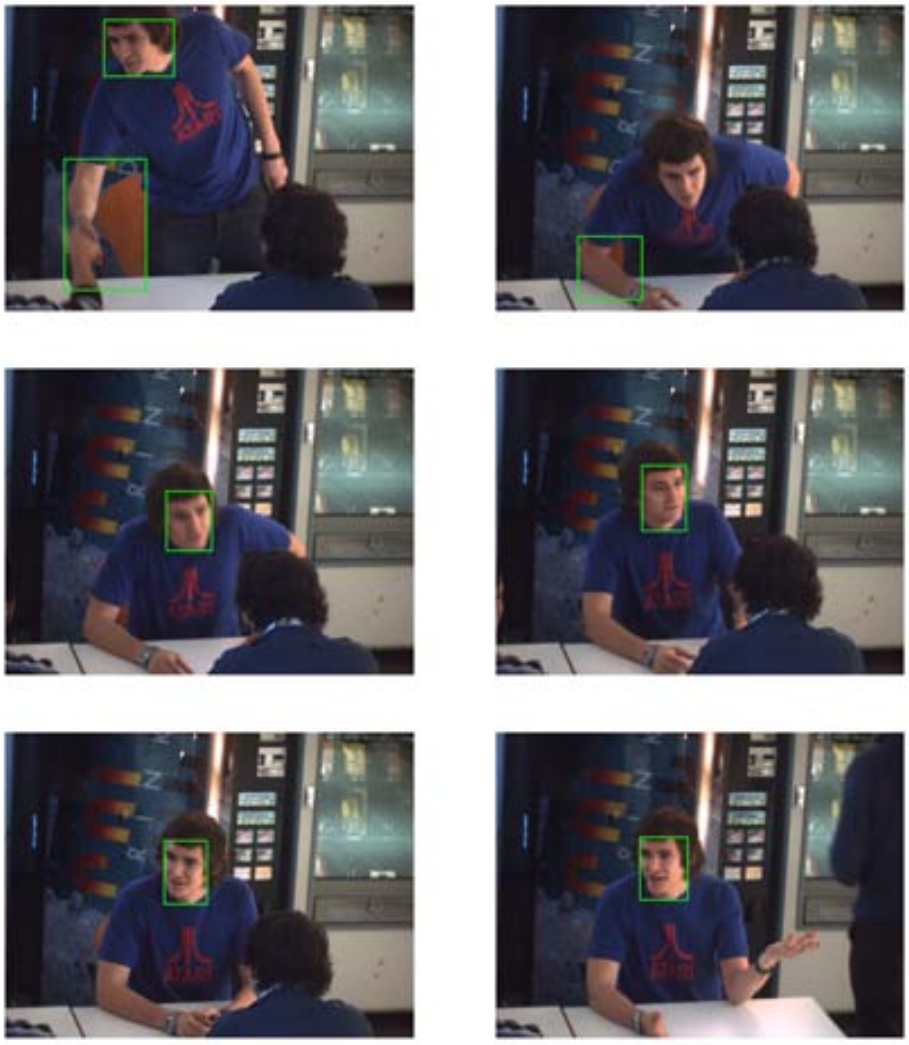


Figure 2.13: Some results of the decision tree classifier.

Figure 2.14: Some results of the knn classifier.

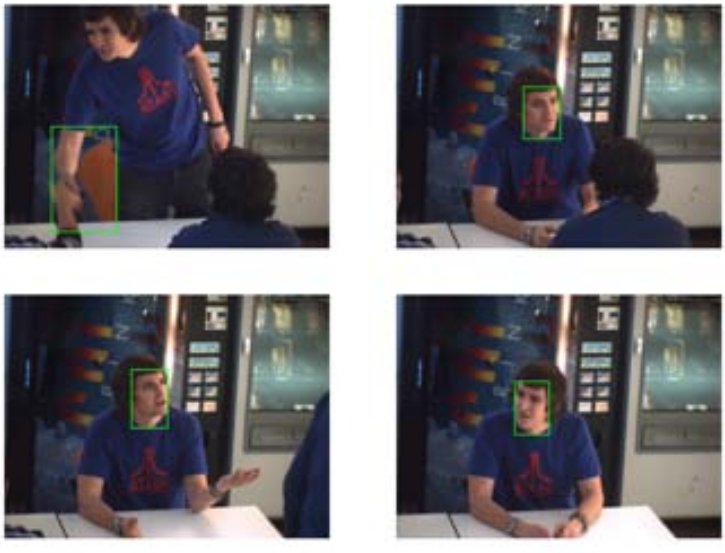


Figure 2.15: Some results of the linear Bayes classifier.

2.5 Closing Remarks

In this chapter, a first and second generation face detection models are presented along with a study on colorspace selection. Our approach is based on segmenting images into skin-colored blobs that are later analyzed to extract scale-independent and view-invariant features. These features are used to discriminate face blobs from non-face blobs.

In our second generation model, statistical pattern classifiers are trained instead of the rules set in the first generation model. The use of scale-invariant features, particularly the Hu-moments characterizing the spatial distribution of skin pixels in a candidate blob, is key to the success of our second generation algorithm. The use of invariant features on segmented blobs obviates the need to scan all possible regions of an image at multiple scales looking for candidate faces. This results in a more efficient algorithm and reduces the need for a classifier with a vanishingly small false-positive rate.

A strong advantage of our approach is its ability to generalize to new data. The Viola and Jones face detector is sensitive to the illumination and imaging conditions under which it is trained, and consequently it does not generalize well to new situations without retraining. In the presented experiments, we show that, with our invariant feature space representation of skin blobs, we can train a classifier on one dataset and it is able to accurately detect faces on an independent sequence it has never seen before.

Chapter 3

Reactive PTZ Tracking

"... with that giant helicopter in the distance.
It's not giant and it's not in the distance.
It's small and it's in our room!"

Homer Simpson. The Simpsons. Season 18 Episode 5.

Many applications in facial analysis, as well as in the general computer vision field, benefit from high resolution imagery. An example of those application is face recognition where it has been observed that higher resolution improves accuracy [117]. Other applications include license-plate identification [11] and identifying people in surveillance videos, where having highly zoomed images is a must. The problem with zoom control is that two opposing aims are desirable: the first one is obtaining a maximum resolution of the tracked object, whereas the second is minimizing the risk of losing this object. Therefore, zoom control can be thought of as a trade-off between the effective resolution per target and the desired coverage of the area of surveillance.

With a finite number of fixed sensors, there is a fundamental limit on the total area that can be observed. Thus, maximizing both the area of coverage and the resolution of each observed target requires an increase in the number of cameras. However, such an increase is highly costly in terms of installation and processing. Therefore, a system utilizing a smaller number of Pan-Tilt-Zoom (PTZ) cameras can be much more efficient if it is properly designed to overcome the obvious drawback of having less information about the target(s).

Towards this end, different works have investigated the use of PTZ cameras to address this problem of *actively* surveying a large area in an attempt to obtain high-quality imagery while maintaining coverage of the region [104]. Starting two decades ago, the area of active vision has been gaining much attention, in an attempt to:
1) improve the quality of the acquired visual data by trying to keep a certain object

This chapter is published in a book titled "Visual Analysis of Humans" in 2011 [5]. An earlier version of this work appeared in ICPR10 [4].

at a desired scale, and 2) react to any changes in the scene dynamics that might risk the loss of the target.

Accurate reactive tracking of moving objects is a problem of both control and estimation. The speed at which the camera is adjusted must be a joint function of current camera position in pan, tilt and focal length, and the position of the tracked object in the 3D environment.

Motivated by reactively tracking a face, this chapter is dedicated to active vision, in which we formulate the problem of jointly estimating the camera state and the 3D object position in a Bayesian estimation framework. Section 3.1 discusses the different design alternatives for active cameras configurations, such as the autonomous camera approach, the master-slave approach and the active camera network approach, in addition to touching upon the advantages that environment reasoning lends to the problem. Section 3.2 describes our camera-world model setting the stage for estimation and control, which are formulated in sections 3.3 and 3.4 respectively. This chapter is concluded with a summary in section 3.6.

3.1 Related Work

The interest in active camera systems started as early as two decades ago. Beginning in the late 80's, Aloimonos et al. introduced the first general framework for active vision in order to improve the perceptual quality of tracking results [10]. Since then, numerous active camera systems have been developed. In this section, we take a look at different approaches for configuring these systems.

3.1.1 The Autonomous Camera Approach

Autonomous cameras are those that can self-direct in their surrounding environment. Recent work addressing this topic includes that of Denzler et al., where the motion of the tracked object is modeled using a Kalman filter. The camera focal length that minimizes the uncertainty in the state estimation is selected [23]. The authors used a stereo set-up, with two zoom cameras, to simplify the 3D estimation problem.

A newer approach is described by Tordoff et al., which tunes a constant velocity Kalman filter in order to ensure reactive zoom tracking while the focal length is varying [110]. Their approach correlates all the parameters of the filter with the focal length. However, they do not concentrate on the overall estimation problem, and their filter does not take into account any real-world object properties. In the work by Nelson et al., a second rotating camera with fixed focal length is introduced in order to solve the problem of lost fixation [79].

The latter two works are primarily focused on zoom control and do not deal with total object-camera position estimation and its use in the control process. An attempt to join estimation and control in the same framework can be found in the work of Bagdanov et al., where a PTZ camera is used to actively track faces [14]. However, both the estimation and control models used are ad-hoc, and the estimation approach is based on image features rather than 3D properties of the target being tracked.

3.1.2 The Master/Slave Approach

In a master/slave configuration, a supervising static camera is used to monitor a wide field of view and to track every moving target of interest. The position of each of these targets over time is then provided to a foveal camera, which tries to observe the targets at a higher resolution. Both the static and the active cameras are calibrated to a common reference, so that data coming from one of them can be easily projected onto the other, in order to coordinate the control of the active sensors. Another possible use of the master/slave approach consists of a static (master) camera extracting visual features of an object of interest, while the active (slave) sensor uses these features to detect the desired object. In this case, features should be invariant to illumination, viewpoint, color distribution and image resolution, and usually consist of any kind of coarse-to-fine region descriptors, as in [139].

The master/slave approach is a simple but effective formulation that has been repeatedly used for solving many active vision problems [40, 139, 85]. Nonetheless, the use of supervising cameras has the disadvantage of requiring a mapping of the image content to the active cameras. This mapping needs to be obtained from restricted camera placements, movements or observations extended over time [16, 25].

3.1.3 The Active Camera Network Approach

In recent years, interest has grown in building networks of active cameras and optional static cameras, in order to cover a large area while also providing high-resolution imagery of multiple targets [87, 50, 22, 17]. An active camera network is a scaling up of a basic active camera approach, which can be either an autonomous active camera or a master/slave configuration, depending on whether fixed master cameras are deployed or not.

Due to the fact that an active camera network involves multiple cameras and is usually required to accomplish multiple tasks, the challenges of this approach mainly arise from two aspects: i) *task assignment* and ii) *task hand-over*. Task assignment is the problem of deciding which camera resources are to be allocated to which task, or in other words, the problem of camera scheduling. On the other hand, task hand-over describes model transferring from one camera to another.

Furthermore, like the master/slave configuration, active camera networks also require calibration information, as well as extensive networking infrastructure. Communications within such systems require clever networking algorithms for routing and decision making. Though theoretically appealing, active camera networks are expensive to build and maintain, and do not scale well.

3.1.4 Environmental Reasoning

In some cases, low-level approaches such as those described above are not enough to address ambitious applications requiring more complex strategies towards sensors collaboration. Smart coordination among camera sensors requires exploiting resources that are often related to artificial intelligence and symbolic models, including techniques for camera selection according to the given task, protocols for allocating such tasks, tools for reasoning about the environment and mechanisms to resolve conflicts.

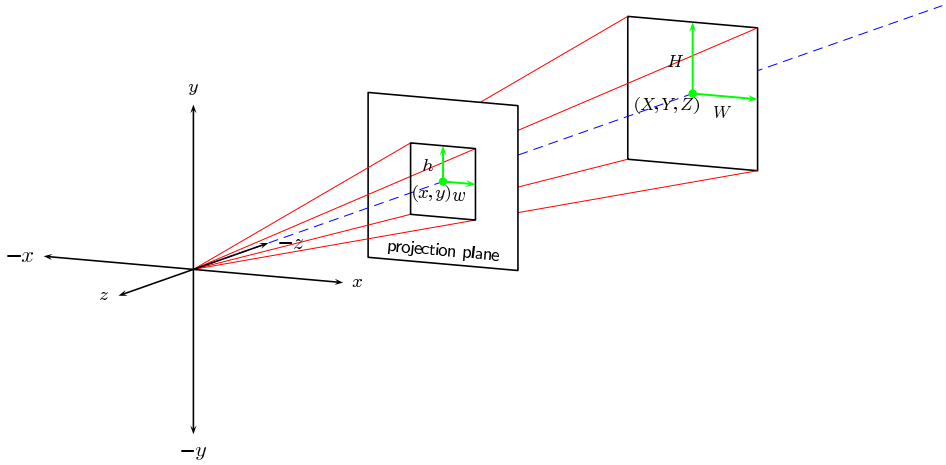


Figure 3.1: The pinhole camera model with the camera positioned at the origin of the world coordinates.

Some examples in which such techniques are used to enhance the collaboration among sensors in a camera network include constraint satisfaction formulations [88], situation graph tree (SGT) [32] and Petri net coordination models [127].

In the remainder of this chapter, we propose a joint framework for camera and object positions estimation. The output of the estimation process is used to control a single PTZ camera, allowing it to reactively track a moving object. Our method belongs to the autonomous camera approach, thus eliminating the cost of multi-cameras while showing robust results on simulated as well as live scenarios.

3.2 Camera-World Model

In our approach, we use a pinhole camera model as shown in figure 3.1. The camera center is located at the origin of the world coordinate system. The principal point is at the origin of the plane of projection at zero pan and tilt. The axis of projection is aligned with the z-axis.

The object being tracked is assumed to be a rigid rectangular patch perpendicular to the axis of projection. It is located at world position (X, Y, Z) with known width W and height H. It is important to note here that upper-case characters, (X, Y, Z, W, H), will be used to denote values in the real-world while lower-case characters, (x, y, w, h), will be used to denote values in the image projection plane.

Changes in camera orientation due to panning and tilting are modeled as pure rotations of the coordinate system:

$$\mathbf{M}(\phi,\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \cos\phi & 0 & -\sin\phi \\ 0 & 1 & 0 \\ \sin\phi & 0 & \cos\phi \end{bmatrix}, \tag{3.1}$$

where ϕ and θ represent the pan and tilt angles, respectively.

We assume that the camera projection is reasonably approximated using equal scaling in the x and y directions (i.e. square pixels). The center of projection is also assumed to be at the origin of the world coordinate system. Then, the camera matrix, $\mathbf{N}$, is fully parametrized by the focal length parameter f:

$$\mathbf{N}(f) = \begin{bmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{bmatrix} . \tag{3.2}$$

The projection of the object at position $\vec{o} = [X, Y, Z]$ onto the plane of projection can now be written as:

$$\mathbf{p}(\phi, \theta, f, \vec{o}) \quad = \quad \begin{bmatrix} \frac{X'}{Z'} & \frac{Y'}{Z'} \end{bmatrix}, \tag{3.3}$$

where X', Y' and Z' are given by the transformation:

$$\begin{bmatrix} X' \\ Y' \\ Z' \end{bmatrix} = \mathbf{N}(f)\mathbf{M}(\phi, \theta)\vec{o}^{\top}. \tag{3.4}$$

The camera model relates the geometry and position of the tracked object in the 3D world to the internal camera parameters. In the next part, we describe how the estimation problem can be formulated.

3.3 Estimation

In this section, we formulate the problem of jointly estimating the camera and object parameters in a recursive Bayesian filter framework.

At time t, the state configuration of the joint camera/object model is represented by the spatial coordinates of the tracked object in the real-world, the camera intrinsics and the velocities of those positions:

$$\vec{s}_t = [\vec{o}_t \mid \vec{c}_t \mid \dot{\vec{o}}_t \mid \dot{\vec{c}}_t]^{\top}, \tag{3.5}$$

where each component is defined as:

$$\vec{o}_t = [X_t, Y_t, Z_t], \tag{3.6}$$
$$\vec{c}_t = [\phi_t, \theta_t, f_t], \tag{3.7}$$
$$\dot{\vec{o}}_t = [\dot{X}_t, \dot{Y}_t, \dot{Z}_t], \tag{3.8}$$
$$\dot{\vec{c}}_t = [\dot{\phi}_t, \dot{\theta}_t, \dot{f}_t]. \tag{3.9}$$

$[X_t, Y_t, Z_t]$ is the position of the planar patch in world coordinates at time t, and $[\phi_t, \theta_t, f_t]$ represent the camera pan angle, tilt angle and focal length at time t, respectively. The remaining elements, $[\dot{X}_t, \dot{Y}_t, \dot{Z}_t, \dot{\phi}_t, \dot{\theta}_t, \dot{f}_t]$, represent the velocities of the previously mentioned components.

From time $t-1$ to time t, the state is updated by the linear matrix $\mathbf{U}$:

$$\vec{s}_t = \mathbf{U}\vec{s}_{t-1} + \vec{v}_{t-1}, \tag{3.10}$$

where $\mathbf{U}$ is defined as:

$$\mathbf{U} = \begin{bmatrix} \mathbf{I}_6 & \mathbf{I}_6 \\ \mathbf{0}_6 & \mathbf{I}_6 \end{bmatrix}, \tag{3.11}$$

and where $\mathbf{I}_n$ and $\mathbf{0}_n$ are the $n \times n$ identity and zero matrices, respectively. The term $\vec{v}_{t-1}$ in equation (3.10) is considered to be a zero-mean, Gaussian random variable adding noise to the system update.

At each time t, an observation $\vec{z}_t$ of the unknown system $\vec{s}_t$ is made:

$$\vec{z}_t = [x_t, y_t, w_t, h_t, \hat{\phi}_t, \hat{\theta}_t, \hat{f}_t], \tag{3.12}$$

where (x_t, y_t) is the center of the object in the image plane measured in pixels, (w_t, h_t) are the width and height of the object in the image plane, also measured in pixels (please refer again to figure 3.1). $(\hat{\phi}_t, \hat{\theta}_t, \hat{f}_t)$ are the camera parameters arriving from the camera imprecise measurements of the pan angle, tilt angle and focal length.

The measurement equation, against which the observation $\vec{z}_t$ is compared, is given by:

$$\mathbf{h}(\vec{s}_t) = [\mathbf{p}(\phi_t, \theta_t, f_t, \vec{o}_t) \mid \mathbf{p}(0, 0, f_t, [W, H, Z'_t]) \mid \mathbf{c}_t]^\top + [\mathbf{n}_t^o \mid \mathbf{n}_t^c]^\top, \tag{3.13}$$

where $\mathbf{n}_t^o$ and $\mathbf{n}_t^c$ are zero-mean Gaussian noise processes on the object and camera measurements, respectively. Z'_t is the projection of the depth Z_t in the new coordinate system resulting from the pan and tilt of the camera. $\mathbf{p}(\phi_t, \theta_t, f_t, \vec{o}_t)$ represents the projection of the object position $\vec{o}_t$ into the image plane and, similarly, $\mathbf{p}(0, 0, f_t, [W, H, Z'_t])$ is the projection of the known object size $W \times H$ into the image plane. The camera vector $\vec{c}_t$ consists of the pan angle, tilt angle and focal length, as estimated by the state vector.

Given the system update and measurement processes defined in equations (3.10) and (3.13), the Bayesian estimation problem is to find an estimate of the unknown state $\vec{s}_t$ that maximizes the posterior density $p(\vec{s}_t|\vec{z}_{1:t})$.

Towards this end, an extended Kalman filter (EKF) is used to recursively solve this estimation problem [125]. The extended Kalman filter (EKF) approximates the likelihood as a Gaussian density with argument $\vec{s}_t$, mean $\vec{m}_t$ and covariance $\mathbf{P}_t$:

$$p(\vec{s}_t|\vec{z}_{1:t}) \approx \mathcal{N}(\vec{s}_t; \vec{m}_t, \mathbf{P}_t). \tag{3.14}$$

Defining $\hat{\mathbf{H}}_t$ as a local linearization, given by the Jacobian, of the nonlinear measurement function, $\mathbf{h}(\vec{s}_t)$:

$$\hat{\mathbf{H}}_t = \left. \frac{\partial \mathbf{h}(\vec{s}_t)}{\partial \vec{s}_t} \right|_{\vec{s}_t = \vec{m}_{t|t-1}}, \tag{3.15}$$

the update from time $t-1$ to time t is given by the following set of equations:

$$\vec{m}_{t|t-1} = \mathbf{U}\vec{m}_{t-1} \tag{3.16}$$
$$\mathbf{P}_{t|t-1} = \mathbf{Q} + \mathbf{U}\mathbf{P}_{t-1}\mathbf{U}^{\top} \tag{3.17}$$
$$\vec{m}_t = \vec{m}_{t|t-1} + \mathbf{K}_t(\vec{z}_t - \mathbf{h}(\vec{m}_{t|t-1})) \tag{3.18}$$
$$\mathbf{P}_t = \mathbf{P}_{t|t-1} - \mathbf{K}_t\hat{\mathbf{H}}_t\mathbf{P}_{t|t-1} \tag{3.19}$$
$$\mathbf{S}_t = \hat{\mathbf{H}}_t\mathbf{P}_{t|t-1}\hat{\mathbf{H}}_t^{\top} + \mathbf{R} \tag{3.20}$$
$$\mathbf{K}_t = \mathbf{P}_{t|t-1}\hat{\mathbf{H}}_t^{\top}\mathbf{S}_t^{-1}. \tag{3.21}$$

$\mathbf{S}_t$ is the covariance of the innovation term $\vec{z}_t - \mathbf{h}(\vec{m}_{t|t-1})$ and $\mathbf{K}_t$ is the Kalman gain. $\mathbf{Q}$ and $\mathbf{R}$ are the covariance of the Gaussian noise added to the system update and measurement, respectively.

3.4 Control

The estimated state outputted at each step of the filter is used to control the movement of the camera. Two PID controllers are used: one for controlling the pan and tilt and another one for the zoom. The control signal, outputted by a PID controller, is given by:

$$\vec{u}(t) = K_p\vec{e}(t) + K_i\int_0^t \vec{e}(\tau)\,d\tau + K_d\frac{d}{dt}\vec{e}(t), \tag{3.22}$$

where $\vec{e}(t)$ is the error signal, K_p is the proportional gain, K_i is the integral gain and K_d is the derivative gain.

In our case, and at each time t, the *error in pan* is defined as the difference between the estimated pan angle and the estimated horizontal angle that the object forms with the world coordinate system, while the *error in tilt* is defined as the difference between the estimated tilt angle and the estimated vertical angle of the object:

$$e_{pan} = \arctan(X_t/Z_t) - \phi_t, \tag{3.23}$$
$$e_{tilt} = \arctan(Y_t/Z_t) - \theta_t. \tag{3.24}$$

The gains are experimentally set to: $K_p = 1$, $K_i = 0$ and $K_d = 0.2$.

To calculate the *error for the zoom controller*, we define the desired area D_a, which is the maximum area in pixels we aim to have and which is usually achieved when the object is static. The error is then defined, at each time t, as:

$$e_{zoom} = D_a - w_{proj} * h_{proj}, \tag{3.25}$$

where w_{proj} and h_{proj} are the projections of the width W and height H of the object in the image plane. The gains are experimentally set to: $K_p = 0.01$, $K_i = 0$ and $K_d = 0$.

The integral phase was bypassed in both controllers, by setting K_i to 0, because the output of the filter was found to be accurate at steady state, i.e. when the object is centered with maximum zoom.

The error e_{zoom} is considered only when both $|e_{pan}|$ and $|e_{tilt}|$ are constant or decreasing; otherwise, a zoom out operation is executed.

3.5 Experimental Results

In this part, we demonstrate the performance of the system on both simulated scenarios and live scenes of a PTZ camera. The simulated scenario consisted of a random motion of an object whose size is 10×10 cm, and the error was averaged over many runs. The camera used in the live scenes was an Axis 214 PTZ network camera.

3.5.1 Simulated Data

The error metric we used in all model parameters estimation is the root mean square deviation (RMSD) defined as:

$$\text{RMSD}(\eta_i) \quad = \quad \sqrt{E((\bar{\eta_i} - \eta_i)^2)}, \tag{3.26}$$

where η_i is one of the model parameters, $[X, Y, Z, \phi, \theta, f, \dot{X}, \dot{Y}, \dot{Z}, \dot{\phi}_t, \dot{\theta}_t, \dot{f}_t]$, composing the state vector in equation 3.5, and $\bar{\eta_i}$ is the estimated model parameter. The expectation, E, is taken over the entire sequence. The RMSD is measured for several runs of the simulation (we used 100 runs in our experiments), and the average RMSD is used as a measure of estimation performance.

Figure 3.2 shows a box-and-whisker summary of the RMSD for a simulation where a moving object is tracked by a moving camera. In these experiments, we simulate the motion the camera would execute due to corrections coming from the PID controller described in the previous section. Also, some noise is introduced in the different state parameters. To investigate sensitivity to varying measurement noise, this value is scaled by a constant $a \in \{1, 5, 10\}$. Similar results can be seen in figure 3.3 for camera parameters estimation. From these figures, one can conclude that scaling the uncertainty, by $a = 5$ and $a = 10$, predictably scales the RMSD error as well as the spread (most notably in Z and f) and increases outliers. However, even with such increase, the estimates of both the object position and the camera parameters show robustness to noise.

3.5.2 Live Cameras

A commodity PTZ camera (Axis 214) was used for tracking different objects. Simple assumptions about object sizes were made: the cup tracked in figure 3.4 is assumed to be 8×12 cm, while the faces in figure 3.5 and figure 3.6 are assumed to be 18×18 cm. For the detection of the blue cup, a simple heuristics-based classifier for detecting blue regions in the normalized RGB colorspace was used; while for face detection, we used our work presented in the previous chapter. The two red dots represent the center of the object and the upper left corner, outputted by the detection process. The green dots represent the projection of the estimates of the center and the bounding box position, which are outputted by the estimation process. The tracker was able to successfully follow the objects taking correct decisions on when to zoom in and when to zoom out.

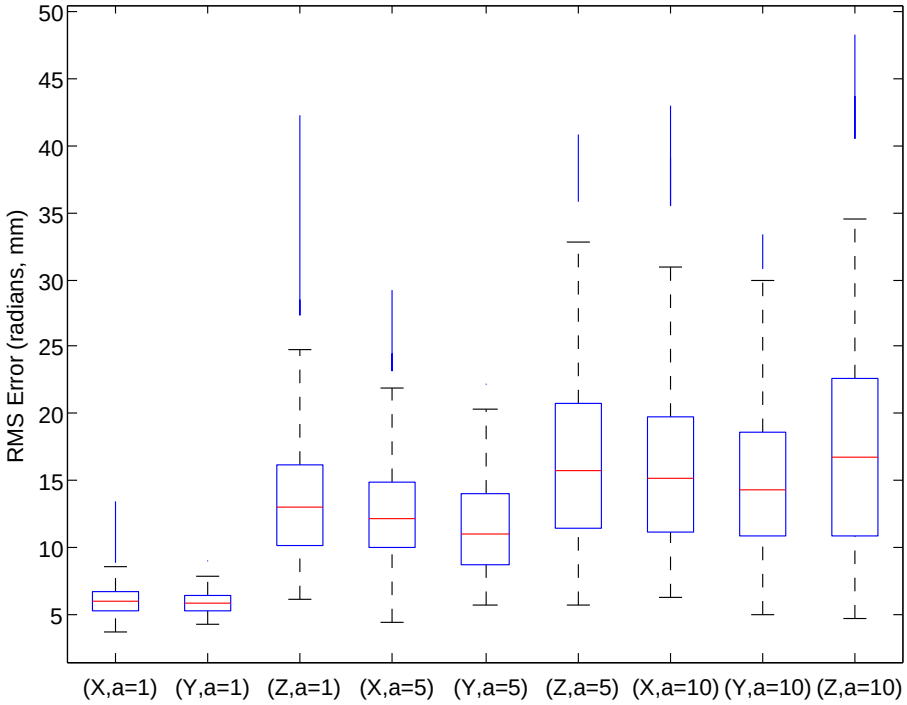


Figure 3.2: Error in 3D position parameters (X, Y, Z), measured in millimeters.

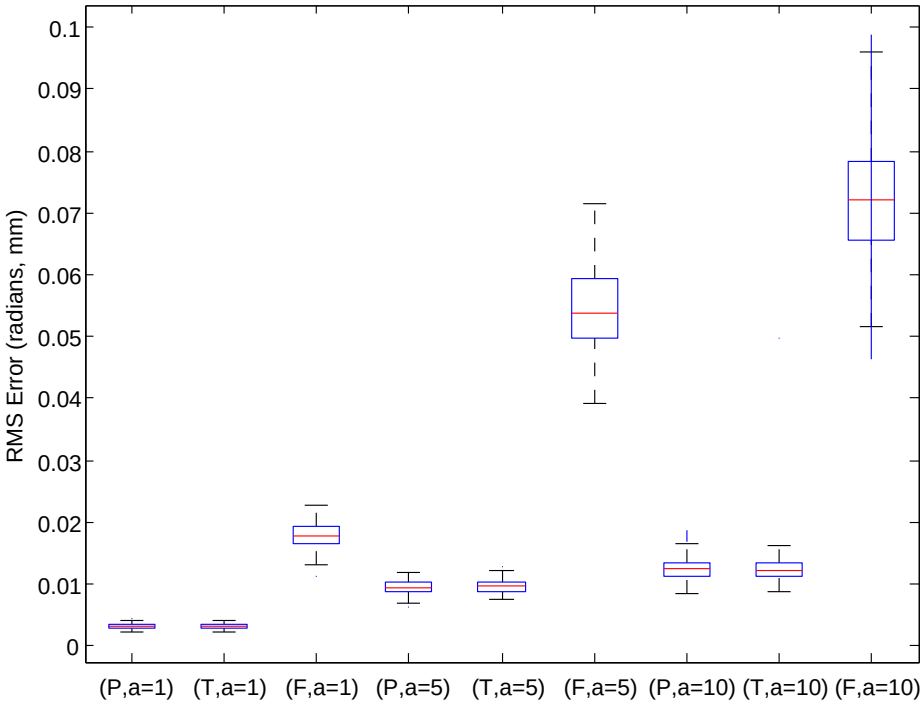
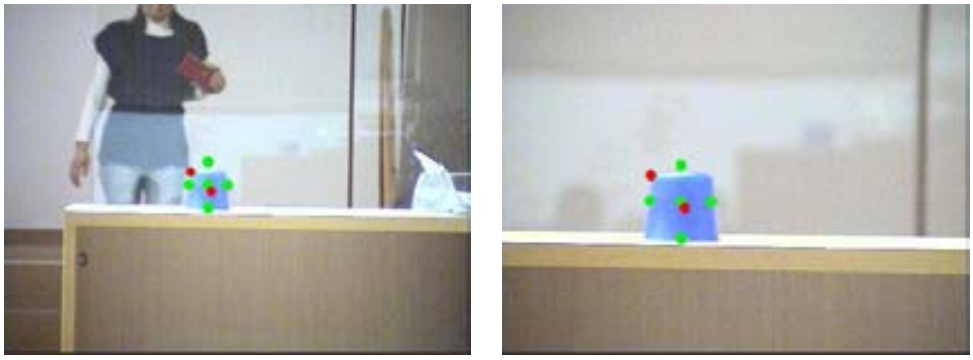


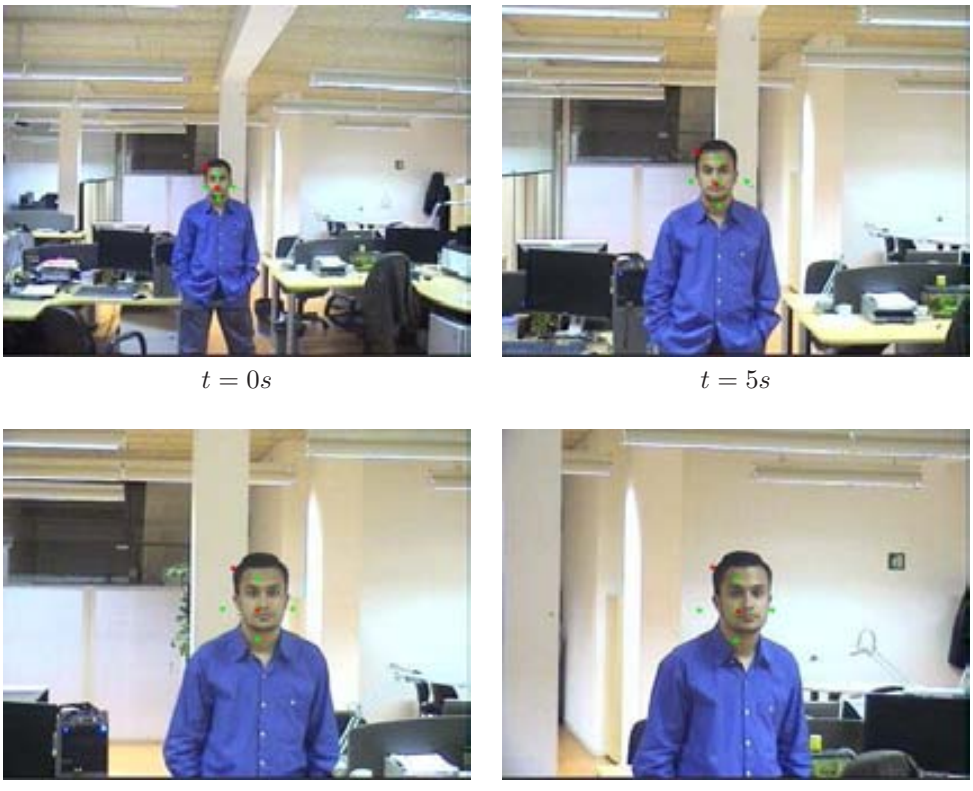
Figure 3.3: Error in pan angle, tilt angle and focal length. Angles are measured in radians, focal length in millimeters.

$t = 0s$ $t = 5s$



$t = 7.5s$ $t = 12.5s$

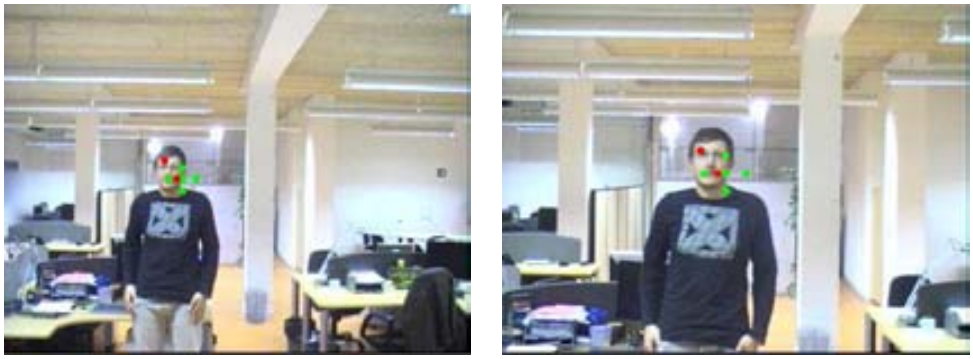
Figure 3.4: Reactive tracking of a stationary object. This figure is best viewed in color.



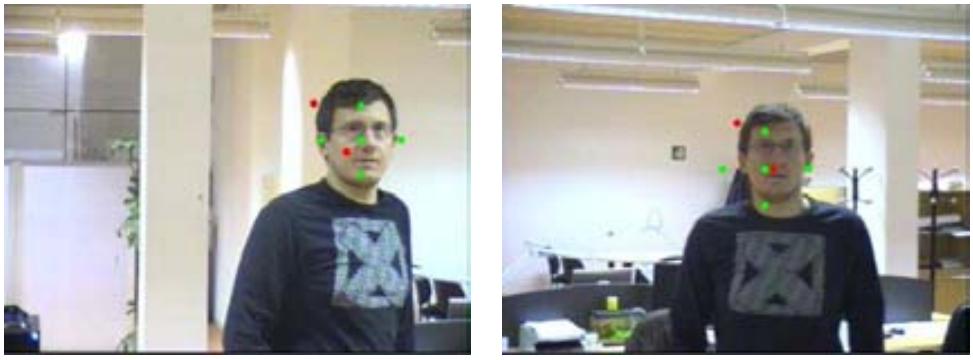
$t = 0s$ $t = 5s$

$t = 12.5s$ $t = 32.5s$

Figure 3.5: Reactive tracking of a moving face. This figure is best viewed in color.



$t = 0s$ $t = 5s$



$t = 12.5s$ $t = 32.5s$

Figure 3.6: Another example of reactive face tracking. This figure is best viewed in color.

3.6 Closing Remarks

In this chapter, a method for reactive object tracking has been described. The system uses a single PTZ camera and jointly estimates, in a Bayesian framework, the orientation and focal length of the camera in addition to the position of the tracked object in the 3D environment. The output of the estimation process is used to drive the control process, allowing the camera to reactively track the moving target. The main limitation of this method is that the output is dependent on the detection, i.e. the measurement process; therefore, and although the method is tolerant to measurement noise, continuous erroneous detection leads to inaccurate tracking. Also, this method does not support multiple objects tracking. Other than that, the estimates are robust in the presence of camera motion and increased measurement noise.

The problem of covering relatively large scenarios with surveillance cameras, in such a manner that the targets of interest are captured with sufficient resolution, is still nowadays an open and active research field. Whereas static camera networks are expensive and hard to manage and scale, active vision appears as a more natural solution to minimize the number of sensors while tackling the aforementioned goal.

Nevertheless, balancing the trade-off between area coverage and resolution per target calls for sensible techniques to control, integrate, and coordinate the possible passive and active components of an active vision system. The most interesting goal in the future of active vision is the control of zoom based on semantics and responding to uncertainty, in particular uncertainties and ambiguities due to high-level interpretations. Semantics-driven control of active cameras will allow a computer vision system to better detect, track and reason about the object it is monitoring.

Chapter 4

Head Pose Estimation

"The mind, once expanded to the dimensions of larger ideas, never returns to its original size."

Oliver Wendell Holmes.

Head pose estimation is a critical problem in many computer vision applications. These include human computer interaction, video surveillance, face and expression recognition. In most prior work on heads pose estimation, the positions of the faces on which the pose is to be estimated are specified manually. Therefore, the results are reported without studying the effect of misalignment. We propose a method based on partial least squares (PLS) regression to estimate pose and solve the alignment problem simultaneously. The contributions of this chapter are two-fold: 1) we show that the kernel version of PLS (kPLS) achieves better than state-of-the-art results on the estimation problem and 2) we develop a technique to reduce misalignment based on the learned PLS factors.

4.1 Introduction

Head pose is an extremely powerful communication tool that conveys important non-verbal messages about subjects. The work of Langton *et al.* [58] showed that head pose is highly correlated with gaze estimation. Like in face detection, the main challenges to accurate head pose estimation include: presence or absence of structural components (beards, mustaches, glasses, ...), facial expressions, occlusion, image orientation, and imaging conditions. Numerous papers have been published describing algorithms for head pose estimation and a good recent survey can be found in [77]. It divides the different methods into categories, including: appearance template methods such as [80], detector array methods where a dedicated face detector is trained

This work is done in collaboration with Prof. Larry Davis at the University of Maryland and published in CVPR12 [6].

for every pose as in [138], regression methods like [35], manifold embedding as [91] and geometric methods akin to [118].

The most successful methods for monocular head pose estimation are those using nonlinear regression [48, 77]. Work in this area include neural networks with locally linear maps [89] and multilayer perceptrons [107], in addition to support vector machine regression after PCA projection [64]. However, these nonlinear regression methods are especially sensitive to alignment errors; therefore, their performance diminishes with small localization inaccuracies.

Alignment is a well-known problem in many recognition algorithms and the authors of [49] attribute the scarcity of fully automated recognition systems to the difficulty of alignment. Alignment is by now well understood as a major subproblem of face recognition [122]. However, it is rarely considered in evaluations of pose estimation; results are typically reported on manually aligned data. A notable exception is Murphy-Chutorian and Trivedi [78]. They developed a system for measuring the position and orientation of a driver's head, and propose the use of localized gradient orientation (LGO) histograms to offset some of the localization error of the underlying face detector.

In this chapter, we present a regression-based pose estimation method that achieves better than state-of-the-art results and handles misalignment effects without the need to include any misaligned sample during training. Given a set of candidate windows from a noisy face detector, we develop a technique that predicts which of those windows is best aligned with the model based on partial least squares (PLS) analysis. The best aligned window is the one to which the pose regression coefficients are then applied. The remainder of the chapter is organized as follows: section 4.2 discusses both linear and kernel PLS regression methods; section 4.3 shows the results of the two methods on Pointing'04 and CMU Multi-PIE databases. In section 4.4, we show how PLS can be used to deal with misalignment as well as demonstrate the results of this framework on simulated noisy detections, and section 4.5 concludes the chapter.

4.2 Partial Least Squares

Although it has been more than three decades since its introduction [126] and more than two decades since its use in the domain of chemometrics [31], partial least squares (PLS) analysis has only recently been attracting attention in computer vision [24, 38, 98]. In its most general form, PLS models the relationship between sets of observed variables by projecting them into a latent space; hence, some researchers refer to PLS as "Projection to Latent Structures". The modeling is done by selecting orthogonal score vectors (a.k.a. latent vectors) that maximize the covariance between the different sets of variables while, at the same time, keeping most of the variance of each set. PLS can be effectively applied to solve regression problems where the number of samples is less than the number of independent variables, as well as in the presence of high collinearity of those variables.

4.2.1 Linear PLS Regression

Consider a matrix of independent variables $\mathbf{X}$ formed from n observations of N dimensional vectors and a matrix of dependent variables $\mathbf{Y}$ obtained as a response to $\mathbf{X}$ and formed of n observations of M dimensional vectors. PLS decomposes the zero-mean nxN matrix $\mathbf{X}$ and the zero-mean nxM matrix $\mathbf{Y}$ as follows:

$$\mathbf{X} = \mathbf{T}\mathbf{P}^T + \mathbf{E} \tag{4.1}$$
$$\mathbf{Y} = \mathbf{U}\mathbf{Q}^T + \mathbf{F} \tag{4.2}$$

where $\mathbf{T}$ and $\mathbf{U}$ are nxd matrices of the d extracted score vectors, i.e. d factors or components. The Nxd matrix $\mathbf{P}$ and the Mxd matrix $\mathbf{Q}$ represent the loadings. The nxN matrix $\mathbf{E}$ and the nxM matrix $\mathbf{F}$ are residual matrices. There exist many methods to obtain the decomposition in equations 4.1 and 4.2, the most classical of which is based on the nonlinear iterative partial least squares (NIPALS) algorithm [126], which finds normalized weights $\mathbf{w}$ and $\mathbf{c}$ that maximize the covariance between the score vectors $\mathbf{t}$ and $\mathbf{u}$. In the modification proposed in [62], the normalization of $\mathbf{t}$ and $\mathbf{u}$, rather than the normalization of $\mathbf{w}$ and $\mathbf{c}$, is used and the computation is done in d-iterations where each iteration is as follows:

1. randomly initialize $\mathbf{u}$;
2. $\mathbf{w} = \mathbf{X}^T\mathbf{u}$; $\mathbf{t} = \mathbf{X}\mathbf{w}$; $\mathbf{t} \leftarrow \mathbf{t}/\|\mathbf{t}\|$;
3. $\mathbf{c} = \mathbf{Y}^T\mathbf{t}$; $\mathbf{u} = \mathbf{Y}\mathbf{c}$; $\mathbf{u} \leftarrow \mathbf{u}/\|\mathbf{u}\|$;
4. repeat steps 2-3 until convergence;
5. deflate $\mathbf{X}$: $\mathbf{X} \leftarrow \mathbf{X} - \mathbf{t}\mathbf{t}^T\mathbf{X}$; deflate $\mathbf{Y}$: $\mathbf{Y} \leftarrow \mathbf{Y} - \mathbf{t}\mathbf{t}^T\mathbf{Y}$;

The matrices $\mathbf{T}$, $\mathbf{U}$, $\mathbf{W}$ and $\mathbf{C}$ are formed by columns of of the vectors $\mathbf{t}$, $\mathbf{u}$, $\mathbf{w}$ and $\mathbf{c}$ respectively, obtained at every iteration.

Once the two sets of variables, $\mathbf{X}$ and $\mathbf{Y}$, are projected to latent subspaces, what is left is to find the NxM regression coefficients matrix $\mathbf{B}$ such that:

$$\mathbf{Y} = \mathbf{X}\mathbf{B} + \mathbf{F}^* \tag{4.3}$$

where $\mathbf{F}^*$ is a residual matrix. From [93], $\mathbf{B}$ can be computed as follows:

$$\mathbf{B} = \mathbf{W}(\mathbf{P}^T\mathbf{W})^{-1}\mathbf{C}^T \tag{4.4}$$

where the following equalities hold:

$$\mathbf{W} = \mathbf{X}^T\mathbf{U} \tag{4.5}$$
$$\mathbf{P} = \mathbf{X}^T\mathbf{T}(\mathbf{T}^T\mathbf{T})^{-1} \tag{4.6}$$
$$\mathbf{C} = \mathbf{Y}^T\mathbf{T}(\mathbf{T}^T\mathbf{T})^{-1}. \tag{4.7}$$

Given the orthonormality of $\mathbf{T}$, i.e. $\mathbf{T}^\mathbf{T}\mathbf{T} = \mathbf{I}$, and substituting equations 4.5, 4.6 and 4.7 to 4.4, $\mathbf{B}$ can be expressed as:

$$\mathbf{B} = \mathbf{X}^T\mathbf{U}(\mathbf{T}^T\mathbf{X}\mathbf{X}^T\mathbf{U})^{-1}\mathbf{T}^T\mathbf{Y}. \tag{4.8}$$

The NIPALS algorithm can be executed in a manner involving only matrix-vector multiplications, rendering the complexity in the order of $O(n^2)$.

4.2.2 Kernel PLS

Consider a nonlinear transformation of each input vector $\mathbf{x}$ into a feature space $\mathcal{F}$, i.e. mapping Φ: $\mathbf{x}_i \in R^N \rightarrow \Phi(\mathbf{x}_i) \in \mathcal{F}$. Denoting all the mapped vectors $\mathbf{x}$, i.e. $\{\Phi(\mathbf{x}_i)\}_{i=1}^n$, by $\mathbf{\Phi}$ and using the theory of Reproducing Kernel Hilbert Spaces (RKHS) [93], the kernel NIPALS algorithm is:

1. randomly initialize $\mathbf{u}$;
2. $\mathbf{t} = \mathbf{\Phi}\mathbf{\Phi}^T\mathbf{u}$; $\mathbf{t} \leftarrow \mathbf{t}/\|\mathbf{t}\|$;
3. $\mathbf{c} = \mathbf{Y}^T\mathbf{t}$; $\mathbf{u} = \mathbf{Y}\mathbf{c}$; $\mathbf{u} \leftarrow \mathbf{u}/\|\mathbf{u}\|$;
4. repeat steps 2-3 until convergence;
5. deflate $\mathbf{\Phi}\mathbf{\Phi}^T$: $\mathbf{\Phi}\mathbf{\Phi}^T \leftarrow (\mathbf{\Phi} - \mathbf{t}\mathbf{t}^T\mathbf{\Phi})(\mathbf{\Phi} - \mathbf{t}\mathbf{t}^T\mathbf{\Phi})^T$; deflate $\mathbf{Y}$: $\mathbf{Y} \leftarrow \mathbf{t}\mathbf{t}^T\mathbf{Y}$;

Using kernel mapping $K(.)$ and the "kernel trick", one can notice that $\mathbf{\Phi}\mathbf{\Phi}^\mathbf{T}$ represents the kernel Gram matrix $\mathbf{K}$ of the cross dot products between all mapped input, $\{\Phi(\mathbf{x}_i)\}_{i=1}^n$. The deflation of $\mathbf{\Phi}\mathbf{\Phi}^T$ in step 5 is now given by:

$$\mathbf{K} \leftarrow (\mathbf{I} - \mathbf{t}\mathbf{t}^T)\mathbf{K}(\mathbf{I} - \mathbf{t}\mathbf{t}^T) \tag{4.9}$$

$$\mathbf{K} \leftarrow \mathbf{K} - \mathbf{t}\mathbf{t}^T\mathbf{K} - \mathbf{K}\mathbf{t}\mathbf{t}^T + \mathbf{t}\mathbf{t}^T\mathbf{K}\mathbf{t}\mathbf{t}^T \tag{4.10}$$

and the regression coefficients by:

$$\mathbf{B} = \mathbf{\Phi}^T\mathbf{U}(\mathbf{T}^T\mathbf{K}\mathbf{U})^{-1}\mathbf{T}^T\mathbf{Y}. \tag{4.11}$$

In our experiments, an rbf kernel was used; therefore, the complexity becomes in the order of $O(n^2N)$, as opposed to quadratic order required by NIPALS: $O(n^2)$ in the linear case.

4.2.3 PLS, MLR and PCR

Like any regression method, the aim of PLS is to find a set of coefficients modeling the relationship between the input data $\mathbf{X}$ and its response $\mathbf{Y}$. Other relevant regression methods include Multiple Linear Regression (MLR) and Principal Component Regression (PCR). MLR solves for the regression coefficients directly by establishing a linear relationship between the input and the output. MLR cannot be applied when the inverse of $\mathbf{X}\mathbf{X}^T$ does not exist. PCR determines the coefficients based on the score (or latent) vectors after projecting $\mathbf{X}$ to a subspace determined by the principal components. Since these components are computed only on $\mathbf{X}$, without any consideration of $\mathbf{Y}$, some of them might be irrelevant in predicting the response. PLS projects both $\mathbf{X}$ and $\mathbf{Y}$ each to its latent subspace before computing the regression coefficients. This can be seen in Figure 4.1. As a rule of thumb, MLR models the maximum correlation between $\mathbf{X}$ and $\mathbf{Y}$, PCR models the maximum variance in $\mathbf{X}$ while PLS models the maximum covariance between $\mathbf{X}$ and $\mathbf{Y}$.

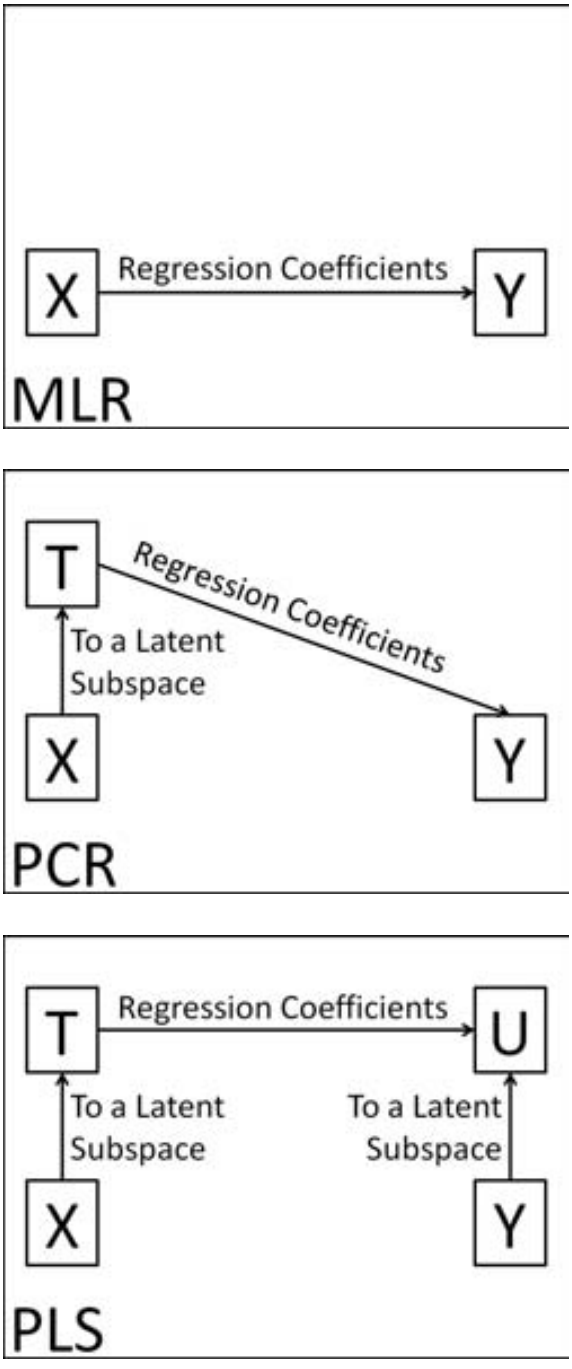


Figure 4.1: MLR, PCR and PLS coefficients calculation.

4.3 Head Pose Estimation

In this section, we will show the results of applying linear and kernel PLS regression to estimate the head pose in two datasets: Pointing'04 and CMU Multi-PIE. We will compare the results with state-of-the-art methods. The feature vector, for each face, is composed of 3-level pyramid Histogram of Oriented Gradients (HOG) extracted from the bounding box and quantized into 8 bins. Therefore, each row of the independent variable $\mathbf{X}$ is composed of 680 dimensions representing the HOG features of the corresponding face while each row of the dependent variable $\mathbf{Y}$ is composed of the corresponding pose; it is two dimensional for Pointing'04 since the dataset contains values for both pitch and yaw while one dimensional, yaw, for CMU Multi-PIE.

4.3.1 Results on Pointing'04

Per subject, the Pointing'04 database [35] contains poses discretized to 9 angles of pitch: $\{-90°, -60°, -30°, -15°, 0°, 15°, 30°, 60°, 90°\}$ and 13 angles of yaw: $\{-90°, -75°, -60°, -45°, -30°, -15°, 0°, 15°, 30°, 45°, 60°, 75°, 90°\}$. However, when the pitch angle is $-90°$ or $90°$, the yaw angle is always $0°$. Therefore, the total number of poses is: $7x13 + 2x1 = 93$ poses. The total number of subjects is 15, each of whom is photographed twice resulting in 2790 images forming the database. The bounding box containing the face for each image is provided. As indicated before, the features that were used in these experiments are 3-level pyramid HOG and 5-level cross validation is employed, where every sample is tested using a model trained on 80% of the remaining samples. The optimal number of factors was found to be 25 for linear PLS and 40 for kPLS. For kPLS, a radial basis function (rbf) kernel was used with a kernel width of $\sigma = 0.05$. A comparison between PLS and other state-of-the-art methods is shown in table 4.1. It is interesting to see that kPLS outperforms all other methods while, at the same time, reducing the feature space significantly, from 680 dimensions to 40 latent dimensions. The error shown in table 4.1 is the mean absolute error (MAE) between the continuous predicted pose and the discrete ground truth pose.

The yaw 'box and whisker' plot for linear PLS regression is shown in figure 4.2, while that for pitch is shown in figure 4.3. It can be seen that the predictions at boundary cases, i.e. poses $-90°$ and $90°$ for yaw and pitch, are less accurate than the rest. This is expected especially that, for pitch, the number of training samples at those poses are much smaller than the others. However, the overall accuracy is good. The improvement of kPLS over linear PLS shown in table 4.1 can be further demonstrated in the kPLS 'box and whisker' plots for yaw and pitch. The yaw 'box and whisker' plot for kPLS regression is shown in figure 4.4, while that for pitch is shown in figure 4.5. Looking at all the poses in pitch and yaw for kPLS, one concludes that the regression is able to accurately predict head pose with little variance and few outliers, performing better than the linear case. To justify the kPLS parameters, i.e. the number of factors and the rbf kernel width, the mean absolute error vs. the number of factors is shown in figure 4.6 and vs. the rbf kernel width in figure 4.7, justifying the 40 factors and $\sigma = 0.05$.

Method	Yaw Error	Pitch Error	Accuracy (Yaw,Pitch)	Notes
Ours (kernel PLS)	**6.56°**	**6.61°**	**(67.36%, 80.36%)**	-
Stiefelhagen [106]	9.5°	9.7°	(52.0%, 66.3%)	1
Ours (linear PLS)	**11.29°**	**10.52°**	**(45.57%, 58.70%)**	-
Human Performance [36]	11.8°	9.4°	(40.7%, 59.0%)	2
Gourier (Associative Memories) [36]	10.1°	15.9°	(50.0%, 43.9%)	3
Tu (High-order SVD) [111]	12.9°	17.97°	(49.25%, 54.84%)	4
Tu (PCA) [111]	14.11°	14.98°	(55.20%, 57.99%)	4
Tu (LEA) [111]	15.88°	17.44°	(45.16%, 50.61%)	4
Voit [116]	12.3°	12.77°	–	-
Li (PCA) [65]	26.9°	35.1°	–	5
Li (LDA) [65]	25.8°	26.9°	–	5
Li (LPP) [65]	24.7°	22.6°	–	5
Li (Local-PCA) [65]	24.5°	37.6°	–	5
Li (Local-LPP) [65]	29.2°	40.2°	–	5
Li (Local-LDA) [65]	19.1°	30.7°	–	5

Notes:
1) Used 80% of Pointing'04 images for training, 10% for cross-evaluation, and 10% for testing.
2) Human performance with training.
3) Best results over different reported methods.
4) Better results have been obtained with manual localization.
5) Results for 32-dim embedding.

Table 4.1
Comparison of our PLS results to state-of-the-art methods (from [77]) in terms of mean absolute error and classification accuracy.

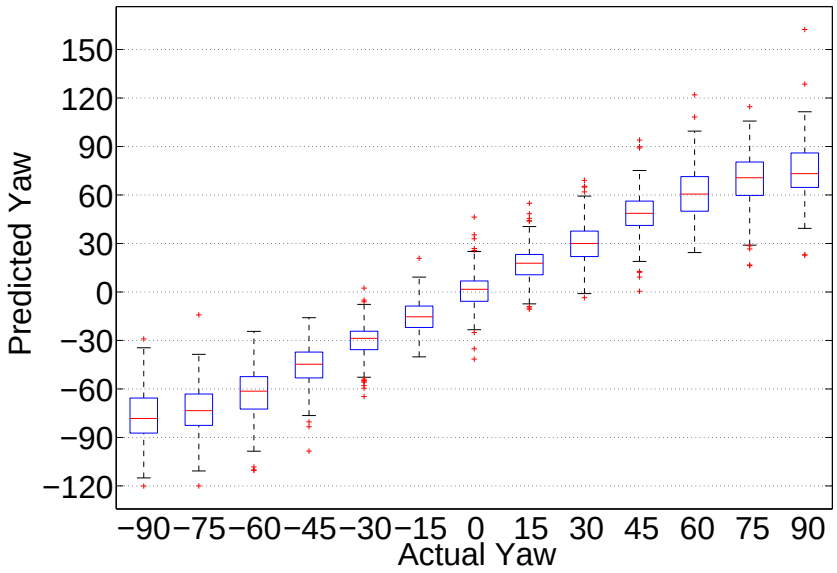


Figure 4.2: Detailed results of linear PLS on Pointing'04 yaw regression.

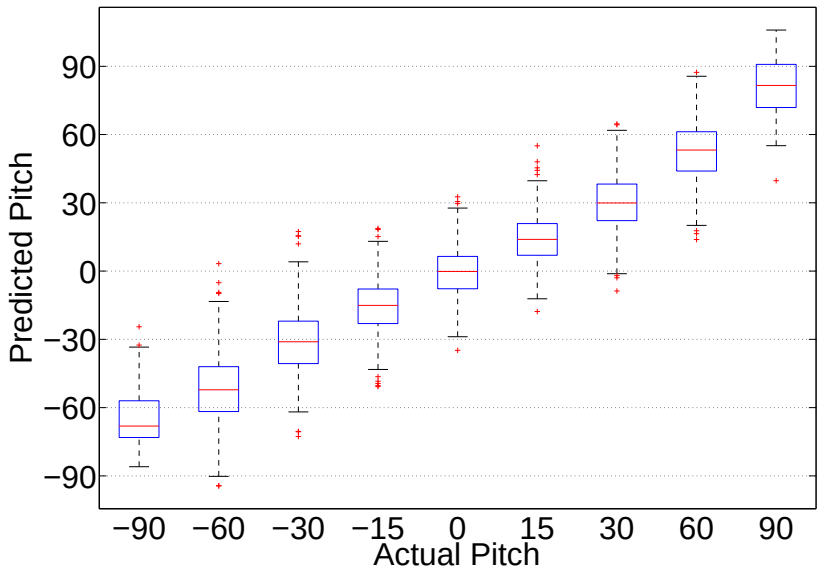


Figure 4.3: Detailed results of linear PLS on Pointing'04 pitch regression.

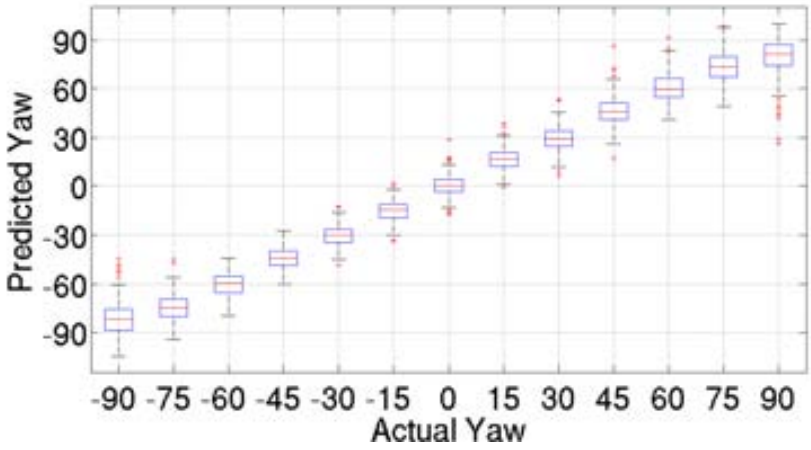


Figure 4.4: Detailed results of kPLS on Pointing'04 yaw regression.

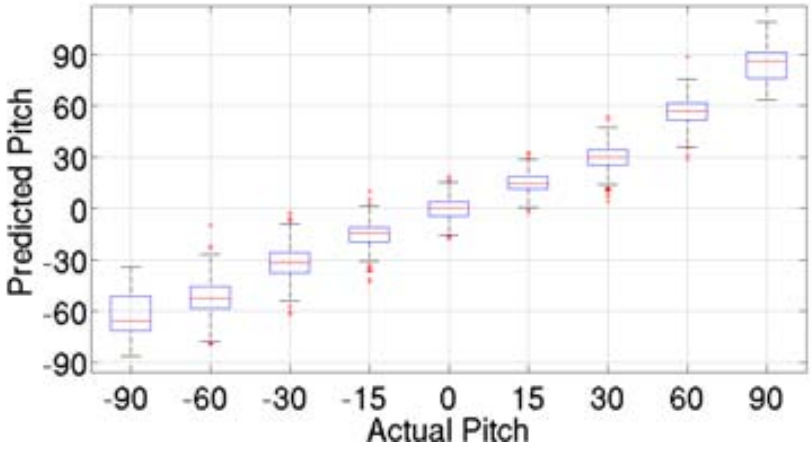


Figure 4.5: Detailed results of kPLS on Pointing'04 pitch regression.

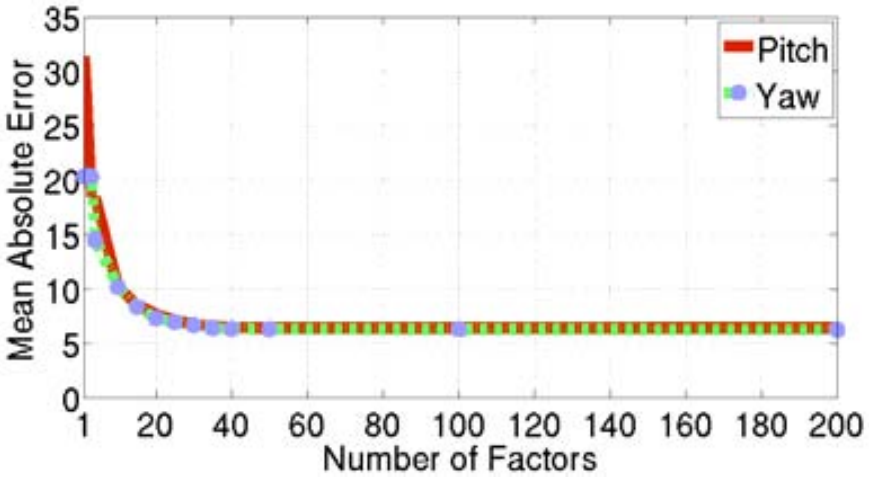


Figure 4.6: Mean absolute error vs. number of factors. This figure is best viewed in color.

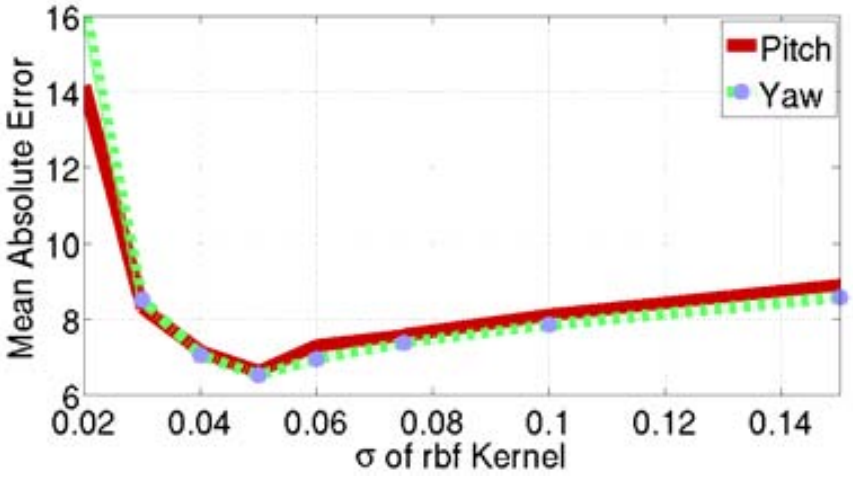


Figure 4.7: Mean absolute error vs. σ. This figure is best viewed in color.

Figure 4.8: Sample unnormalized cropped faces under different expressions from CMU Multi-PIE.

	kPLS	linear PLS	PCR
MAE	**5.31°**	9.11°	11.03°
Accuracy	**79.48%**	57.22%	48.33%

Table 4.2
Comparison of the different algorithms in terms of mean absolute error and classification accuracy.

4.3.2 Results on Multi-PIE

In this experiment, 2700 face images from the CMU Multi-PIE database [37] were manually annotated. These images belong to 144 subjects, under frontal illumination and varying expressions. Multi-PIE yaw angles range between $-90°$ and $90°$ with increments of $15°$ resulting in 13 discrete poses, i.e. the same discrete poses as the Pointing'04 database: $\{-90°, -75°, -60°, -45°, -30°, -15°, 0°, 15°, 30°, 45°, 60°, 75°, 90°\}$. Sampled cropped faces are shown in figure 4.8.

We employed a 2-fold cross validation, where one half of the data was used for training while the other half for testing and vice versa. No tuning was done for the database, so the same parameters that were found to optimize the results on Pointing'04 were used, i.e. 25 factors for linear PLS and 40 factors with $\sigma = 0.05$ for kernel PLS. The results for kPLS and linear PLS regression are shown in Table 4.2 along with those of PCR (also 25 factors were used). MLR could not be applied due to the multicollinearity in the data. kPLS achieved the best results with a mean absolute error of $5.31°$. Detailed yaw 'box and whisker' plot for linear PLS regression is shown in figure 4.9, while that for kPLS is shown in figure 4.10.

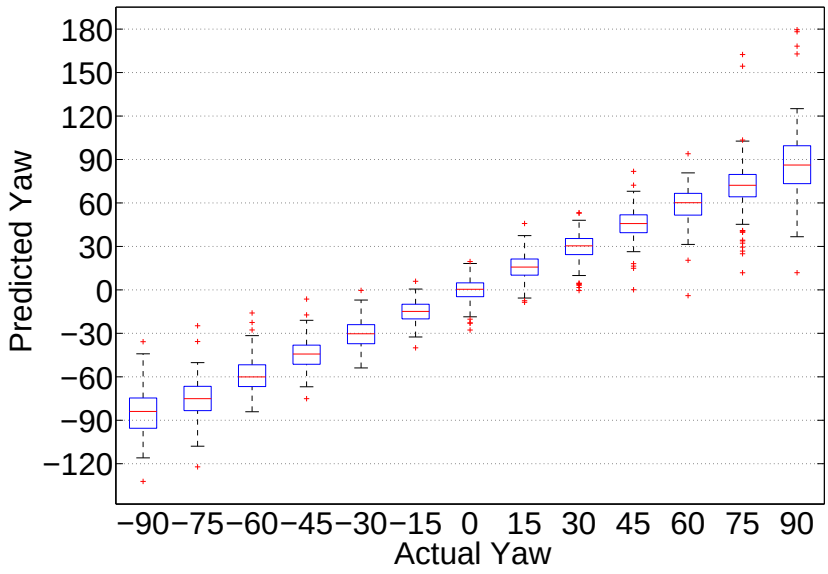


Figure 4.9: Detailed results of linear PLS on Multi-PIE yaw regression.

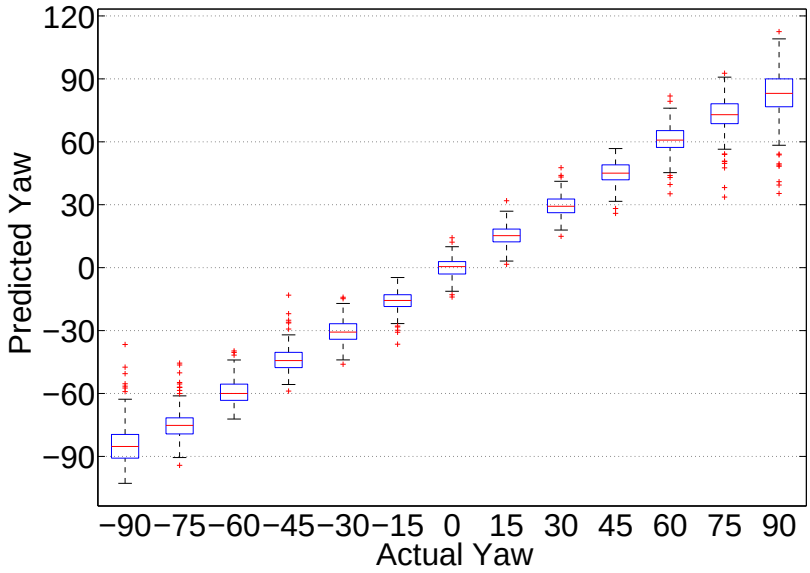


Figure 4.10: Detailed results of kPLS on Multi-PIE yaw regression.

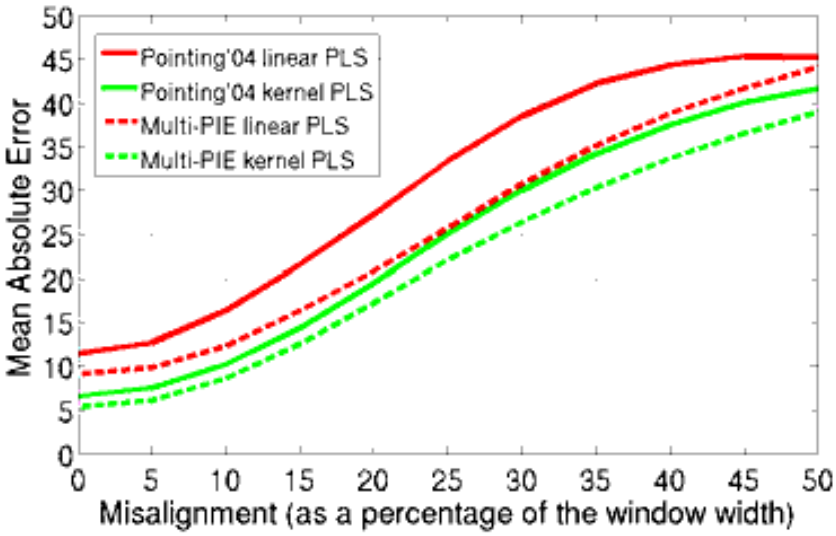


Figure 4.11: Mean absolute error vs. misalignment.

4.4 Misalignment

Regression (as well as classification) algorithms are generally sensitive to localization error. If the object is not accurately registered with the learned model, the comparison between the object features and the model features leads to errors. Most pose estimation methods are evaluated on well-annotated data and no analysis of the sensitivity to misalignment is typically reported. To study this problem, we conducted experiments where the training data is kept unchanged while the test data is regenerated from the images through shifting the face bounding box by a certain percentage of its width. The MAE in yaw pose estimation versus the shift percentage is shown in figure 4.11. As expected, the greater the misalignment, the worse the pose estimation results. It is also interesting to see that the effect in the kernel version is close to that in the linear version.

To deal with misalignment, we propose the following: given that $\mathbf{T}$ and $\mathbf{U}$ of the PLS model are correlated and given a set of noisy observations, the observation that produces the minimum residual when projected to the latent subspace of $\mathbf{X}$ should have the minimum error between its predicted response and actual response. Therefore, to estimate pose on a candidate face produced by a noisy detection process, we consider not only the detected location of the face but also a set of shifted versions of the face, and choose the instance in the set that produces the minimum residual when projected to the latent subspace. The estimated pose for that instant is the face pose. In the remainder of this section, we derive the equations to calculate this residual for both linear and kernel PLS, and show how regressing on the minimum residual instance can reduce misalignment problems.

4.4.1 Linear PLS Residual

Given a new vector $\mathbf{x}$ and a trained PLS model, as described in section 4.2, $\mathbf{x}$ can be approximated as:

$$\mathbf{x} \approx \mathbf{t}\mathbf{P}^T \tag{4.12}$$

where $\mathbf{t}$ is the score vector corresponding to $\mathbf{x}$ and $\mathbf{P}$ represents the learned loadings. Using equation 4.6 and the fact that $\mathbf{T}^\mathbf{T}\mathbf{T} = \mathbf{I}$, $\mathbf{x}$ can be rewritten as:

$$\mathbf{x} \approx \mathbf{t}\mathbf{T}^T\mathbf{X}. \tag{4.13}$$

To approximate $\mathbf{t}$, the following derivation can be made:

$$\mathbf{x}\mathbf{X}^T \approx \mathbf{t}\mathbf{T}^T\mathbf{X}\mathbf{X}^T \tag{4.14}$$
$$\mathbf{x}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1} \approx \mathbf{t}\mathbf{T}^T \tag{4.15}$$
$$\mathbf{x}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{T} \approx \mathbf{t}, \tag{4.16}$$

substituting equation 4.16 to equation 4.13, $\mathbf{x}$ becomes:

$$\mathbf{x} \approx \mathbf{x}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{T}\mathbf{T}^T\mathbf{X}, \tag{4.17}$$

and the residual is the error in the approximation, given by:

$$\mathbf{e} = \mathbf{x} - \mathbf{x}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{T}\mathbf{T}^T\mathbf{X}. \tag{4.18}$$

However, as we stated earlier $(\mathbf{X}\mathbf{X}^T)^{-1}$ might not exist due to multicollinearity in the data. Therefore, a better derivation, that makes use of equation 4.6, is:

$$\mathbf{x} \approx \mathbf{t}\mathbf{P}^T \tag{4.19}$$
$$\mathbf{x}\mathbf{P} \approx \mathbf{t}\mathbf{P}^T\mathbf{P} \tag{4.20}$$
$$\mathbf{x}\mathbf{P}(\mathbf{P}^T\mathbf{P})^{-1} \approx \mathbf{t} \tag{4.21}$$
$$\mathbf{x}\mathbf{X}^T\mathbf{T}(\mathbf{T}^T\mathbf{X}\mathbf{X}^T\mathbf{T})^{-1} \approx \mathbf{t}, \tag{4.22}$$

replacing equation 4.22 in equation 4.13, $\mathbf{x}$ becomes:

$$\mathbf{x} \approx \mathbf{x}\mathbf{X}^T\mathbf{T}(\mathbf{T}^T\mathbf{X}\mathbf{X}^T\mathbf{T})^{-1}\mathbf{T}^T\mathbf{X}, \tag{4.23}$$

and the residual is:

$$\mathbf{e} = \mathbf{x} - \mathbf{x}\mathbf{X}^T\mathbf{T}(\mathbf{T}^T\mathbf{X}\mathbf{X}^T\mathbf{T})^{-1}\mathbf{T}^T\mathbf{X}. \tag{4.24}$$

Therefore, for a candidate face position, a search is done in its vicinity to find the best aligned window, i.e. the one whose feature vector produces the minimum residual in equation 4.24. Starting with the original ground-truth face windows, we shifted each of these windows by 5% of its width in the four directions (up, down, right, left), creating a bag containing those four misaligned samples in addition to the original sample. We gradually increased the shifts from 5% to 50% of the width in steps of 5%; in each step, four new samples, whose misalignment is worse than

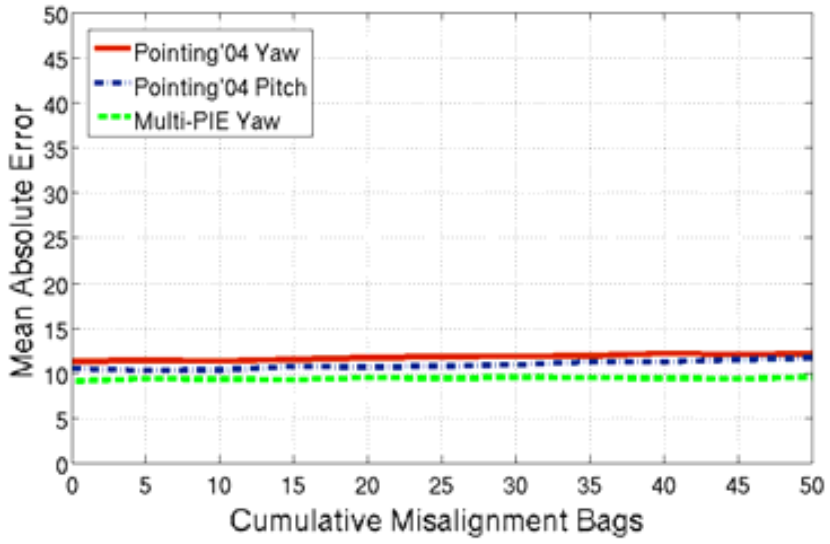


Figure 4.12: Mean absolute error as more misaligned samples are added to the bags (linear algorithm). This figure is best viewed in color.

the previous four, are added to each bag. At 50%, each bag contained 41 feature vectors (40 misaligned and the original). The MAE of applying the regression on the selected minimum residual sample of each bag, at each step, is shown in figure 4.12. Despite the huge increase in the added noise, the effect on our algorithm is negligible; the maximum difference in the lateral figure is for Pointing'04 pitch which goes from 10.52° at 0% shift to 11.68° at 50% shift.

4.4.2 kPLS Residual

Similar to the linear case, and given the formulation developed in subsection 4.2.2, the mapped version of $\mathbf{x}$ can be approximated as:

$$\Phi(\mathbf{x}) \approx \mathbf{t}\mathbf{T}^T\mathbf{\Phi}; \tag{4.25}$$

also, in a similar manner to deriving equation 4.16, $\mathbf{t}$ in this case can be approximated as:

$$\mathbf{t} \approx \Phi(\mathbf{x})\mathbf{\Phi}^T(\mathbf{\Phi}\mathbf{\Phi}^T)^{-1}\mathbf{T}, \tag{4.26}$$

and after the kernel mapping $K(.)$:

$$\mathbf{t} \approx K(\mathbf{x}, \mathbf{X})\mathbf{K}^{-1}\mathbf{T}. \tag{4.27}$$

Unlike the linear product, $\mathbf{X}\mathbf{X}^T$, we can assume that $\mathbf{K}$ is invertible due to the mapping to a much higher dimensional space that eliminates linear dependencies.

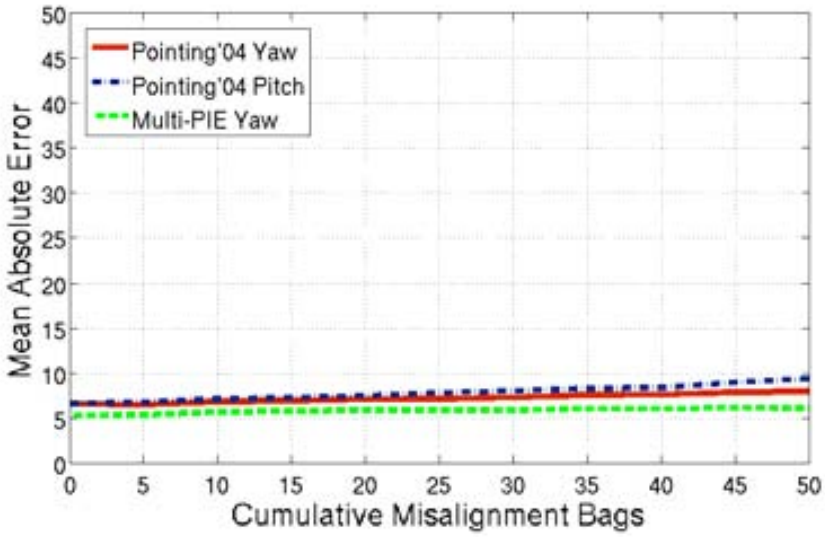


Figure 4.13: Mean absolute error as more misaligned samples are added to the bags (kernel algorithm). This figure is best viewed in color

Starting from equation 4.25, the following derivation can be made:

$$(\Phi(\mathbf{x}) - \mathbf{t}\mathbf{T}^T\mathbf{\Phi})(\Phi(\mathbf{x}) - \mathbf{t}\mathbf{T}^T\mathbf{\Phi})^T \approx 0 \tag{4.28}$$

$$(\Phi(\mathbf{x}) - \mathbf{t}\mathbf{T}^T\mathbf{\Phi})(\Phi^T(\mathbf{x}) - \mathbf{\Phi}^T\mathbf{T}\mathbf{t}^T) \approx 0 \tag{4.29}$$

$$\begin{aligned}\Phi(\mathbf{x})\Phi^T(\mathbf{x}) - \Phi(\mathbf{x})\mathbf{\Phi}^T\mathbf{T}\mathbf{t}^T - \mathbf{t}\mathbf{T}^T\mathbf{\Phi}\Phi^T(\mathbf{x}) \\ +\mathbf{t}\mathbf{T}^T\mathbf{\Phi}\mathbf{\Phi}^T\mathbf{T}\mathbf{t}^T \approx 0\end{aligned} \tag{4.30}$$

$$\begin{aligned}K(\mathbf{x},\mathbf{x}) - K(\mathbf{x},\mathbf{X})\mathbf{T}\mathbf{t}^T - \mathbf{t}\mathbf{T}^T K^T(\mathbf{x},\mathbf{X}) \\ +\mathbf{t}\mathbf{T}^T\mathbf{K}\mathbf{T}\mathbf{t}^T \approx 0.\end{aligned} \tag{4.31}$$

The same experimental setup as in the previous subsection is used here and the vector with minimum residual is defined as the one minimizing equation 4.31. The results are shown in figure 4.13, demonstrating that the method can also be successfully applied in kernel regression.

4.4.3 Comparison with MIL

Multiple instance learning (MIL), where a label is associated with a bag of instances rather than just a single instance, has been proposed to handle misalignment [13, 68]. We compare our algorithm against Multi-Instance Multi-Label SVM (MIMLSVM), which was shown to outperform other well-known multi-instance learning algorithms [140]. Our kPLS results obtained using bags with up to 50% shifts are compared against MIMLSVM on those same bags and the comparison is shown in

	Ours	MIMLSVM
MAE Pointing'04 Yaw	7.94°	10.72°
MAE Pointing'04 Pitch	9.35°	12.32°
MAE Multi-PIE Yaw	6.06°	5.40°

Table 4.3
Comparison between MIMLSVM and our kPLS method when dealing with misalignment

table 4.3. For MIMLSVM, each dataset was divided equally for training and testing and the number of medoids was set to 20% of the training data. After computing the Hausdorff distance, the SVM cost was set to 20 and its rbf kernel width to 90 (these values where experimentally found to provide the best results). Our method outperformed MIMLSVM on average, despite not having any misaligned sample in the training data. It is also worth mentioning that computing the Hausdorff distance on the training data took around 6 hours while our training process took around 3 minutes in total, on the same machine.

4.5 Closing Remarks

In summary, we presented a PLS-based regression method for head pose estimation that significantly reduces sensitivity to misalignment. The method outperforms state-of-the-art methods while simultaneously dealing with misaligned faces. Handling misalignment is done without any need for further training, i.e. it makes use of the same factors that are trained on well aligned faces. Even though no misaligned sample is included in the training, it shows better performance than MIMLSVM where training is done on bags of aligned and misaligned samples.

Chapter 5

Conclusions: (not) the end

"It only ends once.
Anything that happens before that ...
is just progress."

Man in black. Lost. Season 5 finale.

Faces are unique visual stimuli that encode rich information about the individual. Therefore, it is no surprise that studies show the presence of a dedicated neural network in the human brain for the detection of faces and their post-detection analysis. The complexity of this network explains the ease with which humans look at faces and encrypt the information they contain, e.g. identity, expression, etc. However, and despite many years of research, machines still fall short in mimicking this ability.

The applications benefiting from automatic facial analysis span a wide spectrum, such as video surveillance, human-computer interaction, content-based image retrieval, biometric identification, video coding and age/gender recognition. Research has been going on in many areas of facial analysis including: face detection, face recognition, pose estimation, face tracking and pose estimation. Despite the years of research and relative success, a complete solution is still out of reach.

This thesis is dedicated to faces and their automatic perception. As the title indicates, three research lines has been explored: face detection, reactive tracking and pose estimation. The contributions of this thesis to each of those lines are highlighted below.

- **Face Detection.** In this chapter, we showed our progress towards a face detector that is based on skin color segmentation to reduce the search space. We started by a first-generation rule-based model that uses pixel-based heuristics to detect skin regions and pre-defined thresholds to judge if a given skin blob is a face or not. We later studied different colorspaces to improve the skin detection algorithm using statistical models. Those models, alongside the trained skin detector, are used to achieve a second generation face detector that outperforms the Viola and Jones baseline detector, despite reducing the search space and using very small number of shape features.

- **Reactive Tracking.** Trying to establish a trade-off between the resolution per target and the area of coverage is a very important consideration in face (as well as any object) tracking. Robust estimations of where the camera is located at a certain instance and where the object is present at that instance can be exploited towards this end. However, noisy detections and inaccurate camera position parameters read from its hardware can negatively affect those estimations and thus challenging the tracking process. In this chapter, we present a framework that combines estimation and control in a joint scheme to track a face with a single pan-tilt-zoom camera. An extended Kalman filter is used to jointly estimate the object world coordinates and the camera position where the aim of the joint estimation is to increase robustness to noise, and the output is used to drive a PID controller in order to reactively track a face, taking correct decisions when to zoom-in on the face to maximize the size and when to zoom-out to reduce the risk of losing the target. While this work is mainly motivated by tracking faces, it can be easily applied atop of any detector to track different objects. In the experiments, we show its robustness to noise as well as its applicability in live scenarios.

- **Head Pose Estimation.** The most prominent part of this thesis is the work on head pose estimation. In most prior work on heads pose estimation, the positions of the faces on which the pose is to be estimated are specified manually. Therefore, the results are reported without studying the effect of misalignment. Regression, as well as classification, algorithms are generally sensitive to localization error. If the object is not accurately registered with the learned model, the comparison between the object features and the model features leads to errors. We proposed a method based on partial least squares (PLS) regression to estimate pose and simultaneously solve the alignment problem. One of the contributions of this work is demonstrating that the kernel version of PLS (kPLS) achieves better than state-of-the-art results on the estimation problem while the second is developing a technique to reduce misalignment based on the learned PLS factors. This technique is capable of out-performing multiple instance learning without the need to include any misaligned sample in the training set and the resulting complexity in the training process.

Appendix A

Publications

Book Chapter

- Murad Al Haj, Carles Fernández, Zhanwu Xiong, Ivan Huerta, Jordi Gonzàlez and Xavier Roca. "Beyond the Static Camera: Issues and Trends in Active Vision". In *Guide to Visual Analysis of Humans: Looking at People*, Chapter 2. T. Moeslund, A. Hilton, V. Krueger, L. Sigal (eds.), pages 11–30, Springer, 2011.

Journal

- Abhishek Sharma, Murad Al Haj, Jonghyun Choi, Larry S. Davis and David W. Jacobs. "Robust Pose Invariant Face Recognition using Coupled Latent Space Discriminant Analysis". In *Computer Vision and Image Understanding (CVIU)*, 116(11):1095–1110, November, 2012.

International Conferences

- Murad Al Haj, Jordi Gonzàlez and Larry S. Davis. "On Partial Least Squares in Head Pose Estimation: How to simultaneously deal with misalignment", In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, Providence, Rhode Island, USA, June, 2012.

- Murad Al Haj, Andrew D. Bagdanov, Jordi Gonzàlez and Xavier Roca. "Reactive Object Tracking with a Single PTZ Camera". In *20th International Conference on Pattern Recognition (ICPR'2010)*, Istanbul, Turkey, August, 2010.

- Murad Al Haj, Andrew D. Bagdanov, Jordi Gonzàlez and Xavier Roca. "Robust and Efficient Multipose Face Detection Using Skin Color Segmentation". In *4th Iberian Conference on Pattern Recognition and Image Analysis (IbPRIA'2009)*, Póvoa do Varzim, Portugal, June, 2009.

- Murad Al Haj, Javier Orozco, Jordi Gonzàlez and Juan José Villanueva. "Automatic Face and Facial Features Initialization for Robust and Accurate Tracking". In *19th International Conference on Pattern Recognition (ICPR'2008)*, Tampa, Florida, USA, December, 2008.

- Murad Al Haj, Ariel Amato, Xavier Roca and Jordi Gonzàlez. "Face Detection in Color Images using Primitive Shape Features". In *5th International Conference on Computer Recognition Systems (CORES'2007)*, Wroclaw, Poland, October, 2007.

- Mikhail Mozerov, Ariel Amato, Murad Al Haj and Jordi Gonzàlez. "A simple Method of Multiple Camera Calibration for the Joint Top View Projection". In *In 5th International Conference on Computer Recognition Systems (CORES'2007)*, Wroclaw, Poland, October, 2007.

- Ariel Amato, Murad Al Haj, Mikhail Mozerov and Jordi Gonzàlez. "Trajectory fusion for Multiple Camera Tracking". In *5th International Conference on Computer Recognition Systems (CORES'2007)*, Wroclaw, Poland, October, 2007.

International Workshops

- Murad Al Haj, Ariel Amato, Gemma Sánchez and Jordi Gonzàlez. "On-line One Stroke Character Recognition Using Directional Features". In *International Workshop on Advances in Pattern Recognition (IWAPR'2007)*, Plymouth, UK, July, 2007.

- Ariel Amato, Murad Al Haj, Josep Lládos and Jordi Gonzàlez. "Computationally Efficient Graph Matching via Energy Vector Extraction". In *International Workshop on Advances in Pattern Recognition (IWAPR'2007)*, Plymouth, UK, July, 2007.

References

[1] Y. Abramson, B. Steux, and H. Ghorayeb. Yef real-time object detection. In *International Workshop on Automatic Learning and Real-Time*, volume 5, page 7, 2005. [Page **17**]

[2] M. Al Haj, A. Amato, X. Roca, and J. Gonzàlez. Face detection in color images using primitive shape features. *Computer Recognition Systems 2*, pages 179–186, 2007. [Page **13**]

[3] M. Al Haj, A.D. Bagdanov, J. Gonzàlez, and F.X. Roca. Robust and efficient multipose face detection using skin color segmentation. In *IbPRIA*, pages 152–159, 2009. [Page **13**]

[4] M. Al Haj, A.D. Bagdanov, J. Gonzàlez, and F.X. Roca. Reactive object tracking with a single PTZ camera. *ICPR*, pages 1690–1693, 2010. [Page **41**]

[5] M. Al Haj, C. Fernández, Z. Xiong, I. Huerta Casado, J. Gonzàlez, and F.X. Roca. Beyond the static camera: Issues and trends in active vision. In Thomas B. Moeslund, Adrian Hilton, Volker Krüger, and Leonid Sigal, editors, *Visual Analysis of Humans*, pages 11–30. Springer, 2011. [Page **41**]

[6] M Al Haj, J. Gonzalez, and L.S. Davis. On partial least squares in head pose estimation: How to simultaneously deal with misalignment. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2602–2609. IEEE, 2012. [Page **55**]

[7] M. Al Haj, J. Orozco, J. Gonzalez, and J.J. Villanueva. Automatic face and facial features initialization for robust and accurate tracking. In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pages 1–4. IEEE, 2008. [Page **13**]

[8] S.A. Al-Shehri. A simple and novel method for skin detection and face locating and tracking. *Lecture Notes in Computer Science, Computer Human Interaction*, 3101:1–8, 2004. [Page **21**]

[9] A. Albiol, L. Torres, and E. J. Delp. Optimum color spaces for skin detection. In *Proceedings of the 2001 International Conference on Image Processing*, volume 1, pages 122–124, 2001. [Pages **19** and **25**]

[10] J. Aloimonos, I. Weiss, and A. Bandyopadhyay. Active vision. *International Journal of Computer Vision*, 1(4):333–356, 1988. [Page **42**]

[11] C.K. Anagnostopoulos, I.E. Anagnostopoulos, I.D. Psoroulas, and E. Kayafas. License plate recognition from still images and video sequences: A survey. *IEEE Transactions on Intelligent Transportation Systems.*, 9(3):377–391, September 2008. [Page **41**]

[12] T. J. Andrews and M. P. Ewbank. Distinct representations for facial identity and changeable aspects of faces in the human temporal lobe. *NeuroImage*, 23:905–913, 2004. [Page **8**]

[13] B. Babenko, P. Dollár, Z. Tu, and S. Belongie. Simultaneous learning and alignment: Multi-instance and multi-pose learning. In *ECCV 2008: Faces in Real-Life Images*, October 2008. [Page **70**]

[14] A.D. Bagdanov, A. Del Bimbo, and W. Nunziati. Improving evidential quality of surveillance imagery through active face tracking. In *ICPR*, pages 1200–1203, 2006. [Page **42**]

[15] S. Baluja, M. Sahami, and H.A. Rowley. Efficient face orientation discrimination. In *Image Processing, 2004. ICIP'04. 2004 International Conference on*, volume 1, pages 589–592, 2004. [Page **17**]

[16] F. Bashir and F. Porikli. Collaborative tracking of objects in EPTZ cameras. In *Visual Communications and Image Processing*, volume 6508, page 2007. Citeseer, 2007. [Page **43**]

[17] N. Bellotto, E. Sommerlade, B. Benfold, C. Bibby, I. Reid, D. Roth, L. Van Gool, C. Fernández, and J. Gonzàlez. A distributed camera system for multi-resolution surveillance. In *International Conference on Distributed Smart Cameras (ICDSC)*, Como, Italy, 2009. [Page **43**]

[18] S.C. Brubaker, J. Wu, J. Sun, M.D. Mullin, and J.M. Rehg. On the design of cascades of boosted ensembles for face detection. Technical Report GIT-GVU-05-28, Georgia Institute of Technology, 2005. [Page **18**]

[19] X. Chen, L. Gu, S.Z. Li, and H.J. Zhang. Learning representative local features for face detection. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, 2001. [Page **17**]

[20] Y. Dai and Y. Nakano. Face-texture model based on SGLD and its application in face detection in a color scene. *Pattern Recognition*, 29(6):1007–1017, 1996. [Page **15**]

[21] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, 2005. [Page **17**]

[22] A. Del Bimbo, F. Dini, G. Lisanti, and F. Pernici. Exploiting distinctive visual landmark maps in pan-tilt-zoom camera networks. In *Computer Vision and Image Understanding (CVIU)*, volume 6,114, pages 611–623, 2010. [Page **43**]

[23] J. Denzler, M. Zobel, and H. Niemann. Information theoretic focal length selection for real-time active 3-d object tracking. In *ICCV*, pages 400–407. IEEE Computer Society Press, 2003. [Page **42**]

[24] R. Dondera and L.S. Davis. Kernel PLS regression for robust monocular pose estimation. In *CVPR 2011 workshop on Machine Learning for Vision-based Motion Analysis (MLVMA'11)*, pages 24–30, 2011. [Page **56**]

[25] U.M. Erdem and S. Sclaroff. Look there! predicting where to look for motion in an active camera network. In *International Conference on Advanced Video and Signal-based Surveillance (AVSS)*, pages 105–110. IEEE, 2006. [Page **43**]

[26] L. Fei-Fei, R. Fergus, and A. Torralba. Recognizing and learning object categories. ICCV short course, September 2009. [Page **16**]

[27] R. Feraund, O.J. Bernier, J.E. Viallet, and M. Collobert. A fast and accurate face detector based on neural networks. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(1):42–53, 2001. [Page **18**]

[28] M. A. Fischler and R. A. Elschlager. The representation and matching of pictorial structures. *IEEE Trans. Comput.*, 22(1):67–92, January 1973. [Pages **8** and **9**]

[29] B. Froba and A. Ernst. Face detection with the modified census transform. In *Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on*, pages 91–96, 2004. [Page **17**]

[30] C. Garcia and M. Delakis. Convolutional face finder: A neural architecture for fast and robust face detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 26(11):1408–1423, 2004. [Page **18**]

[31] P. Geladi and B. Kowalski. Partial least-squares regression: A tutorial. *Analytica Chimica Acta*, 185(1):1–17, 1986. [Page **56**]

[32] R. Gerber and H.-H. Nagel. Representation of occurrences for road vehicle traffic. *Artificial Intelligence*, 172(4-5):351–391, 2008. [Page **44**]

[33] S. Ghouzali, S.S. Hemami, M. Rziza, D. Aboutajdine, and E.M. Mouaddib. A skin detection algorithm based on discrete cosine transform and generalized gaussian density. In *Proceedings of the 2001 International Conference on Image Processing*, pages 605–608. IEEE, 2008. [Page **25**]

[34] R. Gopalan, W.R. Schwartz, R. Chellappa, and A. Srivastava. Face detection. In Thomas B. Moeslund, Adrian Hilton, Volker Krüger, and Leonid Sigal, editors, *Visual Analysis of Humans: Looking at People*, pages 71–90. Springer, 2011. [Pages **16**, **18** and **19**]

[35] 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*, 2004. [Pages **56** and **60**]

[36] N. Gourier, J. Maisonnasse, D. Hall, and J. L. Crowley. Head pose estimation on low resolution images. In *CLEAR Workshop, In Conjunction with Face and Gesture*, April 2006. [Page **61**]

[37] R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-pie. *Image and Vision Computing*, 28(5):807–813, 2010. [Page **65**]

[38] G. Guo and G. Mu. Simultaneous dimensionality reduction and human age estimation via kernel partial least squares regression. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 657 –664, june 2011. [Page **56**]

[39] T. Gustavson and G.E. House. *500 Cameras: 170 Years of Photographic Innovation*. Sterling Signature, 2011. [Page **8**]

[40] A. Hampapur, S. Pankanti, A. Senior, Y.L. Tian, L. Brown, and R. Bolle. Face cataloger: Multi-scale imaging for relating identity to location. In *International Conference on Advanced Video and Signal-based Surveillance (AVSS)*, pages 13–20. IEEE, 2003. [Page **43**]

[41] C. Harris and M. Stephens. A combined corner and edge detector. In *Alvey vision conference*, volume 15, page 50. Manchester, UK, 1988. [Page **16**]

[42] J. V. Haxby, E. A. Hoffman, and M. I. Gobbini. The distributed human neural system for face perception. *Trends Cogn Sci*, 4(6):223–233, 2000. [Page **7**]

[43] B. Heisele, T. Serre, and T. Poggio. A component-based framework for face detection and identification. *International Journal of Computer Vision*, 74(2):167–181, 2007. [Page **18**]

[44] B. Heisele, T. Serre, S. Prentice, and T. Poggio. Hierarchical classification and feature reduction for fast face detection with support vector machines. *Pattern Recognition*, 36(9):2007–2017, 2003. [Page **18**]

[45] K. Hotta. View independent face detection based on combination of local and global kernels. In *International Conference on Computer Vision Systems*, volume 12, page 13, 2007. [Page **18**]

[46] C. Huang, H. Ai, Y. Li, and S. Lao. Learning sparse features in granular space for multi-view face detection. In *Automatic Face and Gesture Recognition (FGR), 2006. 7th International Conference on*, pages 401–406. IEEE, 2006. [Page **17**]

[47] C. Huang, H. Ai, B. Wu, and S. Lao. Boosting nested cascade detector for multi-view face detection. In *Proceedings of the 17th International Conference on Pattern Recognition (ICPR'04)*, volume 2, pages 415–418, 2004. [Page **14**]

[48] D. Huang, M.S., F. De la Torre, and H. Bischof. Supervised local subspace learning for continuous head pose estimation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011. [Page **56**]

[49] G. B. Huang, V. Jain, and E. Learned-Miller. Unsupervised joint alignment of complex images. In *IEEE 11th International Conference on Computer Vision (ICCV)*, 2007. [Page **56**]

[50] A. Ilie, G. Welch, and M. Macenko. A Stochastic Quality Metric for Optimal Control of Active Camera Network Configurations for 3D Computer Vision Tasks. In *International Workshop on Multi-camera and Multi-modal Sensor Fusion Algorithms and Applications (M2SFA2)*, Marseille, France, 2008. [Page **43**]

[51] V. Jain and E. Learned-Miller. Fddb: A benchmark for face detection in unconstrained settings. Technical Report UM-CS-2010-009, University of Massachusetts, Amherst, 2010. [Page **27**]

[52] H. Jin, Q. Liu, H. Lu, and X. Tong. Face detection using improved lbp under bayesian framework. In *Image and Graphics, 2004. Proceedings. Third International Conference on*, pages 306–309. IEEE, 2004. [Page **17**]

[53] M. Jones and P. Viola. Fast multi-view face detection. *Mitsubishi Electric Research Lab TR-20003-96*, 3, 2003. [Pages **14** and **17**]

[54] M. Junered. Face recognition in mobile devices. *Luleåtekniska university*, 2010. [Page **21**]

[55] P. Kakumanu, S. Makrogiannis, and N. Bourbakis. A survey of skin-color modeling and detection methods. *Pattern recognition*, 40(3):1106–1122, 2007. [Pages **19** and **25**]

[56] T. Kanade. *Picture Processing System by Computer Complex and Recognition of Human Faces*. PhD thesis, Kyoto University, Kyoto, Japan, 1973. [Page **8**]

[57] M.D. Kelly. *Visual identification of people by computer*. PhD thesis, Stanford University, Stanford, CA, USA, 1971. [Page **8**]

[58] S. Langton, H. Honeyman, and E. Tessler. The influence of head contour and nose angle on the perception of eye-gaze direction. *Percept. Psychophys.*, 66(5):752–771, 2004. [Page **55**]

[59] A. Lanitis, C.J. Taylor, and T. F. Cootes. An automatic face identification system using flexible appearance models. *Image and Vision Computing*, 13:393–401, 1995. [Page **15**]

[60] T. K. Leung, M. C. Burl, and P. Perona. Finding faces in cluttered scenes using random labeled graph matching. In *ICCV '95: Proceedings of the Fifth International Conference on Computer Vision*, pages 637–644, Washington, DC, USA, 1995. IEEE Computer Society. [Page **15**]

[61] K. Levi and Y. Weiss. Learning object detection from a small number of examples: the importance of good features. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, 2004. [Page 17]

[62] P. Lewi. Pattern recognition, reflections from a chemometric point of view. *Chemometrics and Intelligent Laboratory Systems*, 28:23–33, 1995. [Page 57]

[63] S. Li, L. Zhu, Z.Q. Zhang, A. Blake, H.J. Zhang, and H. Shum. Statistical learning of multi-view face detection. *Proceedings of ECCV*, 2002. [Pages 17 and 18]

[64] Y. Li, S. Gong, and H. Liddell. Support vector regression and classification based multi-view face detection and recognition. In *Proc. IEEE Int. Conf. Autom. Face Gesture Recog.*, pages 300–305, 2000. [Pages 18 and 56]

[65] Zhu Li, Yun Fu, Junsong Yuan, Thomas S. Huang, and Ying Wu. Query driven localized linear discriminant models for head pose estimation. In *ICME'07*, pages 1810–1813, 2007. [Page 61]

[66] R. Lienhart, A. Kuranov, and V. Pisarevsky. Empirical analysis of detection cascades of boosted classifiers for rapid object detection. Technical report, Microprocessor Research Lab, Intel Labs, 2002. [Page 18]

[67] R. Lienhart and J. Maydt. An extended set of haar-like features for rapid object detection. In *Image Processing. 2002. Proceedings. 2002 International Conference on*. IEEE, 2002. [Page 17]

[68] Z. Lin, G. Hua, and L.S. Davis. Multiple Instance Feature for Robust Part-based Object Detection. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. [Page 70]

[69] C. Liu. A bayesian discriminating features method for face detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 25(6):725–740, 2003. [Page 18]

[70] C. Liu and H.Y. Shum. Kullback-leibler boosting. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, 2003. [Page 17]

[71] S.J. McKenna, S. Gong, and Y. Raja. Modelling facial colour and identity with gaussian mixtures. *Pattern Recognition*, 31(12):1883–1892, 1998. [Page 15]

[72] J. Meynet, V. Popovici, and J.P. Thiran. Face detection with boosted gaussian features. *Pattern Recognition*, 40(8):2283–2291, 2007. [Page 17]

[73] K. Mikolajczyk and C. Schmid. Scale & affine invariant interest point detectors. *International journal of computer vision*, 60(1):63–86, 2004. [Page 16]

[74] K. Mikolajczyk, C. Schmid, and A. Zisserman. Human detection based on a probabilistic assembly of robust part detectors. *Computer Vision-ECCV 2004*, pages 69–82, 2004. [Pages 16 and 18]

[75] T. Mita, T. Kaneko, and O. Hori. Joint haar-like features for face detection. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, 2005. [Pages **17** and **18**]

[76] T.B. Moeslund, J.S. Petersen, and L.D. Skalski. Face detection using multiple cues. In *Proceedings of the 15th Scandinavian conference on Image analysis*, SCIA'07, pages 51–60, Berlin, Heidelberg, 2007. Springer-Verlag. [Page **15**]

[77] E. Murphy-Chutorian and M.M. Trivedi. Head pose estimation in computer vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31:607–626, 2009. [Pages **55**, **56** and **61**]

[78] E. Murphy-Chutorian and M.M. Trivedi. Head pose estimation and augmented reality tracking: An integrated system and evaluation for monitoring driver awareness. *IEEE Transactions on Intelligent Transportation Systems*, 11:300–311, 2010. [Page **56**]

[79] E.D. Nelson and J.C. Cockburn. Dual camera zoom control: A study of zoom tracking stability. In *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE Computer Society Press, April 2007. [Page **42**]

[80] J. Ng and S. Gong. Composite support vector machines for detection of faces across views and pose estimation. *Image and Vision Computing*, 20(5–6):359–368, 2002. [Page **55**]

[81] M. Osadchy, Y. Le Cun, M.L. Miller, and P. Perona. Synergistic face detection and pose estimation with energy-based model. In *In Advances in Neural Information Processing Systems (NIPS*, 2005. [Page **18**]

[82] E. Osuna, R. Freund, and F. Girosit. Training support vector machines: an application to face detection. In *Computer Vision and Pattern Recognition, 1997. Proceedings., 1997 IEEE Computer Society Conference on*, 1997. [Page **17**]

[83] C.P. Papageorgiou, M. Oren, and T. Poggio. A general framework for object detection. In *Proceedings of the 6th International Conference on Computer Vision*, volume 2, pages 555–562, January 1998. [Page **14**]

[84] P. Peer. CVL Face Database. http://www.lrv.fri.uni-lj.si/facedb.html. [Page **32**]

[85] P. Peixoto, J. Batista, and H. Araujo. A surveillance system combining peripheral and foveated motion tracking. In *ICPR*, volume 1, pages 574–577. IEEE, 2002. [Page **43**]

[86] T. Poggio and K.K. Sung. Example-based learning for view-based human face detection. In *Proceedings of ARPA Image Understanding Workshop*, volume 2, pages 843–850, 1994. [Page **14**]

[87] F.Z. Qureshi and D. Terzopoulos. Surveillance in virtual reality: System design and multi-camera control. In *CVPR*, pages 1–8, 2007. [Page **43**]

[88] F.Z. Qureshi and D. Terzopoulos. Multi-camera Control through Constraint Satisfaction for Persistent Surveillance. In *International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 211–218. IEEE, 2008. [Page 44]

[89] R. Rae and H. Ritter. Recognition of human head orientation based on artificial neural networks. *IEEE Transactions on Neural Networks*, 9(2):257–265, 1998. [Page 56]

[90] M. Rätsch, S. Romdhani, and T. Vetter. Efficient face detection by a cascaded support vector machine using haar-like features. *Pattern Recognition*, pages 62–70, 2004. [Page 18]

[91] B. Raytchev, I. Yoda, and K. Sakaue. Head pose estimation by nonlinear manifold learning. In *IEEE Conference on Pattern Recognition (ICPR)*, 2004. [Page 56]

[92] S. Romdhani, P. Torr, B. Scholkopf, and A. Blake. Computationally efficient face detection. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, 2001. [Page 18]

[93] R. Rosipal and L.J. Trejo. Kernel partial least squares regression in reproducing kernel hilbert space. *Journal of Machine Learning Research*, 2:97–123, 2001. [Pages 57 and 58]

[94] H.A. Rowley, S. Baluja, and T. Kanade. Nueral network-based face detection. *IEEE Tras. on Pattern Analysis and Machine Intelligence (PAMI)*, 20:22–38, 1998. [Page 14]

[95] P. Sabzmeydani and G. Mori. Detecting pedestrians by learning shapelet features. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, 2007. [Page 17]

[96] H. Schneiderman. Learning a restricted bayesian network for object detection. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, 2004. [Page 18]

[97] H. Schneiderman and T. Kanade. Object detection using the statistics of parts. *International Journal of Computer Vision*, 56(3):151–177, 2004. [Page 18]

[98] Abhishek Sharma and David W. Jacobs. Bypassing synthesis: PLS for face recognition with pose, low-resolution and sketch. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 593–600, june 2011. [Page 56]

[99] P. Sharma and R.B. Reilly. A colour face image database for benchmarking of automatic face detection algorithms. In *Video/Image Processing and Multimedia Communications, 2003. 4th EURASIP Conference focused on*, volume 1, pages 423–428. IEEE, 2003. [Page 21]

[100] M.C. Shin, K.I. Chang, and L.V. Tsap. Does colorspace transformation make any difference on skin detection? In *Proceedings of 6th IEEE Workshop on Applications of Computer Vision (WACV'02)*, pages 275–279, 2002. [Page **19**]

[101] J. Shotton, A. Blake, and R. Cipolla. Contour-based learning for object detection. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, 2005. [Page **17**]

[102] A. Slater and P.C. Quinn. Face recognition in the newborn infant. *Infant and Child Development*, 10:21–24, 2001. [Page **7**]

[103] F. Solina, P. Peers, B. Batagelj, S. Juvan, and J. Kovac. Color-based face detection in '15 seconds of fame' at installation. In *Mirage 2003, Conference on Computer Vision / Computer Graphics Collaboration for Model-Based Imaging, Rendering, Image Analysis and Graphical Special Effects, March 10-11 2003, INRIA Rocquencourt, France, Wilfried Philips, Rocquencourt, INRIA*, pages 38–47, 2003. [Page **32**]

[104] E. Sommerlade and I. Reid. Information-theoretic active scene exploration. In *CVPR*, June 2008. [Page **41**]

[105] S. Stein and G.A. Fink. A new method for combined face detection and identification using interest point descriptors. In *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 519–524, 2011. [Page **16**]

[106] R. Stiefelhagen. Estimating head pose with neural networks - results on the pointing04 icpr workshop evaluation data. In *Proc. Pointing 2004 Workshop: Visual Observation of Deictic Gestures*, 2004. [Page **61**]

[107] R. Stiefelhagen, J. Yang, and A. Waibel. Modeling focus of attention for meeting indexing based on multiple cues. *IEEE Transactions on Neural Networks*, 13(4):928–938, 2002. [Page **56**]

[108] V.B. Subburaman. *Alternative search techniques for face detection using location estimation and binary features*. PhD thesis, ECOLE POLYTECHNIQUE FEDERALE DE LAUSANNE, March 2012. [Pages **13** and **16**]

[109] R. Szeliski. *Computer Vision: Algorithms and Applications*. Springer-Verlag New York, Inc., New York, NY, USA, 1st edition, 2010. [Page **8**]

[110] B.J. Tordoff and D.W. Murray. A method of reactive zoom control from uncertainty in tracking. *Computer Vision and Image Understanding.*, 105(2):131–144, 2007. [Page **42**]

[111] J. Tu, Y. Fu, Y. Hu, and T.S. Huang. Evaluation of head pose estimation for studio data. In *CLEAR Workshop, In Conjunction with Face and Gesture*, pages 281–290, April 2006. [Page **61**]

[112] O. Tuzel, F. Porikli, and P. Meer. Region covariance: A fast descriptor for detection and classification. *Computer Vision–ECCV 2006*, pages 589–600, 2006. [Page 17]

[113] V. Vezhnevets, V. Sazonov, and A. Andreeva. A survey on pixel-based skin color detection techniques. In *Proceedings of International Conference on Computer Graphics (GRAPHICON'03)*, pages 85–92, 2003. [Page 19]

[114] P. Viola and M. Jones. Robust real-time object detection. In *International Journal of Computer Vision*, 2001. [Pages 14, 16 and 18]

[115] P. Viola, M.J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. *International Journal of Computer Vision*, 63(2):153–161, 2005. [Page 17]

[116] 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*, April 2006. [Page 61]

[117] J. Wang, C. Zhang, and H. Shum. Face image resolution versus face recognition performance based on two global methods. In *ACCV*, 2004. [Page 41]

[118] J.G. Wang and E. Sung. EM enhancement of 3d head pose estimated by point at infinity. *Image and Vision Computing*, 25(12):1864–1874, 2007. [Page 56]

[119] P. Wang and Q. Ji. Multi-view face detection under complex scene based on combined svms. In *Proceedings of the 17th Intl. Conf. on Pattern Recognition (ICPR'04)*, volume 4, pages 179–182, 2004. [Page 14]

[120] P. Wang and Q. Ji. Multi-view face detection under complex scene based on combined svms. In *Proc. of ICPR*, volume 12, page 13, 2004. [Page 18]

[121] P. Wang and Q. Ji. Learning discriminant features for multi-view face and eye detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, 2005. [Page 17]

[122] P. Wang, L.C. Tran, and Q. Ji. Improving face recognition by online image alignment. *International Conference on Pattern Recognition (ICPR2006)*, pages 311–314, 2006. [Page 56]

[123] X. Wang, T.X. Han, and S. Yan. An hog-lbp human detector with partial occlusion handling. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 32–39, 2009. [Page 17]

[124] C.A. Waring and X. Liu. Face detection using spectral histograms and svms. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 35(3):467–476, 2005. [Page 17]

[125] G. Welch and G. Bishop. An introduction to the kalman filter. Technical report, University of North Carolina at Chapel Hill, Chapel Hill, NC, USA, 1995. [Page 46]

[126] H. Wold. Soft modeling by latent variables; the nonlinear iterative partial least squares approach. *Perspectives in Probability and Statistics. Papers in Honour of M. S. Bartlett*, 1975. [Pages **56** and **57**]

[127] S. Wrede, M. Hanheide, S. Wachsmuth, and G. Sagerer. Integration and Coordination in a Cognitive Vision System. In *International Conference on Computer Vision Systems (ICVS)*. IEEE Computer Society, 2006. [Page **44**]

[128] B. Wu, H. Ai, C. Huang, and S. Lao. Fast rotation invariant multi-view face detection based on real adaboost. In *Automatic Face and Gesture Recognition, 2004. Proceedings. Sixth IEEE International Conference on*, pages 79–84, 2004. [Page **18**]

[129] B. Wu and R. Nevatia. Detection of multiple, partially occluded humans in a single image by bayesian combination of edgelet part detectors. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, 2005. [Page **17**]

[130] J. Yan. Ensemble svm regression based multi-view face detection system. In *Machine Learning for Signal Processing, 2007 IEEE Workshop on*, pages 163–169, 2007. [Page **18**]

[131] S. Yan, S. Shan, X. Chen, and W. Gao. Locally assembled binary (lab) feature with feature-centric cascade for fast and accurate face detection. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–7. IEEE, 2008. [Page **17**]

[132] G. Yang and T.S. Huang. Human face detection in a complex background. *Pattern Recognition*, 27(1):53–63, 1994. [Page **15**]

[133] J. Yang, D.J. Kriegman, and N. Ahuja. Detecting faces in images: A survey. *IEEE Trasactions on Pattern Analysis and Machine Intelligence (PAMI)*, 24(1):34–58, 2002. [Pages **13** and **14**]

[134] C. Zhang and Z. Zhang. A Survey of Recent Advances in Face detection. Technical report, Microsoft Research, 2010. [Pages **16**, **17** and **18**]

[135] H. Zhang, W. Gao, X. Chen, and D. Zhao. Object detection using spatial histogram features. *Image and Vision Computing*, 24(4):327–341, 2006. [Page **17**]

[136] H. Zhang and D. Zhao. Spatial histogram features for face detection in color images. In *Proceedings of the 5th Pacific Rim conference on Advances in Multimedia Information Processing*, volume Part I of *PCM'04*, pages 377–384, 2004. [Page **24**]

[137] L. Zhang, R. Chu, S. Xiang, S. Liao, and S. Li. Face detection based on multi-block lbp representation. *Advances in Biometrics*, pages 11–18, 2007. [Page **17**]

[138] Z. Zhang, Y. Hu, M. Liu, and T. Huang. Head pose estimation in seminar room using multi view face detectors. In *CLEAR Workshop, In Conjunction with Face and Gesture*, April 2006. [Page **56**]

[139] X. Zhou, R.T. Collins, T. Kanade, and P. Metes. A master-slave system to acquire biometric imagery of humans at distance. In *International Workshop on Video Surveillance (VS)*, pages 113–120, 2003. [Page **43**]

[140] Z.-H. Zhou and M.-L. Zhang. Multi-instance multilabel learning with application to scene classification. In *In Advances in Neural Information Processing Systems (NIPS)*, 2007. [Page **70**]