Engineering, Statistical Modeling and Performance Characterization of a Real-Time Dual Camera Surveillance System

Der Technischen Fakultät der
Universität Erlangen-Nürnberg
zur Erlangung des Grades

DOKTOR-INGENIEUR

vorgelegt von

MICHAEL GREIFFENHAGEN

Erlangen – 2001

Als Dissertation genehmigt von
der Technischen Fakultät der
Universität Erlangen-Nürnberg

Tag der Einreichung:

Tag der Promotion:

Dekan:

Berichterstatter:

Abstrakt

Rascher Fortschritt in Rechenleistung, Verfügbarkeit günstiger Sensoren und flexibler Algorithmen fördern die Entwicklung von Echtzeit-Videoüberwachungs-Systemen. In gewissen Anwendungsbereichen kann die Nutzung von intelligenten Videosystemen nur erfolgen, wenn erforderliche Qualitätsmerkmale in Hinblick auf die System-Leistung garaniert werden können. Die vorliegende Arbeit beschäftigt sich mit der Frage, wie ein solches System designt werden sollte, das vom Anwender vorgegebene Anforderungen erfüllt. Es wird gezeigt, dass es unter Zuhilfenahme statistischer Methoden möglich ist, Kontroll-Parameter des Systems automatisch zu bestimmen/optimiern und quantitativ den Einsatzbereich des Systems zu bestimmen. Voraussetzung ist die vernünftiger Wahl der System-Module und eine gezielter Untersuchung, wie die verschidenen Parameter das Systemverhalten beeinflussen.

Die vorliegende Arbeit konzentriert sich auf das Entwerfen und Bilden eines Zwei-Kamera Systems, das Personen im Raum detektiert und ein gezoomtes Bild ihrer Köpfe liefert und vordefinerte anwendungsspezifische Anforderungen erfüllt. Ziel des Systems ist einerseits die kontinuierliche Bereitstellung eines Überblick-Bildes der Szene und andererseits die gleichzeitige Bereitstellung eines hochauflösenden herangezoomten Bildes vom Kopf einer Person, die sich irgendwo in dem zu überwachenden Bereich befindet. Hierzu wird ein omni-direktionals Video verarbeitet. Nach Lokalisierung der Person im omni-direktionalen Bild erfolgt eine Koordinaten-Transformation mithilfe der der Senk- und Neigewinkel sowie der Zoom einer aktive Kamera präzise kontrolliert werden. Wir werden feststellen, dass sowohl die Schätzwerte als auch die zugehörige Datenunsicherheit einen Fuktion der zugrunde liegenden Geometrie, Lichtbedingungen, des Hintergrund-Kontrastes, der relativen Position zwischen der Person und beiden Kameras sowie der Kalibierungs-Fehler und des Sensor-Rauschens sind. Die Unsischheit in den Schatzwerten wird benutzt, um den Zoom-Parameter adaptiv einzustellen, so dass mit einer vom Anwender vorgegebenen Wahrschinlichkeit α_Z der komplette Kopf einer Person im Bild der aktiven Kamera abgebildet wird. Je grösser die Wahrschinlichkeit α_Z gewählt wird, desto weniger weit zoomt das System. In unserem System haben wir α_Z auf 95% gesetzt.

Im zweten Teil der Arbeit wird erläutert, wie mit nur minimalem Aufwand an Re-Design und Analyse, der Einsatzbereich das existiernede System erweitert werden kann, wenn bereits in der Design-Phase systemeatischen Entwicklungs-Prinzipien gefolgt wurde. Die Schlussfolgerung wird sein, dass wenn geeignete Module und statistische Representationen gewählt werden, es möglich ist, das bereits existierende System-Design und Analyse-Ergebnisse zu übernehemen. Während das original System für Innenanwendungen mit statischer Beleuchtung konzipiert wurde, wird das endgültige System dahingehend erweitert, dass es auch in Bereichen eingesetzt werden kann, die unter sich dynamisch ändernden natürlichen Beleuchtungseinflüssen stehen. Es wird gezeigt, dass nach Erweiterung fast sämtliche Module und fast die gesammte Leistungsanalyse des alten Systems übernommen werden können. Das System ist im Eingagnsbereich eines Bürogebäudes tags und nachts zuverlässig im Einsatz.

Schlüsselworte: System Entwickklung, Statistische Modellierung, Fehler Analyse, Leistungs Charakterisierung, Echtzeit, Video Überwachung.

Curriculum Vitae

Michael Greiffenhagen was born on December 10, 1970. He received the Dipl. Eng. degree in electrical engineering from the University of Technology, Hamburg-Harburg, Germany, in 1997. Since 1997, he has been with Siemens Corporate Research (SCR), Princeton, NJ, USA, where he has been pursuing his Ph.D. under the supervision of Dr. Visvanathan Ramesh. His academic supervisor is Prof. Dr. Heinrich Niemann from the University of Erlangen-Nürnberg, Germany. His research interest is in the field of real-time video processing, and systems performance analysis. While working on his Ph.D. thesis, he is investigating in how to build robust and reliable vision-systems for surveillance and monitoring. He is focusing on building real-time and adaptive systems by using statistical methods for systems engineering and analysis. Refereed book, journal and conference articles are listed in the following, as well as invited talks.

Publications

Refereed Book Publications

1. **Michael Greiffenhagen**, Visvanathan Ramesh. Performance Analysis of Multi-Sensor Based Real-Time People Detection and Tracking System. *Multimedia Video-Based Surveillance Systems: Requirements, Issues and Solutions.* Foresti, Mahonen, Regazzoni Editors. Kluwer Academic Press, New York, 2000.

Refereed Journal Publications

1. **Michael Greiffenhagen**, Dorin Comaniciu, Heinrich Niemann, and Visvanathan Ramesh. Design, Analysis and Engineering of Video Monitoring Systems: An Approach and a Case Study. *Proceedings of the IEEE, Special Issue on Video Surveillance 2001.*

Refereed Conference Publications

1. **Michael Greiffenhagen**, Visvanathan Ramesh, Heinrich Niemann. The Systematic Design and Analysis Cycle of a Vision System: A Case Study in Video Surveillance. Submitted for *IEEE Conference on Computer Vision and Pattern Recognition*, 2001, under review.

2. **Michael Greiffenhagen**, Visvanathan Ramesh, Dorin Comaniciu, Heinrich Niemann. Statistical Modeling and Performance Characterization of a Real-Time Dual

Camera Surveillance System *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2000).* IEEE Computer Society (publisher); Volume 2, pp. 335-342. Hilton Head Island, South Carolina, USA; June 13-15, 2000.

3. Visvanathan Ramesh, **Michael Greiffenhagen**, Marie-Pierre Jolly. Performance Characterization of Image & Video Analysis Systems at Siemens Corporate Research. *Proceedings of the Special Workshop on Performance Evaluation in Medical Imaging* (in conjunction with SPIE Medical Imaging 2000). California, USA, Feb. 2000.

4. Visvanathan Ramesh, **Michael Greiffenhagen**, Serge Bouverie, Alain Giralt. Real-Time Video Surveillance and Monitoring for Automotive Applications. *Proceedings of the Society of Automotive Engineers 2000 World Congress.* Detroit, USA, 2000.

5. Yuntao Cui, Supun Samarasekera, Qian Huang, **Michael Greiffenhagen**. Indoor Monitoring via the Collaboration between a Peripheral Sensor and a Foveal Camera. *Proceedings of the IEEE Workshop on Visual Surveillance*, Bombay, India, 1998.

6. Qian Huang, Yuntao Cui, Supun Samarasekera, **Michael Greiffenhagen**. Auto Cameraman Via Collaborative Sensing Agents *Proceedings of the Third Asian Conference on Computer Vision*, Hong Kong, 1998.

Invited Talks

1. Performance Characterization of a People Detection and Tracking system *Accenture, Center of Strategic Research (CStR)*, Sophia Antipolis, France, January 2001.

2. Real-Time Video Analysis at Siemens Corporate Research: Systems Research and Statistical Performance Characterization" *Special Session on Advanced Video-Based Surveillance Systems, International Conference on Image Analysis and Processing, ICIAP*, Venice, September 1999.

Abstract

Rapid improvement in computing power, cheap sensing and more flexible algorithms are facilitating increased development of real-time video surveillance and monitoring systems. The deployment of video understanding systems in certain critical applications in the real world can be done only if performance guarantees can be provided for these systems. This work emphasizes on how to systematically design such a system, which matches user defined requirements. It will be illustrated that by judiciously choosing the system modules and by performing a careful analysis of the influence of various tuning parameters on the system it is possible to perform proper statistical inference, to automatically set control parameters and to quantify performance limits.

This work focuses on engineering a dual-camera real-time people detection and zooming system that meets given application requirements. The goal of the system is to continuously provide an image of the entire scene as well as a high resolution zoomed-in image of a person's head at any location of the monitored area. An omni-directional camera video is processed to detect people and to precisely control a high-resolution foveal camera, which has pan, tilt and zoom capabilities. The pan and tilt parameters of the foveal camera and its uncertainties are shown to be functions of the underlying geometry, lighting conditions, background color/contrast, relative position of the person with respect to both cameras as well as of sensor noise and calibration errors. The uncertainty in the estimates is used to adaptively estimate the zoom parameter that guarantees with a user specified probability, α_Z, that the detected person's face/head is contained and zoomed within the image. The higher the probability α_Z the more conservative the zoom factor would be. We set α_Z to 0.95 in our current system.

In the second part it will be shown how the existing system designed and analyzed by following rigorous systematic engineering principles can be extended to relax the system operating conditions with minimal re-design and analysis efforts. The key conclusion is that by choosing appropriate modules and suitable statistical representations, we are able to re-use existing system design and performance analysis results. While the original system was designed for indoor (static illumination) settings the final system is extended to deal with dynamic illumination changes in a quasi outdoor setting. It is shown that extensive re-use of the original system and its performance characterization results can be achieved. The system operates reliably during days and night conditions in an office building lobby.

Keywords: System Engineering, Statistical Modeling, Error Analysis, Performance Characterization, Real-Time, Video Surveillance, Monitoring.

To my parents Bärbel and Wilhelm.

Acknowledgements

My sincere thanks go to my academic advisor, Professor Heinrich Niemann. He gave me the opportunity to simultaneously gain experience in academia and industry abroad, and learn how to combine the best of both worlds. The insights learned will be of immeasurable value for my future professional career. Even though the North-Atlantic resided between his department for pattern recognition at University of Erlangen in Germany and Siemens Corporate Research in Princeton, New Jersey, USA, the lab where I conducted my research studies at, we maintained close contact through email and had many valuable and fruitful discussions in Erlangen. I gratefully appreciate his flexibility and helpfulness in arranging meetings on a short notice and giving immediate feedback despite his tight schedule on one side and limited travel options on my side.

My deepest thanks go to my local advisor, project manager and friend Dr. Visvanathan Ramesh at Siemens Corporate Research in Princeton, NJ, USA. He continuously challenged me, questioned every little detail to make me strive for excellence and go beyond what i thought my limits were. He greatly influenced my way of thinking and taught me how to face, attack and overcome academic problems. Whatever dead-end I experienced along the way, he wanted me to learn a lesson and never viewed it as a failure. His capability to fuse academic brilliance with the industrial need for precision and reliability was a great example to me of how to solve real world problems.

I am also very grateful to Dr. Dietrich Paulus. His organizational talent and help made it possible to smoothly operate between Germany and the USA. He provided any support and pointers one could wish of.

I also like to thank Professor Robert M. Haralick for his time and effort he spent to provide valuable comments and discussion to improve the work. Despite his tight schedule he offered to serve on the committee and provided a second opinion.

Many thanks also to Dr. Alok Gupta, head of the Imaging and Visualization Department at Siemens Corporate Research (SCR). By sponsoring this work during the last 3.5 years, he gave me the opportunity to gain industrial work experience while pursuing my Ph.D. dissertation. He constantly provided great support since I joined the company to write my Master thesis under supervision of Dr. Qian Huang from SCR and Professor Otto Lange from University of Technology, Hamburg-Harburg. I also like to thank them for encouraging me and suggesting the pursuit of a Ph.D. dissertation.

My colleagues Dr. Dorin Comaniciu and Dr. Nikos Paragios from the ”Real time video group” at SCR I like to thank for valuable discussions and support. It was great fun working in this group.

Many thanks also to my office mate and friend Markus Kukuk. Together we went through lows and highs during our dissertation. We had great fun and valuable discussions beyond academic bounds.

Thanks to all colleagues in the imaging and visualization department at SCR. They provided a wonderful and encouraging working atmosphere. Special thanks to my colleagues Drs. Marie-Pierre Dubuisson-Jolly, James Williams, Arun Krishnan, Klaus Neubauer, and Benedict Bascle for the fun time we spend on get-togethers, and conferences. A special thank to our department secretary Angelika Formento, who made the department run smoothly and helped us focus on our research work.

Many thanks to Steven F. Owens, head of Siemens Medical Software Company West, Iselin, NJ, USA and Rainer Schultz. If I had not enjoyed my first student-internship abroad so much after my fellow-student Markus Breyer provided the initial contact address, my future might have been totally different. Shortly before, I first got in contact with image processing, when I was student in the lab of Professor Hans Burkhardt from University of Technology in Hamburg-Harburg (he is currently with University of Freiburg), Germany, and worked on a project together with Dr. Sven Utcke (currently post-doc at University of Hamburg). At this point I like to thank them all for the impact they made on my career.

My warmest thanks go to my family, my mother Bärbel, my father Wilhelm and my brother Lars. By having provided a wonderful childhood, by always being open-minded, by combining care and yet providing enough freedom to grow independent they set the base for everything. Every success was only possible through their help, support and continuous love.

I like to extend my warmest thanks to my friend and Uncle, Professor Claus Unger. He dramatically influenced my ideals, goals and way of thinking. He continuously offers generous help and advice in personal and career matters no matter how busy he currently is. Only his encouragement made so many things possible to me.

Finally, I like to express my greatest thanks to my fianceé and future wife, Katja Sakiewicz. Thank you so much for your endless love and support, for encouragement on our USA-adventure, and for making this experience fun and successful.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The combination of drop in price of visual sensors along with increasing availability of cheap computational power is making feasible real-time systems for video processing [65]. In the commercial sector, there is a growing need for video surveillance monitoring, for example to improve public safety and security. There is an increased use of video surveillance systems in urban areas, public transportation systems, etc. This growth is accelerated by facts that the sensors are getting advanced and cheaper (e.g. novel sensing methods such as the omni-directional video camera, omni-directional stereo sensor, real-time stereo sensors, are now products) and processing is getting cheaper. Visual surveillance and monitoring (VSAM) systems are increasingly becoming strongest factors in prevention and reduction of crime, and in the improvement of effective management of resources (e.g. traffic management, subway monitoring). The engineering of such systems to meet application specific computational and accuracy requirements is crucial to the rapid deployment of these systems.

This thesis focuses on three main aspects:

- It illustrates the use of systematic engineering methodology outlined in [97] to design and validate a real-time system with given computational and accuracy constraints.

- It shows that by judicious choice of the intermediate transforms (components of the system) along with a careful analysis of the influence of various parameters in the system, it is possible to perform proper statistical inference, to automatically set the control parameters and to quantify and predict the limits of a real-time dual-camera video surveillance system.

- It investigates how it is possible to extend the system by taking it from constraint indoor settings to quasi-outdoor.

1.1 Problem Statement

This section briefly states the problems being addressed in this thesis. On the abstract level the problem of translating requirements to system design is investigated before the insights gained are then applied to designing a real-time dual-camera people detection and zooming system. Finally, we address the challenge of how to relax constraints and add requirements to a given system without changing the architecture, existing modules and analysis.

1.1.1 Translating Requirements to System Design:

The typical scenario in an industrial research and development unit developing vision systems is that a customer defines a system specification and its requirements. The engineer then translates these requirements to a system design and validates that the system design meets the user-specified requirements. The system requirements in the video analysis setting often involves the specification of the operating conditions, the types of sensors, the accuracy requirements, and the computational requirements to be met by the system. The operating conditions essentially restrict the space of possible inputs by restricting the type of scene geometry, the physical properties such as object material types, and illumination conditions, and object dynamics. The accuracy requirements are usually defined in terms of detection and false alarm rates for objects, while the computational requirement is specified typically by the system response time to an object's presence (e.g. real-time or delayed?). The objective of the vision systems engineer is to then exploit these restrictions (i.e. constraints) and design a system that is optimal in the sense that it meets customer requirements in terms of speed, accuracy and cost.

The main problem, however, is that there is no known systematic way for vision systems engineers to go about doing this translation of the system requirements to a detailed design. It is still an art to engineer systems that meet given application specific requirements. There are two fundamental steps in the design process: The choice of the system architecture and modules for accomplishing the task, and the statistical analysis and validation of the system to check if it meets user requirements. In real-life, the system design and analysis phases typically follow each other in a cycle until the engineer obtains a design and an appropriate analysis that meets the user specifications. Figure 1.1 illustrates the design and analysis process.

Automation of the design process is a research area with many open issues, although there have been some studies in the context of image analysis (e.g. automatic programming); please see: [39], [118], [64]. The systems analysis (performance characterization) phase in the context of video processing systems is an active area of research in the last few years. Performance evaluation of image and video analysis components or systems is

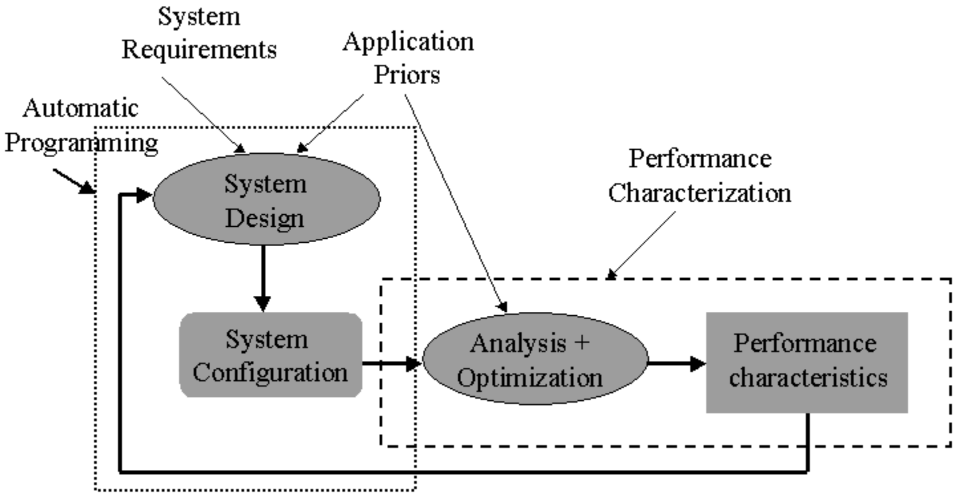

Figure 1.1: System Design and Analysis Phases for Vision Systems

an active research topic in the vision community ([17], [35], [47], [53], [54], [97], [102], [134]). In chapter 2 the literature in performance characterization and other related work in computer vision is reviewed.

1.1.2 Dual-Camera Surveillance System Design & Analysis:

Given that background, this work focuses on how to engineer an actual visual surveillance system that

- constantly monitors wide regions of interest (ROI)
- automatically detects people in the ROI
- simultaneously provides high resolution images of a person's head if within the ROI
- meets precision requirements
- provides quantitative performance measures at any given time
- provides a framework that is extendable and promotes re-use of modules

The use of the methodology above will be illustrated in the context of video surveillance. The analysis (Greiffenhagen et al [40, 41]) involves statistical modelling and performance characterization of a real-time dual-camera surveillance system. The design of the system has to be chosen such that application specific priors in the 3D geometry, camera geometry, the illumination model parameters, and object interaction/motion parameters are taken into account while designing the object detection and zooming system. It will be described how it is possible to meet the real-time constraints while satisfying accuracy constraints (under certain restrictive assumptions).

1.1.3 Relaxing Constraints, System Extension:

In general, systems are designed for a certain scenario. Nevertheless, often the same system is to be taken to perform in a different environment with requirements that do not correspond to the original system requirements and constraints at the time the system was designed. One part of this work deals with the question how to adapt an existing system, that was design by following the methodology mentioned above, to new, relaxed constraints and added requirements without changing the architecture, existing modules or analysis.

1.2 Contribution of the Thesis

There are two main contributions of this thesis:

- One contribution is the demonstration of a systematic design methodology for building a complete real-time video surveillance system.

- The other contribution deals with the adaptation of the existing system to show how one can incrementally evolve the current system design to meet added requirements.

First, this thesis demonstrates how one can design a complete vision system that takes application specific priors into account and propagates them through each transform of the system. It will be shown that tuning constants can be derived automatically and adaptively, given certain performance constraints such as real-time operation, adaptive zooming, and the need to adapt automatically to different settings. It will be demonstrated how quantitative performance measures can be defined in respect to the system's task and application requirements to boost performance and predict consistent and reliable system behavior. While [97] does propagate probability density functions (*pdf*s) and uncertainties through the entire chain of transforms and tunes system parameters in respect to the final task, it does not address real-time issues and the choice of the architecture itself. In this thesis, a complete real-time dual-camera people detection and zooming system is built and analyzed. The system tracks a person in an omnidirectional image and controls pan tilt and zoom of a second (active) camera. By following the proposed design methodology we were able to design the system such that the control parameters are automatically set based on the underlying geometry, current lighting conditions, current background/contrast, relative position of the person with respect to both cameras and sensor noise and calibration errors. The analysis conducted helps to obtain the performance limits of the system.

Secondly, a new approach based on the framework and module design chosen is proposed to extend the system such that new, user-defined requirements can be added, and operating constraints be relaxed while maintaining the previously conducted statistical analysis valid and untouched. For that, re-use of existing modules is proposed as well as incorporation of augmented third party modules. This allows fast improvements of a given system without redesigning modules from scratch and adapting them to new feature spaces. The theoretical results are demonstrated on the illumination module. It is demonstrated how to migrate a system originally designed for indoor and static background to a system, which runs stable and predictable under varying light conditions (influenced by artificial and natural light sources) and changing background. It is shown how to augment a third party module in order to meet the application requirements and how to statistically correctly fuse the existing with the augmented module. The final system combines advantages of both modules and operates during day and night conditions in an office building entrance lobby, which is lit by daylight and artificial light during day and lit by artificial light only during the night. For each system module, model assumptions are validated. Finally, the system modules are validated to ensure that the modules model the real world correctly.

1.3 Organisationtion of the Thesis

Chapter 2 reviews work on performance characterization and system analysis methodology. In the second section, we review work on visual surveillance and monitoring systems. Since in our application precise segmentation greatly influences the performance, the review also addresses illumination invariant background adaptation.

Chapter 3 provides background information and motivation for why the proposed approach is chosen. Since this work draws on ideas outlined in [97] a detailed review of this work is provided. The section also motivates the choice to use a catadioptric imaging system as proposed in [84] and reviews alternative approaches to wide field of view image processing systems.

Chapter 4 explains how the application specific priors and requirements influence the choice of the system architecture. It illustrates how perturbations can be propagated through the chosen surveillance system configuration involving change detection, people detection, people location determination and camera parameter estimation, and points out how this approach differs from the work outlined in [97]. The first part involves the design issues (choice of the system configuration given application requirements and priors), and emphasizes on statistical modelling, uncertainty propagation and on how to consider prior information. The second part describes in detail the chosen algorithm and explains the transforms module-wise. The third section involves the systems analysis of the system configuration chosen. The final section provides an experimental validation of the theoretical results in the analysis. Finally, it demonstrates the system's performance under a variety of settings. The system is therefore installed at different locations to show automatic adaptation and reliability.

Chapter 5 describes how for the given system user-defined requirements can be added to the existing system and how constraints can be relaxed while re-using existing modules, and maintaining the previously conducted statistical analysis valid and untouched. It demonstrates yet another loop in the design phase of a vision system. The approach is illustrated on the illumination module, which was primarily designed for constant background in indoor environments, and is extended to cope with changing background influenced by natural light. The chapter emphasizes on adding and augmenting a third party module to the existing system, such that the system operates stable during day and night conditions. Under the new conditions, long-term experiments similar to the ones with the original system configuration are conducted. It is shown that the experimental results match the theoretically expected results. Finally it is demonstrated that the analysis conducted for the old system remains valid.

Chapter 6 presents results and insights gained using the proposed methodology for system engineering and design in respect to the application of our real-time dual-camera people detection and zooming system.

Chapters 7 summarizes this thesis; chapter 8 closes with an outlook on future work.

Chapter 2

Literature Review

This work focuses on both, the theoretical as well as the design aspects of an actual video surveillance and monitoring (VSAM) system. Therefore, this chapter reviews literature on both, A) on system engineering methodology, which aims to build robust and predictable vision systems that work under a variety of predefined conditions, and B) on existing VSAM systems. The second part particularly focuses on available segmentation and background adaptation techniques, since precise segmentation is crucial in our application.

2.1 System Methodology

This section reviews past work on performance characterization and system analysis methodology.

2.1.1 Performance Characterization Literature Review

The need for rigorous performance evaluation of vision algorithms have been stressed in the 80's and in the early 90's (for example: Haralick [53], [45], Jain and Binford [63], Petkovic [91], Price [93]). Early work in performance characterization was aimed at the identification of the methodology for characterizing limits of vision systems. For example, Haralick [54] outlines the necessity for a well planned experimental protocol to evaluate the performance of vision systems and provides details of the recipe for constructing a typical experimental plan. Ramesh et al [97] summarize a systems engineering methodology for building vision systems and illustrate performance characterization of a system for building parameter estimation. In this methodology there are two main steps: statistical modeling or performance characterization of component algorithms (component identification) and application domain characterization. Component identification (see figure 2.1) involves the derivation of the deterministic and stochastic behavior of each module. This

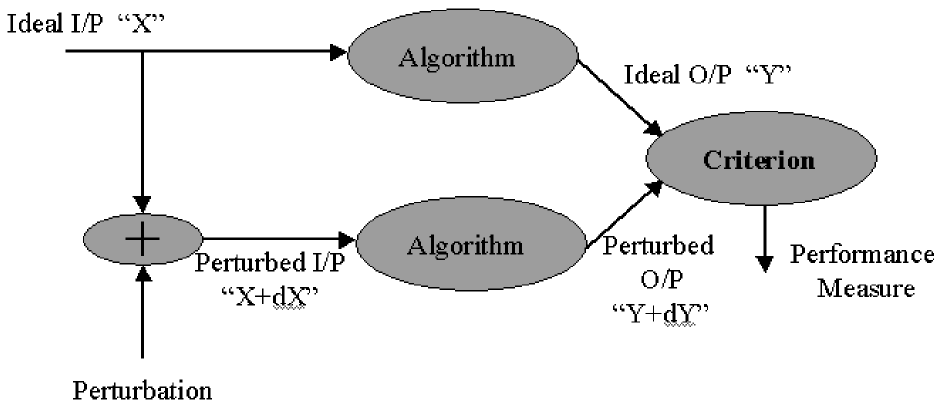

Figure 2.1: Component Identification or Performance Characterization

entails the specification of the ideal model and an error model in the input and relating their parameters to the output ideal model and error model parameters. The essence of the methodology is that each sub-step used in a vision system is treated as an estimator and therefore the estimator's behavior has to be characterized in terms of its distribution. The distribution of each estimator is a function of the input samples and error distribution parameters. When a system is composed of multiple estimation steps concatenated together then performance characterization is a daunting task. Some of the issues related to this step will be described in the next section. Application domain characterization (see Figure 2.2) is a learning or estimation step wherein the restrictions on the application data relevant to the task at hand are specified in terms of prior distributions of parameters relevant to the algorithm/system representation chosen. These prior distributions can be viewed as specifying the range of possible images for the given application. The average or worst case performance of the system can be determined by combining the Component Identification steps and the Application domain modelling steps. In [97] optimal control parameter settings are chosen to select the parameters that provide the best expected performance over the distribution input images. A detailed review of the methodology will be given in section 3.1.2. The use of the methodology for the design and analysis of a dual-camera monitoring system will be illustrated in chapter 4.

A special issue on performance evaluation contains a number of related references on vision algorithm evaluation (Please see: Forstner [35]). More recently there have been several workshops dedicated to empirical evaluation of vision algorithms (for instance, see [17], [102] and [134]). Most of the papers in the empirical evaluation workshops aim at addressing black box evaluation methodologies for vision algorithms. A black box approach essentially involves empirical evaluation without knowledge of the system transform functions. This is in contrast to Ramesh and Haralick [94], [95] and Ramesh et al [97] that address white box evaluation. The above methodology essentially can be seen as a white

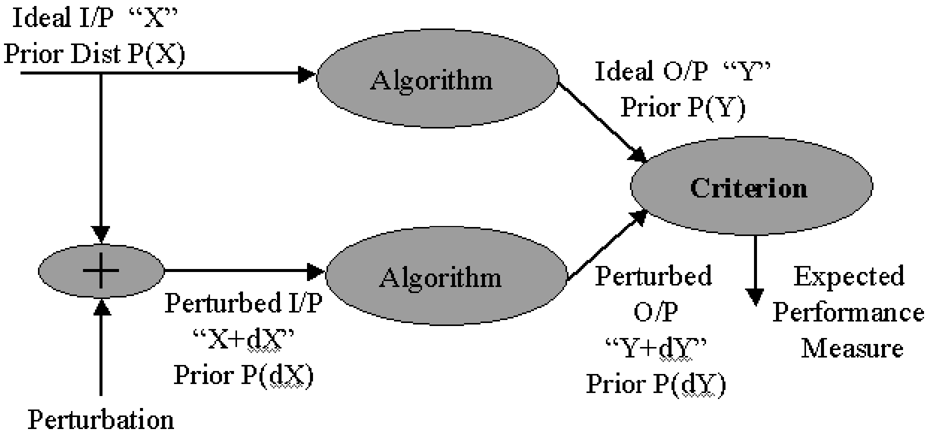

Figure 2.2: Application Domain Characterization

box analysis of a given system. Courtney et al [23] describe an empirical approach to systems evaluation. Cho and Meer [21] were the first to use re-sampling techniques (e.g. Bootstrap) as a tool for studying the performance of edge detection techniques. Papers have also appeared in the literature on boundary-segmentation performance evaluation. Some of the early papers include Ramesh and Haralick [94], [96], [98], Heath et al [58], and Wang and Binford [121]. The first set of papers evaluates edge-parameter estimation errors in terms of the probability of false alarm and miss-detection as a function of the gradient threshold. In addition, the edge location uncertainty and the orientation estimate distribution is derived to illustrate that at low signal-to-noise ratios the orientation estimate has large uncertainty. The paper by Heath et al visually compares the outputs from various edge detectors. More recently, Shin et al [104] studies object recognition system performance as a function of the edge operator chosen at the first step of the recognition system. The conclusion is that the Canny edge detector is superior to the others compared. Most of the papers described above use simulations or hand drawn groundtruth to compare algorithm results with groundtruth results. Baker and Nayar's [7] work is unique in that it does not require groundtruth. The evaluation is made by examining statistics that measure global coherence of edge points detected (e.g. collinearity etc.). Konishi et al [68] address edge detector evaluation by using information theoretic principals. Papers have also been published on performance evaluation in the context of document analysis systems, other feature extraction methods (e.g. corner extraction [97]), etc. Early papers on sensitivity analysis in the context of geometric hashing and Hough transform for object recognition should also be mentioned in this context (Please see Grimson and Huttenlocher for example [44],[43]). They study the false alarm characteristics of the recognition technique when a spatially random clutter model is assumed with a given density and a bounded error model is assumed for the object feature points that are detected. The analysis provides the mechanism to automatically set up the recognition threshold so that a given false alarm rate can be met by the system. Recent work on performance evaluation in the context of object recognition was done by Boshra and Bhanu [12].

2.1.2 Systems Analysis Methodology Review

The systems analysis methodology is described in [97]. The methodology essentially addresses the problem of analyzing and setting up tuning constants for a vision system with a chosen architecture. However, the methodology proposed does not address computational issues and the choice of the architecture itself, which is part of this work. Since this work draws on core ideas presented in [97] a detailed discussion is presented in ”Background and Motivation”, section 3.1.2.

2.2 Video Surveillance and Monitoring

An excellent review of surveillance systems is provided by Boult et al. [16]. In the following, we summarize some of the references. Also, the August 2000 special issue on video surveillance of the IEEE Transactions on ”Pattern Analysis and Machine Intelligence” provides a good overview on many state of the art systems.

Tracking techniques based on features, edges or boundaries are presented in e.g. [60], [127], [110], [67], [28]. However, for our application, the variable and at places small object-size limit the applicability of feature-based approaches.

Optic flow is another class of techniques used. E.g. [125] uses correlation or sum-of-squared-differences (SSD) over windows. These will not work well with small objects, large amounts of occlusion or non-rigid objects. Others use feature-based optic flow, e.g. [110] computes and tracks features over time.

Using features to initialize a stronger model is a tracking technique which has been used frequently, e.g. with strong models for vehicles in [67], [110] and weak models for people in [100], [128], [92], [56], [27]. Models restrict the search area for likely features while increasing sensitivity without significantly increasing the chance of false alarms. Nevertheless, these systems require a large number of object pixels as well as model initialization. Required initialization limits trackers that use deformable models, e.g. [101], [60], [127], where the initialization is required to be quite close to the objects outline. Furthermore, often deformable models are too expensive for real-time tracking, e.g. [127] or [101] which could not deal with changing illumination. For some domains color is used to simplify the initialization (and even model tracking), e.g. in [123] and in numerous face tracking systems e.g. [22], skin color is critical to both detection and tracking, .

A large number of papers investigates into tracking and analyzing human motion, e.g.

[100], [128], [92], [56], [27]' [57]. For example, [73] uses motion parameters as the primary method to distinguish between human and vehicle. However, it is presumed that objects consist of many pixels in the range of hundreds or thousands), and are not occluded (. A system that uses both motion parameters and target size/shape information to classify targets as human, bird, rabbit, fox or squirrel is presented in [100]. E.g. [11], [6], [26], and [99] worked on developing target motion estimators. Color and intensity histograms combined with motion parameters are used for tracking in [61],and [22]. However, for the general case, initialization is an open issue. Furthermore, both algorithms require a sufficiently large number of target pixels.

In some application domains, a wide field of view is required. This is usually accomplished by using either multiple passive cameras or a single active pan-tilt-zoom-camera. Nayar [83] proposed an omni-directional camera which combines a standard standard camera with a parabolic mirror to capture a full viewing hemisphere. As the result, the system generates an image that sees in all directions, with some apparent distortions. Since the system proposed in this work is built using this kind of sensor – Shree Nayar's omni-camera – a detailed discussion is given in the chapter on background and motivation (see chapter 2.2.1).

Although there has been much work on frame-to-frame matching, feature-based techniques, and motion estimation, most systems focused on aspects other than change detection. Therefore, segmentation techniques used, may work well for indoor, but are not likely to be sensitive and robust enough to handle objects of varying size in areas illuminated by sun light during day and artificial light during the night. For our application, the detection phase is crucial; undetected people can not be zoomed on.

2.2.1 Change Detection Review

One of the most common types of change detection algorithms is based on subtracting a background model (or models) from the current image followed by thresholding. These techniques involve for each pixel the modelling of an expected value. In the following, such techniques are discussed.

Many background-modelling approaches assume a single Gaussian to model a pixel value. Since lighting can change and over time, different objects may project onto the same pixel, more recent systems assume multiple (usually 2 to 5) models, e.g. a Mixture of Gaussians (MOG), per pixel, e.g. [101], [111]. For computational reasons, the covariance matrix is often assumed diagonal (i.e. uncorrelated). Obviously, the special case K = 1 is the traditional Gaussian model. Please note that a MOG can also approximate a single pixel's unimodal intensity distribution if this can not be modelled well by a single Gaussian.

To use a MOG model, one needs to assume that the underlying data satisfy a quasi-stationary criterion: A change in the pixel intensity value is slow compared to the update rate of the model. For dynamic MOG models, a high-level labelling process is presumed to correctly indicate which part of the mixture is to be updated. In the following, previous work on background modelling and the standard approaches are reviewed.

A multi-class statistical model for the tracked objects is used by the P-finder system [128] uses, but it the background a single Gaussian per pixel. A single Gaussian per pixel, used in many systems, is easy to estimate. Thresholding based on the standard deviation is statistically well justified, if the model is appropriate. Some systems simply track the mean or some other models of central tendency and use an ad-hoc thresholding process while ignoring the formal modelling of standard deviation.

Other systems support multi-background models per pixel to improve robustness, especially with outdoor scenes containing significant clutter, see [14], [101], [111], [56]. [101], and [111] fit a MOG to the given input samples. The parametric from of the MOG distributions can then be used to classify pixels. [14] uses a simpler form that tracks only the central values of the two primary distributions for a pixel. These papers draw mostly on intuition and insight, and do not present experimental justification for their multi-background model assumption and parameter settings.

Maintaining or updating the background model can be achieved through multi-sample or per-frame processing. E.g. [32], and [101] gather many samples per pixel to compute statistical models such as Gaussian, MOG, or non-parametric respectively. These methods require considerably more memory and processing and are more complex.

For the single Gaussian model, only the mean and variance need to be computed. In order to adapt to changing backgrounds, mean and variance need to be computed over a window of time. While cheaper than other multi-sample techniques, computation of a running mean and standard deviation requires storage of $2KW$ images (when W is the temporal window size and K the number of MOG-components). If the input data matches the model assumption (i.e. Gaussian), setting the thresholds via the variance estimates is well statistically well motivated. Nevertheless, for this type of systems, N remains a critical "blending" parameter. It determines how fast the system adapts to changes and how sensitive it is to random fluctuations.

In per-frame processing approaches, an updated background model for each new frame is computed. These approaches require much less storage and much less computation. The background model is updated via temporal blending. [62] combines temporal difference and background subtraction techniques in the change detection phase and adapts to changes in the background by temporal blending.

To determine which of the many backgrounds is to be updated, systems with multiple backgrounds have implemented a separate higher-level process. Main components of these systems include background modelling and thresholding.

2.2.2 Illumination Invariance Review

During the segmentation phase many natural factors like shadows, illumination conditions (e.g. changing illumination color or illumination direction), cause problems. Details and further references on color image understanding are given in [66].

In terms of requirements concerning illumination-invariance, there is no significant difference if the application domain is tracking objects in a video scene, or image retrieval in a database. In both cases, one wishes to find methods to segment objects from the background. This should be possible independently of the current illumination conditions under which the image is taken (E.g. lights being switched on/off, changes in the camera gain, clouds covering the sun, different object location).

In the past, some work was done on illumination invariant image retrieval. In most of the presented experiments, the illumination itself and the change of illumination conditions have been well controlled.

Recent work regarding illumination invariance can split into two groups. One group deals with *global* color distribution analysis of a single image, the other with a set of the same image taken under various illumination conditions, and in *local* color distribution and structure analysis of a single image. There exist also methods combining these cues for indexing purpose.

Calculation of illumination invariants from *global color distribution* of a single image were proposed in [49, 115, 33, 107, 48, 51, 50]. Healey et al. use a linear illumination model and approximate the surface spectral reflectance function by linear combination of fixed basis functions. They find illumination invariant features by computing eigenvalues of moment matrixes of histograms of the image itself [107, 48], or eigenvectors of a matrix of correlation functions within and between sensor bands [51, 50]. Due to their finite-dimensional linear surface reflectance model, a change in illumination color results in a linear transformation of their feature representation. Also [115] follows a histogram approach. Instead of using a 3D RGB color histogram they use three 1D histograms along the principal components of the image data. Based on these histograms they derive energy, entropy, variance, and covariance features, and use these features as illumination-invariant color features. [33] represents an RGB image by three vectors corresponding to each color band. He than calculates the three angles between the different bands of both, the original image and the corresponding edge-image. Assuming a diagonal model of illumination change, these six angles are used as an illumination independent invariant.

[72, 124] follow another approach using global color distribution features. Their approach is characterized by analyzing a *set* of the same image taken under different illumination conditions. [72] distinguishes between illumination change due to varying illumination pose of objects in the scene (shading effects) and varying illumination color. To get a handle on varying illumination color they use a set of normalized images of the

same scene taken under different illumination conditions. Each image is represented in a vector. From this object-database, a corresponding eigenspace based on N eigenvectors is calculated. Objects projected in this eigenspace are considered being the same object; depending on their position in the eigenspace they were illuminated under specific illumination conditions. To find illumination invariant spectra, [124] filters out the two most dominant eigenvectors of a set of sensor outputs taken under different illuminations of the same surface (same reflectance model). Calculating an orthogonal projection from these eigenvectors, projects all outputs of the same reflectance spectrum onto the same illumination invariant spectrum, independent if generated under different illuminations.

Global normalization schemes are proposed by [78]. If illumination changes simultaneously over the entire scene, they propose a normalization scheme based on a nearly white area, which is always present in the image. This way, they get a handle on short-term illumination changes due to flickering light or automatic gain control in background subtraction algorithms. In the [24] the scene is assumed to be a ”gray-world” scene, such that the main principle component of all RGB color pixels in an image contiguously fall onto the gray axis in the color cube.

Calculation of illumination invariants from *local color distribution* of a single image are proposed in [106, 108]. Healey et al. use the same linear illumination model as described above, and again, a change in illumination color results in a linear transformation of their local feature representation. [106] calculate a feature matrix based on radial integration of sensor values. Its eigenvectors are used to test for illumination-invariant image retrieval. [108] calculates a moment matrix based on local histograms. Its eigenvalues are used for illumination invariants. [2] present the generalized color ratio (GCR) model. Neighborhood-based color ratios are proposed for illumination (spectral and intensity) invariant indexing. They assume that the variation in the illuminance color, spectral energy distribution function, and the surface reflectance function, can each be captured buy a small set of linearly independent basis functions. They also assume that neighboring points on a surface will receive equal amounts of illumination at the same time instant. [36] tries to classify each pixel in an image as part of a moving object, shadow, or background. Since each pixel can change its class over time, its value is modelled as a Gaussian mixture over time. The parameters of the distribution are unknown and are learned by an unsupervised technique: an incremental version of the expectation maximization (EM) algorithm.

Local normalization schemes are proposed by [87, 9, 10, 72]. [87] shows that if the light source can be expected to be almost white and a saturation value of object color is sufficiently large enough, the hue band of a color image is invariant against illumination change. [9, 10] normalize each pixel value by its region mean-value. As mentioned earlier, [72] distinguishes between illumination change due to varying illumination pose of objects in the scene (shading effects) and varying illumination color. To handle shading effects they normalize R and G values by the intensity (sum of R, G, B).

Based on the same linear surface reflectance model, Healey et al. present in [52] illumination-invariant recognition of local image *structure* by using spatial filters. In [120, 105, 117, 119] they incorporate both *color and spatial* information. [117, 119] follow an approach, based on symmetries in textures. Both apply spatial filter banks to an image and calculate energy matrixes [119], or opponent features [117], which compute cross-band correlation from the filter output. Finally, [109] *combines several* of these methods. In all their work, a change in illumination color results in a linear transformation of their feature representation.

Chapter 3

Background and Motivation

This chapter provides background information on and motivation for why it is beneficial to follow systematic system engineering methodology to build vision systems. In the second part, the system hardware-configuration choice is motivated, given application requirements.

3.1 System Engineering for Vision

This section illustrates why it is beneficial for the system's performance to apply performance characterization techniques to VSAM systems. Furthermore, this chapter describes in detail the underlying theory and ideas outlined in [97], and provides arguments why this approach is chosen. It finally summarizes various tools used to build the actual vision system which will be described later in chapter 4.

3.1.1 Classification of Performance Characterization Work

Papers referred in 2.1 can be categorized into two major classifications. Those that deal with evaluation of system performance and study the impact of a given module on overall system performance, and those that deal with the evaluation of components. In this work, both types of evaluations are pursued. It primarily follows the performance characterization methodology that is outlined in Ramesh et al [97] as it allows for individual component improvements and provides an understanding of the impact of a given component's use in the context of the total system. In general, performance characterization techniques can be used for the following purposes:

- To optimize control parameters of an algorithm that is being incorporated into a product

- To compare alternative components and identify the suitability of a given system for a customer application. Results of the evaluation are feed back to refine the design of the product.

- To identify limits of algorithm performance for a given application. Performance curves derived are provided to the customer (in some cases to identify scenarios in which the system is unavailable).

The goal is to illustrate the use of the methodology for systems analysis of a video surveillance system. In the following, systems analysis methodology is reviewed, and it is described how the methodology applies to a problem involving people detection and zooming. Subtle differences between the methodology described in [97] and the one followed in practice will be pointed out. Black box evaluation of video analysis systems is by itself an area of active research. Open questions include the choices of: the criterion functions useful to evaluate a given system output against groundtruth, experimental design and sampling methods for obtaining reasonable estimates of performance measures with minimal data, etc. In this work black-box evaluation will not be addressed.

The following subsection reviews the methodology and draws on material from [97]. It addresses the problem of analyzing the system and its modules, while automatically set-up tuning constants for a vision system with a chosen architecture. Although the methodology proposed does not address computational aspects and the choice of the system architecture, we will base our approach on insights gained from it.

3.1.2 Systems Analysis Given Chosen Algorithm Sequence

Let A denote an algorithm. At the abstract level, the algorithm takes in as input, a set of observations, call them input units U_{In}, and produces a set of output units U_{Out}. Associated with the algorithm is a vector of tuning parameters $\mathbf{T}$. The algorithm can be thought of as a mapping $A : (U_{In}, \mathbf{T}) \rightarrow U_{Out}$. Under ideal circumstances, if the input data is ideal (perfect), the algorithm will produce the ideal output. In this situation, doing performance characterization is meaningless. In reality, the input data is perturbed, perhaps due to sensor noise or perhaps because the implicit model assumed in the algorithm is violated. Hence, the output data is also perturbed. Under this case the inputs to (and the outputs from) an algorithm are observations of random variables. Therefore, the algorithm can be viewed as a mapping: $A : (\hat{U}_{In}, \mathbf{T}) \rightarrow \hat{U}_{Out}$, where the $\hat{}$ symbol is used to indicate that the data values are observations of random variables. This leads to the verbal definition of performance characterization with respect to an algorithm:

"Performance characterization or Component Identification for an algorithm has to do with establishing the correspondence between the random variations and imperfections on the output data and the random variations and imperfections on the input data."

More specifically, the essential steps for performance characterization of an algorithm include:

1. the specification of a model (with parameter $\mathbf{D}$) for the ideal input data.
2. the specification of a model for the ideal output data.
3. the specification of an appropriate perturbation model (with parameter $\mathbf{P_{In}}$) for the input data.
4. the derivation of the appropriate perturbation model (with parameter $\mathbf{P_{Out}}$) for the output data (for the given input perturbation model and algorithm).
5. the specification and the evaluation of an appropriate criterion function (denoted by Q_{Out}) relative to the final calculation that the algorithm makes to characterize the performance of the algorithm.

The main challenge is in the derivation of appropriate perturbation models for the output data and relating the parameters of the output perturbation model to the input perturbation, the algorithm tuning constants, and the ideal input data model parameters. This is because the specification of the perturbation model must be natural and suitable for characterization of the performance of the subsequent higher level process. Once an output perturbation model is specified, estimation schemes for obtaining the model parameters have to be devised. In addition, the model has to be validated, as theoretical derivations may often involve approximations.

The ideal input data is often specified by a model parameter vector $\mathbf{D}$, and the algorithm is often an estimator of these parameters. Please note that the ideal input data is nothing but a sample from a population of ideal inputs. The characteristics of this population, i.e. the exact nature of the probability distributions for $\mathbf{D}$, is dependent on the problem domain. The process of generation of a given ideal input can be visualized as the random sampling of a value of $\mathbf{D}$ according to a given probability distribution $F_{\mathbf{D}}$.

Let $\mathbf{P_{In}}$ denote the vector of parameters for the input perturbation model. Let $Q_{Out}(\mathbf{T}, \mathbf{P_{In}}, \mathbf{D})$ denote the criterion function that is to be optimized [1]. Then the problem is to select $\mathbf{T}$ so as to optimize the performance measure Q, over the entire population, that is given by:

$$Q(\mathbf{T}, \mathbf{P_{In}}) = \int Q_{Out}(\mathbf{T}, \mathbf{P_{In}}, \mathbf{D}) dF_{\mathbf{D}} \tag{3.1}$$

In the situation where the perturbation model parameters, $\mathbf{P_{In}}$, are not fixed, but have a specific prior distribution then one can evaluate the overall performance measure by

[1] Note that the input data $\hat{U}_{In}$ is not one of the parameters in the criterion function. This is correct if no input-data violate any of the assumptions about the distribution(s) of $\mathbf{D}$ and $\mathbf{P_{In}}$.

integrating out $\mathbf{P_{In}}$. That is:

$$Q(\mathbf{T}) = \int Q(\mathbf{T}, \mathbf{P_{In}}) dF_{\mathbf{P_{In}}} \tag{3.2}$$

Having discussed the meaning of performance characterization with respect to a single algorithm, we now turn to the situation where simple algorithms are cascaded to form complex systems.

Let Φ denote the collection of all algorithms. Let $A^{(i)} \in \Phi$, then $A^{(i)} : U_{In}^{(i)} \rightarrow U_{Out}^{(i)}$ is the mapping of the input data $U_{In}^{(i)}$ to the output $U_{Out}^{(i)}$. Note that the unit for $U_{In}^{(i)}$ may not be the same as the unit for $U_{Out}^{(i)}$ and perturbations in the input unit type causes perturbations in the output unit type. A performance measure, $Q^{(i)}$, is associated with A_i. Associated with each algorithm is the set of input parameters $\mathbf{T^{(i)}}$. The performance measure is a function of the parameters $\mathbf{T^{(i)}}$ as well.

An algorithm sequence, S, is an ordered tuple:

$$S : (A^{(1)}, A^{(2)}, \ldots, A^{(n)})$$

where n is the number of algorithms utilized in the sequence. Associated with an algorithm sequence is a parameter vector sequence

$$\mathbf{T} : (\mathbf{T}^{(1)}, \mathbf{T}^{(2)}, \ldots, \mathbf{T}^{(n)})$$

and a ideal input data model parameter sequence:

$$\mathbf{D} : (\mathbf{D}^{(1)}, \mathbf{D}^{(2)}, \ldots, \mathbf{D}^{(n)})$$

The performance at one step of the sequence is dependent on the tuning parameters, and the perturbation model parameters at all previous stages. So

$$Q_i = f_i(\mathbf{T}^{(i)}, \mathbf{T}^{(i-1)}, \ldots, \mathbf{T}^{(1)}, \mathbf{P_{In}}^{(i-1)}, \ldots, \mathbf{P_{In}}^{(1)}).$$

The overall performance of the sequence is given by:

$$\begin{aligned} Q_n(\mathbf{T}, \mathbf{P_{In}}) &= f_n(\mathbf{T}^{(n)}, \mathbf{T}^{(n-1)}, \ldots, \mathbf{T}^{(1)}, \\ &\quad \mathbf{P_{In}}^{(n-1)}, \ldots, \mathbf{P_{In}}^{(1)}). \end{aligned}$$

The free parameter selection problem can now be stated as follows: *Given an algorithm sequence S along with the parameter vector sequence $\mathbf{T}$ and performance measure Q_n, select the parameter vector $\mathbf{T}$ that maximizes Q_n* . Note that Q_n is actually the integral:

$$\begin{aligned} &Q_n(\mathbf{T}, \mathbf{P_{In}}) = \\ &\int \cdots \int f_n(\mathbf{T}^{(n-1)}, \ldots, \mathbf{T}^{(1)}, \mathbf{P_{In}}^{(n-1)}, \ldots, \\ &\mathbf{P_{In}}^{(1)}, \mathbf{D}^{(n-1)}, \ldots, \mathbf{D}^{(1)}) dF_{\mathbf{D}^{(n-1)}} \ldots dF_{\mathbf{D}^{(1)}}. \end{aligned}$$

Note that at each stage a different set of prior distributions $F_{\mathbf{D}^{(i)}}$ comes into play. Also, the perturbation model parameters ${\mathbf{P}_{\mathbf{In}}}^{(i)}$ is a function $g_i(T^{(i-1)}, {\mathbf{P}_{\mathbf{In}}}^{(i-1)}, \mathbf{D}^{(i-1)}, A^{(i-1)})$. In other words, the perturbation model parameters at the output of stage i are a function of the tuning parameters at stage $i-1$, the input perturbation model parameters in the stage $i-1$, the ideal input data model parameters, and the algorithm employed in the stage $i-1$. It is important to note that the functions g_i depend on the algorithm used. No assumption is made about the form of the function g_i.

The derivation of the optimal parameters $\mathbf{T}$ that maximize $Q_n(\mathbf{T}, \mathbf{P}_{\mathbf{In}})$ is rather tedious and involved. Therefore in practice the thresholds $\mathbf{T}$ are selected in each individual stage relative to the final task. For example, in [96], thresholds for a sequence of operations involving boundary extraction and linking were chosen relative to the global classification task of extracting building features to satisfy a given miss-detection rate for building feature detection and a given false alarm rate for clutter boundary pixels. In the following work that focuses on video surveillance a similar strategy is adopted: Pruning thresholds will be set up by defining probability of missing valid hypotheses and probability of false hypotheses as criteria. Please note that these criterion functions are essentially functions of the ideal parameters $\mathbf{D}$'s and one has to integrate over the prior distribution of the $\mathbf{D}$'s.

3.1.3 Tools Used for Systems Analysis

To facilitate the propagation of models, tools defined in [97] are used along with other numerical methods (e.g. bootstrap ([31])) to perform the characterization with analytical statistical models. The tools/steps used include:

- Distribution propagation: The input to an algorithm (i.e. an estimator) is characterized by one or more random variables specifying the ideal model, its parameters, and by a noise model with a given probability density function (pdf) [89]. The output distribution is derived as a function of the tuning constants, the input model parameters, and the noise model parameters.

- Covariance propagation: The algorithm output is thought of as a non-linear function of the input data and noise model parameters. Liberalization is used to propagate the uncertainty in the input to the output uncertainty. Care should be taken while using this tool since the approximation may be only good when the liberalization and first order error approximations are valid. For details please see [55].

- Empirical methods: Statistical re-sampling techniques (e.g. bootstrap) are used to characterize the behavior of the estimator. For example, the bias and uncertainty can be calculated numerically (Please see Cho and Meer [21] for a description of edge detection performance characterization using bootstrap.). Monte-Carlo methods are used for the verification of theoretical results derived in the previous two steps.

- Statistical modeling: Modeling at the level of sensor errors (Gaussian perturbations in input), Prior models for 3D geometry, spatial distribution of objects, and modeling of physical properties (e.g. constraints on the types of light sources in the scene), etc. The related literature is rather vast and encompasses methods from fields such as Bayesian statistics, Spatial statistics, and Computer Vision.

3.2 Dual-Camera Video Surveillance System

As outlined in 1.1.2 the main goal of the application presented in this work is to monitor a wide area, localize people and provide a high resolution image of their faces/heads for further processing, no matter how far they are apart from the camera locations. Furthermore, we are interested to provide location estimates of any person in the scene and be able to switch attention from one to the other. To achieve this goal it is important to continuously monitor the wide region of interest while simultaneously providing high-resolution images of a person in the scene.

Obviously, two kind of cameras are needed: one processing the activities in the monitored area to detect and localize all people, and another class of active cameras which in parallel focuses on the people detected in the scene [25]. Depending on the requirements in terms of how many people should simultaneously be monitored closely such that a camera zooms onto its face the number of active cameras providing pan, tilt, zoom features is determined. Nevertheless, this work does not focus on control issues and strategies.

This work rather focuses on how to acquire data from the region of interest, which help to control active cameras as laid out in the requirements. E.g. in a way that it can capture the face respectively head of any person in the scene such that it is contained in the frame to a maximal extent.

The question is what kind of camera system to use in order to fulfill the task best in terms of coverage, real-time and precision. For reasons explained later, we decided for the omni-directional camera[2] as proposed by Nayar in [84] and [83]. The system is comprised of an orthographic lens attached to a standard camera and a parabolic mirror. In a concept study, an first version of our system [25] uses a spherical mirror

[2] Commercially available under the name ParaCamera

3.2.1 Omni-Directional Camera

Before discussing the omni-directional camera, imaging systems that seek to cover wide fields of view are reviewed. A detailed review of omni-directional viewers can be found in [81]. The following review is in parts adapted from [84].

Most imaging systems in use today have a lens attached to a video camera. For most camera lenses, the image projection model is perspective with a single center of projection. Since the imaging devices like CCD arrays are of finite size and the camera lens occludes itself while receiving incoming rays, lenses typically have a small field of view. This field of view does not correspond to a hemisphere but rather to only a small cone. To simultaneously sense a wide field of view, one could think of arranging a number of cameras accordingly, each one pointing in a different direction. However, such a configuration proves infeasible, since the centers of projection reside inside each lens.

Another solution is to rotate the entire system comprised of camera and lens about its center of projection. To obtain a single panoramic view of the scene, the sequence of images acquired by rotation are put together (see [20], [77], [69], [132]). Unfortunately, rotating imaging systems require the use of moving parts and precise positioning. If location information in 3D is to be derived from these images, precise registration becomes an issue. Another drawback lies in the total time required to obtain such a panoramic image. The domain for rotating systems is therefore restricted to static scenes and non-real-time applications.

Using a fish-eye lens ([126],[79]) instead of a conventional camera lens is another approach to wide-angle imaging. Such a lens has a very short focal length such that objects within as much as a hemisphere can be viewed. Use of fish-eye lenses for wide-angle imaging is proposed in [88] and [70], among others.

Unfortunately, it is difficult to design a fish-eye lens that ensures that all incoming principal rays intersect at a single point to yield a fixed viewpoint (see [81] for details). This has dramatic drawbacks in calibrating such a system and in performing real time mapping between image coordinates and real world coordinates for precise location estimation. In addition, to capture a hemispherical view, the fish-eye lens must be quite complex and large, and hence expensive.

A catadioptric imaging system uses a reflecting surface to enhance the field of view (like the rear-view mirror in a car). However, the viewpoint and field of view is a rather complex function of the shape, position, and orientation of the reflecting surface. It is easy to construct a configuration that uses one ore multiple mirrors to dramatically increase the field of view of the imaging system, but is hard to keep the effective viewpoint fixed in space. See [130],[59],[131], and [81] for examples on catadioptric image sensors. [82]) presents the complete class of catadioptric imaging systems that satisfy the single viewpoint constraint.

While all of the above approaches use mirrors placed in the view of perspective lenses, the omni-directional camera used here and proposed by Shree Nayar [83] uses an orthographic lens. For orthographic rather than perspective projection, the geometrical mappings between the image, the mirror and the world are invariant to translations of the mirror with respect to the imaging system. This greatly simplifies calibration and mapping between image and real world coordinates.

Figure 3.1 shows the underlying geometric relations. In [84] a surface function for a mirror is derived by solving a first-order differential equation such that

$$z(r) = \frac{r_m^2 - r^2}{2r_m} \tag{3.3}$$

The corresponding shape of the surface ensures that each ray, which virtually projects into the omni-directional viewpoint, is reflected parallel to the optical axis. The image is then captured by an orthographic imaging lens.

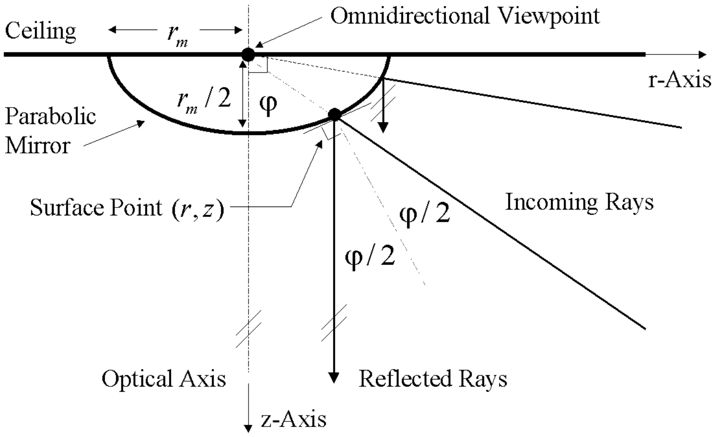

Figure 3.1: Underlying geometry, which is used to derive the surface of the parabolic mirror. Please note, that each ray, which virtually projects into the omni-directional viewpoint is reflected parallel to the optical axis. The image is then captured by an orthographic imaging lens.

3.2.2 Active Pan-Tilt-Zoom Camera

To follow and zoom onto a person's heads, we use an active camera that allows to control pan, tilt and zoom independently. Currently, control functions are implemented for a Sony EVI D30. However, this work does not focus on control issues, but on the analysis necessary to optimally set pan, tilt and zoom parameter to capture a persons head given a location estimate and corresponding uncertainties. For details on how to calibrate the zoom unit in a medical application that uses Sony EVI D30, please refer to [38].

Chapter 4

The System: Architecture, Design & Analysis

This chapter explains how the application specific priors and requirements such as real-time operation and adaptive zooming influence the choice of the system architecture. It illustrates how perturbations can be propagated through the chosen surveillance system configuration involving change detection, people detection, people location determination and camera parameter estimation, and points out how this approach differs from the work outlined in [97]. The first part involves the design issues (choice of the system configuration given application requirements and priors), and emphasizes on statistical modeling, uncertainty propagation and on how to consider prior information. The second part describes in detail the chosen algorithm and explains the transforms module-wise. Furthermore, it involves the systems analysis of a chosen system configuration.

4.1 Systems Architecture Choice, Priors & Requirements

We have seen that in the computer vision literature there has been a significant amount of work in evaluation of modules/components. However, there has been limited work on making systems design choices from user requirements. The more recent trend in the community is to emphasize statistical learning methods, more appropriately Bayesian methods for solving computer vision problems (See for example [80]). However, there still exists the problem of choosing the right statistical likelihood model and right priors that suit an application. Even if this were possible, it is still computationally infeasible to satisfy real-time application needs. In the context of video analysis systems, real-time considerations play a big role in the design of video processing systems.

Sequential decomposition of the total task into manageable sub-tasks (with reasonable computational complexity) and the introduction of pruning thresholds, is the common way to tackle the problem. This introduces problems because of the difficulty in approximating the probability distributions of observables at the final step of the system so that Bayesian inference is plausible. This approach to perceptual Bayesian inference has been attempted, (see for example [97], [74]). [97]'s work places more emphasis on performance characterization of a system, while [74] attempted Bayesian inference (using Bayesian networks) for visual recognition. The idea of gradual pruning of candidate hypotheses to tame the computational complexity of the estimation/classification problem has been presented in [5]. Learning decision trees to perform object detection (by gradually reducing the uncertainty in a step-wise fashion, wherein each pruning step has probability of miss-detection approximately zero while the probability of false alarm is reduced after each application of a decision rule) is discussed in [4] and [34]. Note that none of the works identifies how the sub-tasks (e.g. feature extraction steps) can be chosen automatically given an application context. The approach proposed in this work, involves the following key-steps:

System Configuration choice: The first step is to choose the modules for the system. This is done by use of context (in other words: application specific prior-distributions for object geometry, camera geometry, and error models, illumination models). Real-time constraints are satisfied by choosing pruning methods or indexing functions that restrict the search space for hypotheses. The choice of the pruning functions is derived from the application context and prior knowledge. The choice of the indexing function is not necessarily critical, except that the following criterion is met. The indexing function has to be of a form, which simplifies the computation of the probability of generating a false hypothesis or the probability of missing a true hypothesis as a function of the tuning constants. To satisfy the accuracy constraint hypothesis verification and 3D-parameter estimation steps are employed. Bayesian estimation is used to evaluate candidate hypotheses and estimate object parameters by using a likelihood model, $P(measurements|hypothesis)$, that takes into account the effects of the pre-processing steps and tuning parameters. Note that this likelihood model is actually derived from the statistical characterization step that is described below. The indexing step provides computational efficiency, while the hypothesis verification and estimation step addresses accuracy.

Statistical Model Derivations, Performance Characterization and Model Validation: The second step involves the application of the methodology described above to derive statistical models for errors at various stages in the chosen vision-system configuration. That allows for quantifying the indexing step and tuning the parameters to achieve a given probability of miss-detection and false alarm rate. In addition, a validation of theoretical models for correctness (through Monte-Carlo simulations) and closeness to reality (through real experiments) is performed. For the given system configuration choice, a statistical analysis is conducted to setup the tuning constants at the indexing steps, to derive likelihood models for feature measurements that are used in the hypothesis verification and in the estimation step, and to obtain the uncertainty of the estimate provided by the hypothesis verification step.

4.1.1 Application Requirements

The task of the dual-camera surveillance system is to continuously detect locations of people in the scene and provide zoomed-in high resolution images of the head of a person present in the room. The current implementation, which uses only one foveal camera, tracks the first person entering the scene. Nevertheless, the system is extendable for integration of multiple foveal cameras that could follow simultaneously multiple people being tracked in the omni-view. The foveal images could represent the input to higher-level vision modules, e.g. face recognition, compaction and event logging. However, that is not part of this work.

The application requirements are as follows: 1) Real-time performance on a low-cost PC [1], 2) Person miss-detection rate equal to or less than α_m, 3) Person false-alarm rate equal to or less than α_f, 4) Adaptive zooming of a person irrespective of the background scene structure (with maximal possible zoom based on uncertainty of person attributes estimated (e.g. location in 3D, height, etc). The performance of the result is characterized by the face resolution attainable in terms of the area of face pixel region in the foveal view (as a function of distance, contrast between background and object, sensor noise variance, and resolution). It is also characterized by the bias in the centering of the face. In addition to these requirements, the following assumptions can be made about scene structure: A) The scene illuminant consists of light sources with similar spectrum (e.g. identical light sources in an office area), B) the number of people to be detected and tracked is bounded, and the probability of occlusion of persons (due to other persons in the Omni-view) is small, and C) people are standing upright.

4.1.2 System Hardware Configuration

To continuously monitor the entire scene we use a passive catadioptric sensor (OmniCam [84]) mounted below the ceiling. To obtain high-resolution images of an object of interest (in our case the face of a detected person in the region of interest) we use a active pan-tilt-zoom camera (foveal camera). Both sensors work together in a collaborative

[1] Not all system resources in the PC are allocated for visual processing.

fashion to achieve these two goals simultaneously. The catadioptric sensor consists of two parts: A parabolic mirror and a standard CCD camera looking into it. The image obtained through the system provides an omni-directional view of a wide area as seen from the ceiling. Omni-images are used to detect and estimate the precise location of a given person's foot in the room. This information is used to identify the pan, tilt and zoom-settings for a high-resolution foveal camera, which is then directed towards the person's head. Figure 4.1 shows the system's interface: the overview image, and the high resolution zoomed image of the detected person's face.

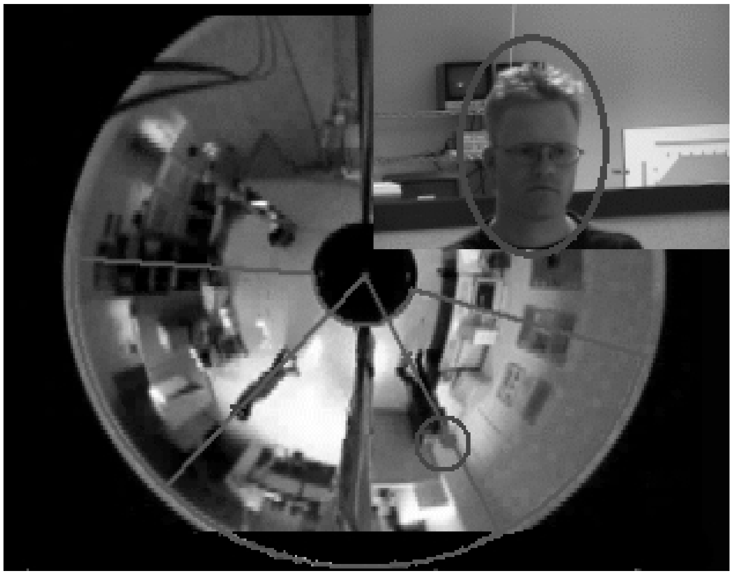

Figure 4.1: Top: Omni-directional overview image. Red sector: Region of interest. Radial lines (green and red) show detected persons. Crosses denote estimated foot/head position. Insert: Foveal camera view.

The omni-directional camera projection-geometry satisfies the single-view point constraint and it simplifies calibration. It also simplifies the relation between the world coordinates of a given point on the ground plane and the corresponding image point [85],[84]. Figures 4.2, and 4.3 illustrate in detail the geometric relations between the two cameras, and how the output of the omni-image-processing module (location estimation of the person in the scene) can be used to estimate the pan, tilt and zoom parameters of the foveal camera.

We denote the geometric model parameters as shown in table 4.1

During the calibration step (combination of real world and image measurements) H_o, H_f, D_c, r_m and (x_c, y_c) are initialized.

Let α, and β be the foveal camera control parameters for the tilt respectively pan angle, and D_p the projected real world distance between the foveal camera and the person. Assuming, the person's head is approximately located over his/her feet, and using basic trigonometry, it can easily be seen that D_p, α (see Figure 4.2), and β (see Figure 4.3) are

Table 4.1: Geometric model parameters (see also Figure 4.2, 4.3). Capital variables are variables in 3D, and small variables are given in image coordinates. $\hat{}$ (hat) indicates data values being observation of a random variable.

$\hat{H}_o$	height of OmniCam above floor (meters)
$\hat{H}_f$	height of foveal camera above floor (meters)
$\hat{H}_p$	person's height (meters)
$\hat{R}_h$	person's head radius (meters)
$\hat{R}_p$	person's foot position in world coordinates (meters)
$\hat{S}_p$	person's size (meters)
$\hat{D}_c$	on floor projected distance between cameras (meters)
$\hat{D}_p$	on floor projected distance between foveal camera and person (meters)
$\hat{D}'_p$	direct distance between foveal camera and person's center of face (meters)
$(\hat{x}_c, \hat{y}_c)$	position of OmniCam center, (in omni-image, pixel coordinates, Cartesian)
$(\hat{x}, \hat{y})$	position in omni space, (in omni-image, pixel coordinates, Cartesian)
$\hat{r}_m$	radius of parabolic mirror (in omni-image) (pixels)
$\hat{r}_h$	distance person's head – (in omni-image) (pixels)
$\hat{r}_f$	distance person's foot – (in omni-image) (pixels)
$\hat{s}$	projected size of person – (in omni-image) (pixels)
$\hat{k}$	number of pixels a person projects onto omni image plane
$\hat{\vartheta}$	angle between the person and the foveal camera relative to the OmniCam image center (please see Figure 4.3)
$\hat{\theta}_l$	angle between the left side of person and the foveal camera relative to the OmniCam image center.
$\hat{\theta}_r$	angle between the right side of person and the foveal camera relative to the OmniCam image center.
$\hat{\theta}$	angle between the radial line corresponding to the person and the zero reference line (please see Figure 4.3)
$\sigma^2_{(.)}$	Denotes variance of the variable used in the subscript
$\hat{\alpha}$	Tilt angle
$\hat{\beta}$	Pan angle
Z	Zoom factor
q'	Number of pixels summed within sector of interest in radial direction
	to generate feature $\hat{M}_{theta}$.
s'	Number of pixels summed in direction orthogonal to radial direction
	to generate feature $\hat{M}^{\top}_{r,\theta}$.
r'_i	Number of pixels summed in ith sub-sector along radial line in direction
$\hat{M}_\theta$	Sum of d^2 along radial line
$^i\hat{M}_\theta$	Sum of d^2 along radial line within sub-sector
$\hat{M}^{\top}_{r,\theta_f}$	Sum in orthogonal direction bounded by θ_l and θ_r
$b^i(\theta)$	Binary sub-sector profile

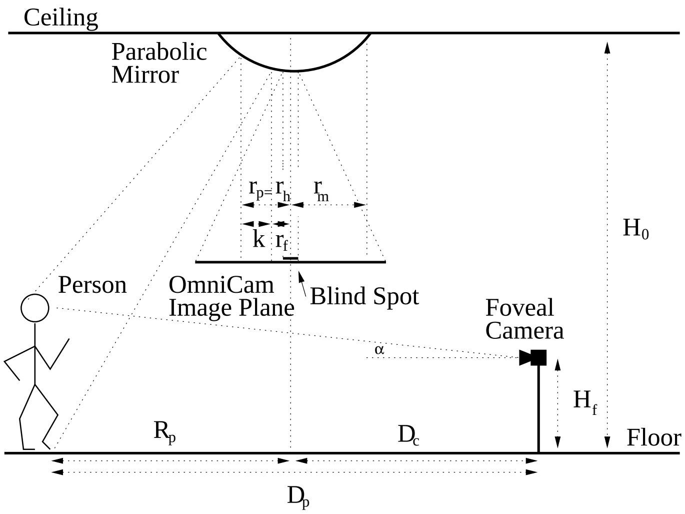

Figure 4.2: Geometry (viewed from the side) in real world and omni-image coordinates. OmniCam is looking into the parabolic mirror at the ceiling.

equal to:

$$\begin{aligned} D_p &= \sqrt{D_c^2 + R_p^2 - 2D_cR_p\cos(\vartheta)} && (4.1) \\ \tan(\alpha) &= \frac{H_p - R_h - H_f}{D_p} && (4.2) \\ \sin(\beta) &= \frac{R_p}{D_p}\sin(\vartheta) && (4.3) \end{aligned}$$

where ϑ is the angle between the person and the foveal camera relative to the OmniCam position.

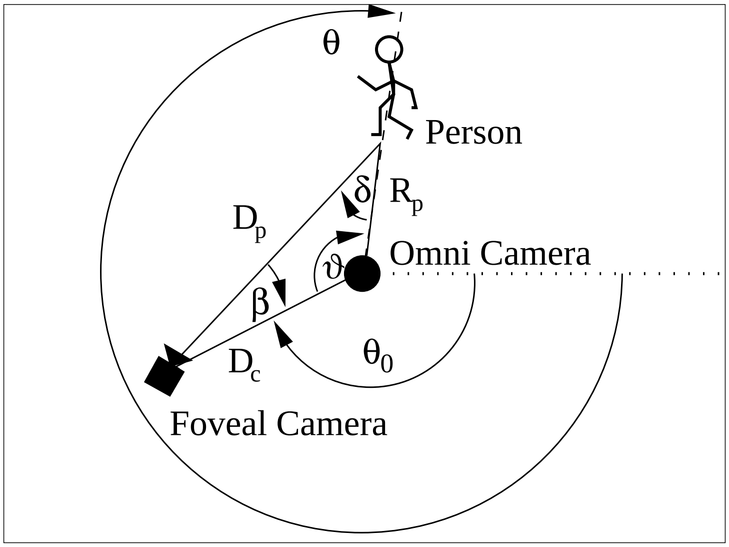

Figure 4.3: 3D geometry, viewed from atop.

As illustrated in Figure 4.2 the relationship between the person location in world coordinates can be related to the measurements for the foot and head coordinates in the image plane assuming that the person is standing upright.

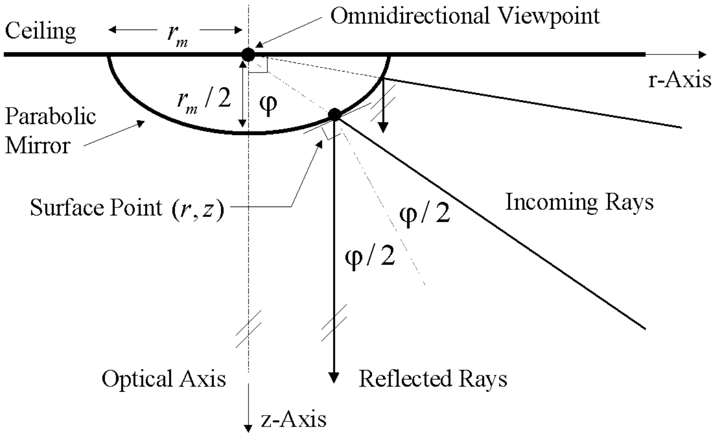

Figure 4.4: Optics at parabolic mirror.

In Nayar's work on the omnidirectional-camera [84] the equation for the mirror surface satisfying the single viewpoint constraint (please see also Figure 4.4) is derived as

$$z(r) = \frac{r_m^2 - r^2}{2r_m} \tag{4.4}$$

such that with

$$\frac{R_p}{H_o} = \frac{r_p}{z(r)} \tag{4.5}$$

we can derive

$$R_p = 2aH_o \quad \text{with} \quad a = \frac{r_m r_f}{r_m^2 - r_f^2} \tag{4.6}$$

$$R_p = 2b(H_o - H_p) \quad \text{with} \quad b = \frac{r_m r_h}{r_m^2 - r_h^2} \tag{4.7}$$

Therefor, the ultimate goal is to estimate the radial distance r_p of the foot from the omni-camera projection in the omni-image and map it to 3D real world distance R_p. Of course, ϑ needs to be estimated as well; it is invariant to the transformation, though.

In the following we describe the design phase as illustrated in the left block in Figure 1.1. In the system engineering process, the developer might loop through this phase multiple times until the analysis and verification steps following the design phase prove that the system requirements are satisfied.

Before describing the details of how the application requirements translate to the design of individual modules, let's discuss the prior distributions (of the 3D scene) reasonable for the given application and identify how these priors induce image priors. The choices of the various estimation steps in the system are motivated from these image priors and real-time requirements. The camera control parameters (pan and tilt) are selected based on the location estimate and its uncertainty (that is derived from statistical analysis of the estimation steps) to center the person's head in the foveal image frame. The zoom parameter is set to maximum value possible so that the camera view still encloses the person's head within the image.

4.1.3 Priors, Camera Models, Illumination Models

The general Bayesian formulation of the person detection and location estimation problem is as follows: Given the Omni-image data I_o, mean of the reference image corresponding to the static scene B_o, and its covariance Σ_{B_o}, the objective is to estimate the parameter vector $\mathbf{\Theta}$ that maximize the aposteriori probability $P(\mathbf{\Theta}|I_o, B_o, \Sigma_{B_o})$. Here, vector $\mathbf{\Theta}$ represents: N_p the number of persons in the scene and their attributes, e.g. for the i. person in the scene: foot position in a reference coordinate system specified by $(R_{p,i}, \theta_i))$, height $H_{p,i}$ and size $S_{p,i}$. That is:

$$\hat{\Theta} = \underset{\Theta}{argmax} P(\Theta|I_o, B_o, \Sigma_{B_o}) \tag{4.8}$$

Typically, one uses Bayes rule to convert this posterior probability to a product of likelihood term and prior probability and uses independence conditions to factor the joint density into a product of simpler terms. In this context, the prior probabilities are over Θ_i's that decompose further into a product of terms defining the prior probability of a person at a given location specified by $R_p(i), \theta_i$ and with height and size of $H_p(i), S_p(i)$ in 3D. It is possible to define a spatial Markov model for the priors in 3D (indicating a marked point process where points repel (See for example [112])) and define likelihood term describing the image observations and estimate the Bayes optimal number of people and their locations in the image. The general Bayesian formulation of the person detection and location estimation problem does not suit the real-time constraints imposed by the application. This approach is to use this formulation only after a pruning step that rules out a majority of false alarms. This is done by designing an indexing step motivated by the 2D image priors (region size, shape, and intensity characteristics) induced by the prior distribution in the 3D scene. The prior distributions for person shape parameters: size, height, and his/her 3D location are reasonably simple. These priors on the person model parameters induce 2D spatially variant prior distributions in the projections (e.g., the region parameters for a given person in the image depend on the position in the image).

Its form depends on the camera projection model and the 3D-object shape[2]. In addition to shape priors, the image intensity/color priors are of importance in this application. Typically, assumptions are made about the object intensity (e.g. homogeneity of object since people can wear variety of clothing and the color spectrum of the light source is not necessarily constrained). However, in this surveillance application, the background is typically assumed to be a static scene (or a slowly time varying scene) with known background statistics (Gaussian mixtures are typically used to approximate these densities). In chapter 5 it will be shown how to relax these constraints while maintaining analysis and remaining modules. To handle shadowing and illumination changes, these distributions are computed after calculating an illumination invariant measure from a local region in an image. The prior distributions of the spectral components of the illuminant in our application are assumed to have the same but unknown spectral distribution. Finally, the noise model for the CCD sensor noise is to be specified. This is typically chosen to be i.i.d. zero mean Gaussian noise in each color band. Figure 4.5 illustrates the dependencies of the priors.

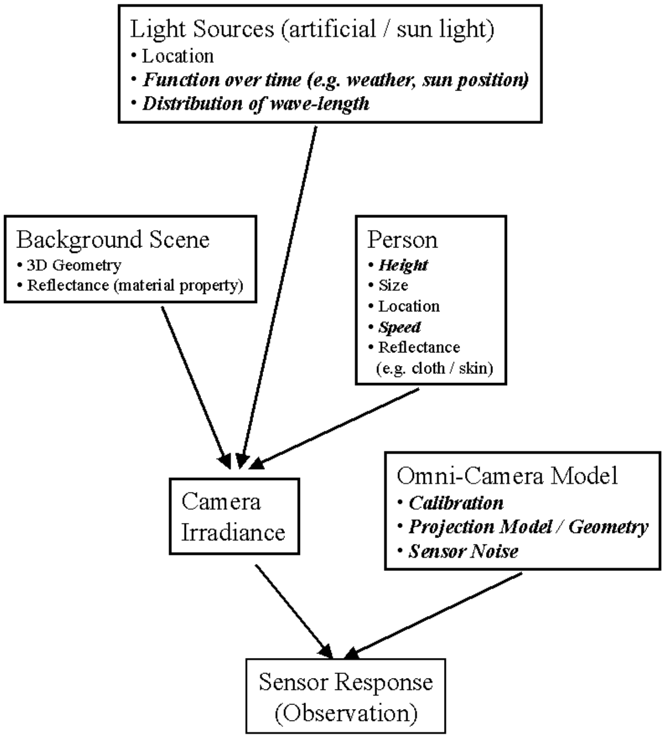
Light Sources (artificial / sun light)
• Location
• Function over time (e.g. weather, sun position)
• Distribution of wave-length
Background Scene
• 3D Geometry
• Reflectance (material property)
Person
• Height
• Size
• Location
• Speed
• Reflectance (e.g. cloth / skin)
Camera Irradiance
Omni-Camera Model
• Calibration
• Projection Model / Geometry
• Sensor Noise
Sensor Response (Observation)

Figure 4.5: Block diagram: Prior dependencies. Distributions of bolt parameters are modeled.

[2] For this application we found that modeling the person as an upright cylinder is a reasonable approximation.

4.2 Design & System Software Configuration

The software is composed seven functional modules, which contain eight major transforms T1–T8:

- Calibration
- Illumination-invariant measure computation at each pixel (T1)
- Distance measure between current image and a background model at each pixel (T2)
- Indexing functions to select sectors of interest (T3)
- Feature generation (T4)
- Statistical estimation of person parameters (e.g. foot location estimation) (T5)
- Mapping between image space and real world (T6)
- Foveal camera control parameter estimation (T7)
- Zoom parameter estimation T8)

Figure 4.6 illustrates the step by step transformations applied to the input. Before we explain the transforms in detail in the following subsection, we briefly summarize the chain of transforms and illustrate how the priors influence these transforms.

The input color image, $\hat{I}_o(x,y) = \{\hat{R}(x,y), \hat{G}(x,y), \hat{B}(x,y)\}$, is transformed ($T1 : R^3 \rightarrow R^2$) to compute an illumination invariant measure $\hat{r_c}(x,y), \hat{g}_c(x,y)$. The statistical model for the distribution of the invariant measure is influenced by the sensor noise model and the transformation $T1(.)$. The invariant measure mean $\hat{b}_o(x,y) = (\hat{r}_b(x,y), \hat{g}_b(x,y))$ and covariance matrix $\Sigma_{\hat{r}_b,\hat{g}_b}(x,y)$, is computed off-line at each pixel (x,y) from several samples of $\hat{R}(x,y), \hat{G}(x,y), \hat{B}(x,y)$ for the reference image of the static scene.

A change detection measure $\hat{d}^2(x,y)$ image is obtained by computing the Mahalanobis distance (denoted by transform T2(.)) between the current image data values $\hat{r_c}(x,y), \hat{g}_c(x,y)$ and the reference image data $b_o(x,y)$. This distance image is used as input to the indexing functions $T3(.)$ which discards the radial lines parameterized by their angle θ by choosing Canny's [18] hysteresis-thresholding parameters that satisfy a given combination of probability of false alarm and miss-detection values. $T4(.)$ generates features. The result is a set of regions with high probability of significant change along with features (i.e. change detection measures) in the sectors of interest.

At this point, we employ a Bayesian estimation technique for person localization that uses the 3D-model information, camera geometry information, and priors on objects, and

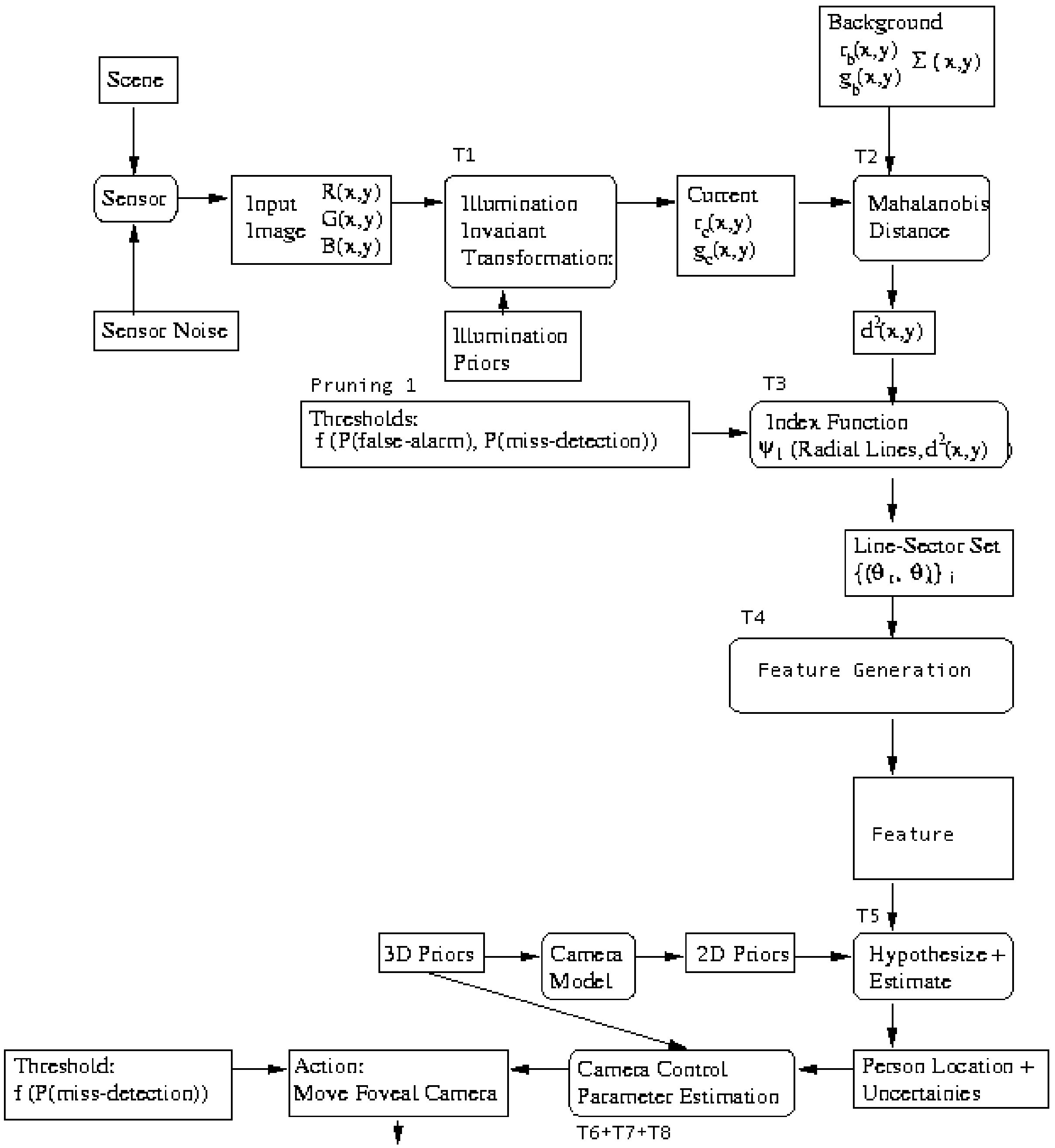

Figure 4.6: Block diagram: Boxes with rounded corners represent transformations while boxes represent data objects.

3D location to estimate the number of objects and their positions. The optimum position of the person foot location along a hypotheses radial line is estimated by minimizing a Bayesian error criterion. In essence, the best hypothesis out of multiple possible location hypotheses is estimated. The cost function used is nonlinear because the projection geometry implies that the projected length of the person height varies spatially with radial position. For illustrations of the Bayes error as a function of the radial index parameter please see Figure 4.9. It also illustrates the typical profile for the projected person height as a function of index. We will see that it is possible to derive the uncertainty of the estimated index by using numerical techniques in the following section on system analysis.

The last step is to estimate the control parameters for the foveal camera based on the location estimates and uncertainties. Equations 4.2 and 4.3 give the expressions used to compute the pan and tilt angles from given values for the 3D world coordinates of the foot position of the person, the height of the person, and the calibration parameters. The 3D-world coordinates for the foot position is derived from the image coordinate of the projected foot position (computed via the Bayesian location estimator). Transform T8 will demonstrate how the analysis results will influence the optimal zoom setting given application requirements

The following sections illustrate in detail the system design and configuration phase as shown in the left part of Figure 1.1. For each module, it is explained how requirements and prior assumptions influence and motivate the transforms chosen.

4.2.1 T1: Illumination Invariant Measure Estimation

The first module transforms the sensor output such that it can later be compared with a corresponding background model for segmentation purpose.

Since the camera used is equipped with an automatic gain control, and since the application requires to precisely segment people from background including shadow, an illumination intensity invariant representation has to be found to meaningfully compare the current image with a background representation.

The illumination prior assumption is that the scene contains multiple light sources with no constraint on individual intensities but with the same spectral distribution such that the sensor model can be modelled as follows: The amplitude for the ith channel sensor response can be written as (e.g. see [106])

$$C_i(x, y, t) = \sum_{j=1}^{N} a_j \int_{\lambda} c_j(t) l_j(t) s(x, y, \lambda) f_i(\lambda) d\lambda \tag{4.9}$$

Please note, that intensity $C_i(x, y, t)$ of the ith band at position (x, y) at time t is a function of the spectral density $l_j(\lambda)$ of the jth light source, the spectral surface reflection $s(x, y, \lambda)$, the spectral sensitivity $f_i(\lambda)$ of the ith sensor band, the mixture parameter $0 \leq a_j \leq 1$ for the jth light source, and the cut-off factor $0 \leq c_j \leq 1$ for the jth light source.

To compensate for shift in the gain control and shadows which are often present in the image a shadow invariant representation of the color data ([129], pp.347) is employed. The illumination normalizing transform $T1 : R^3 \rightarrow R^2$ appropriate to our assumption is:

$$r = \frac{R}{R + G + B}, \qquad g = \frac{G}{R + G + B} \tag{4.10}$$

For redundancy reasons, normalization of the B channel can be omitted. Since summation and integration are linear operations, it is obvious that c_j cancels out in the normalized space, if $l_j(\lambda) = l(\lambda) \forall j$, and the normalized color representation becomes invariant to shadow, intensity and camera gain variations, if linear change in the camera gain is assumed.

4.2.2 T2: Probability of Background (at Pixel Level)

Since the goal is to establish quantitative performance measures, the segmentation module should provide a probability that a pixel is background (which still encodes contrast information) rather than just a binary classification result. Since the covariance matrices that are spatially varying in the normalized space are intensity-dependent we calculate the test statistic, i.e. the Mahalanobis distance d^2, that provides a normalized distance measure of a current pixel being background.

The underlying model assumes a stationary background[3] with known mean and known covariance. Let $\hat{\mu}_{\mathbf{b}}$ be the vector of mean r_b, and mean g_b at a certain background position (mean b_b is redundant, due to normalization; $\hat{}$ indicates estimates of the true parameter), and $\hat{\mu}_{\mathbf{c}}$ be the corresponding vector of the current image pixel[4] such that

$$\hat{\mu}_{\mathbf{b}} = \begin{pmatrix} \hat{r}_b \\ \hat{g}_b \end{pmatrix} \sim N\left(\begin{pmatrix} r_b \\ g_b \end{pmatrix}, \mathbf{\Sigma}_{\hat{\mathbf{r}}_{\mathbf{b}},\hat{\mathbf{g}}_{\mathbf{b}}}\right) \tag{4.11}$$

$$\hat{\mu}_{\mathbf{c}} = \begin{pmatrix} \hat{r}_c \\ \hat{g}_c \end{pmatrix} \sim N\left(\begin{pmatrix} r_c \\ g_c \end{pmatrix}, \mathbf{\Sigma}_{\hat{\mathbf{r}}_{\mathbf{c}},\hat{\mathbf{g}}_{\mathbf{c}}}\right) \tag{4.12}$$

Subscript c indicates current values subscript b corresponds to background values.
For each pixel the metric d^2 between its background and current value is defined as:

$$\hat{d^2} = (\hat{\mu}_{\mathbf{b}} - \hat{\mu}_{\mathbf{c}})^T \left(\mathbf{\Sigma}_{\hat{\mathbf{r}}_{\mathbf{c}},\hat{\mathbf{g}}_{\mathbf{c}}} + \mathbf{\Sigma}_{\hat{\mathbf{r}}_{\mathbf{b}},\hat{\mathbf{g}}_{\mathbf{b}}}\right)^{-1} (\hat{\mu}_{\mathbf{b}} - \hat{\mu}_{\mathbf{c}}) \tag{4.13}$$

Since the background statistics are assumed to be stationary, the following it is assumption holds if the current pixel is a background pixel: $\mathbf{\Sigma}_{\hat{\mathbf{r}}_{\mathbf{c}},\hat{\mathbf{g}}_{\mathbf{c}}} = \mathbf{\Sigma}_{\hat{\mathbf{r}}_{\mathbf{b}},\hat{\mathbf{g}}_{\mathbf{b}}}$

Under this assumption, equation (4.13) turns into the following, where d^2 corresponds to the probability, that $\hat{\mu}_{\mathbf{c}}$ is background pixel:

$$\hat{d^2} = (\hat{\mu}_{\mathbf{b}} - \hat{\mu}_{\mathbf{c}})^T \left(2\mathbf{\Sigma}_{\hat{\mathbf{r}}_{\mathbf{b}},\hat{\mathbf{g}}_{\mathbf{b}}}\right)^{-1} (\hat{\mu}_{\mathbf{b}} - \hat{\mu}_{\mathbf{c}}) \tag{4.14}$$

4.2.3 T3: Indexing for Hypothesis Generation

To address real-time computational requirements of the application it is crucial to identify sectors in the image that potentially contain people of interest with probabilities of false alarm α_f, and miss-detection α_m.

To perform this indexing step in a computational efficient manner an index functions $\psi_1()$ is defined. Essentially, $\psi_1()$ is a projection operation. Let $\hat{d^2}(r, \theta)$ denote the change detection measure $\hat{d^2}$ at image position (r, θ) in polar coordinates with coordinate system origin at the omni-image center (x_c, y_c) (in Cartesian coordinates). Please note, that ' (prime) denotes the number of projected pixels corresponding to the true length of projection. The relationship between the number indicated by ' (prime) and the true length is a function of θ: e.g a distance of true length l projects onto $l' = \text{floor}(l\cos(\theta) + 0.5)$ pixels.

$\psi_1(\theta)$ is chosen to be the projection along a given radial line in direction θ to obtain $\hat{M}_\theta$, the test statistic that is used to identify changes along direction θ. This test statistic

[3] In Chapter 5 it will be shown how to relax these constraints while re-using the remaining modules.

[4] Indices for Cartesian or polar coordinates are omitted for simplification reasons.

is justified by the fact that the object projection is approximated by a line-set (which itself can be approximated as an ellipse) whose major axis passes through the omni-image center. The line-set's length distribution is a function of the radial foot position coordinates of the person in the omni-image.

For the following transformation, people are modelled as a cylinder and assumed to standing upright. Given the omni-camera projection model, any line orthogonal to the ground plane and parallel to the optical axis z is projected along radial lines through the omni camera center in the omni-image at (x_c, y_c) (in Cartesian coordinates).

Given the geometric relations induced by the omni-camera model (see equations (4.6), and (4.7))

$$k' = r'_h - r'_f = r'_m \left(\frac{H_p}{R_p} + \sqrt{\left(\frac{H_o - H_p}{R_p}\right)^2 + 1} - \sqrt{\left(\frac{H_o}{R_p}\right)^2 + 1} \right) \tag{4.15}$$

$$\lim_{R_p \to 0} k(R_p) = \lim_{R_p \to \infty} k(R_p) = 0.$$

This can easily be verified as follows: For $R_p \to 0$ the addend 1 is negligible in both expressions such that the remaining terms cancel out. For $R_p \to \infty$ every term with R_p in the denominator becomes zero; the remaining terms cancel out as well. For visualization please see Figure 4.9, center.

To increase discrimination power, we operate on an interleaved set of subsections (rings), where for the ith ring the inner and outer radii are $r'_{\mathrm{in},i}$, and $r'_{out,i}$, respectively. We define $r'_{in,i}$, and $r'_{out,i}$ as functions of person height H_p and the omni-camera geometry. Please see Figure 4.7 for illustration. One sub-sector combines 2 full projection-lengths of a person to ensure, that a projected person is fully contained in one subsection (maybe partially in another as well). The number of projected person-pixels for a person of height H_p at position $r'_{in,i}$ and $r'_{out,i}$ are $k'_{\mathrm{in},i}$ respectively $k'_{\mathrm{out},i}$, such that

$$\begin{array}{lcll} k'_{\mathrm{in},i} & = & r'_{\mathrm{out},i-1} - r'_{\mathrm{in},i} & \\ k'_{\mathrm{out},i} & = & r'_{\mathrm{out},i} \quad - r'_{\mathrm{out},i-1} & \text{with} \\ r'_{\mathrm{out},i-1} & = & r'_{\mathrm{in},i+1} & \\ k'_{\mathrm{out},i} & = & k'_{\mathrm{in},i+1} & \end{array} \tag{4.16}$$

We initialize $r'_{\mathrm{in},0}$ according to the region under investigation, while $k'_{\mathrm{in},0}$ and $k'_{\mathrm{out},0}$ can easily be calculated as follows: In equation (4.15) R_p is replaced by $H_o 2 \frac{r'_m r'_{\mathrm{in},0}}{r'^2_m - r'^2_{\mathrm{in},0}}$ such that

$$k'_{\mathrm{in},i} = r'_m \left(\frac{H_p}{H_o 2 \frac{r'_m r'_{\mathrm{in},0}}{r'^2_m - r'^2_{\mathrm{in},0}}} + \sqrt{\left(\frac{H_o - H_p}{H_o 2 \frac{r'_m r'_{\mathrm{in},0}}{r'^2_m - r'^2_{\mathrm{in},0}}} \right)^2 + 1} - \sqrt{\left(\frac{H_o}{H_o 2 \frac{r'_m r'_{\mathrm{in},0}}{r'^2_m - r'^2_{\mathrm{in},0}}} \right)^2 + 1} \right) \tag{4.17}$$

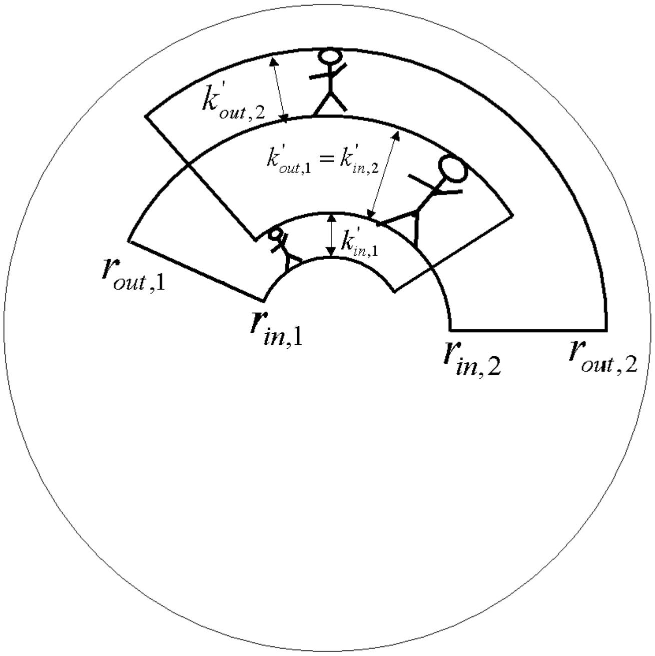

Figure 4.7: Interleaving sub-sectors in omnidirectional view: Width is a function of inner radius. Width is chosen to ensure a person of height H_p to be entirely projected into at least one sub-sector. Please note, that the persons depicted are of same height in the real world, only the projection model projects them onto different length as a function of location (see equation (4.17) and Figure 4.9.

With $r'_{\text{in},1} = r'_{\text{in},0} + k'_{\text{in},0}$ and $k'_{\text{out},i} = k'_{\text{in},i+1}$ all ring radii can be calculated.

We separately calculate for every sub-sector i a profile ${}^i\hat{M}_\theta$, which will later be our test statistic. Let $L_\theta^{x_c,y_c} = \{(x,y)|(x_c - x)\sin\theta - (y_c - y)\cos\theta = 0\}$ be a radial line through (x_c, y_c), parameterized by angle θ, then

$$ {}^i\hat{M}_\theta = \sum_{r=r'_{\text{in},i}}^{r'_{\text{out},i}} \hat{d}^2(r', \theta) \tag{4.18}$$

denotes an accumulative measure of d^2-values along the radial line in direction θ between radius $r_{\text{in},i}$ and $r_{\text{out},i+1}$, which border the sub-sector (ring). The number of values added in the ith sub-sector corresponds to $q'_i = r'_{\text{out},i+1} - r_{\text{in},i}$.

Thresholding ${}^i\hat{M}_\theta$ provides sectors of interest, where significant change is detected. The sector of significant change is bounded by ${}^i\theta_l$ and ${}^i\theta_r$, such that values ${}^i\hat{M}_\theta$ with $\theta \in [{}^i\theta_l ... {}^i\theta_r]$ define an interval of significant change in the angle space. The thresholds can be set automatically, based on a statistical analysis of the data and on the user defined requirements as miss-detection and false alarm. How to adaptively set thresholds and do the actual thrsholding will be explained in detail in the analysis section 4.3.3 that corresponds to this transform. It is obvious that the analysis is interwoven with the algorithm design at this point.

For each sub-sector we generate a binary profile $p_b^i(\theta)$.

$$p_b^i(\theta) \begin{cases} = 1, & \text{if} \quad {}^i\theta_l < \theta <^i \theta_r, \\ = 0, & \text{otherwise.} \end{cases} \tag{4.19}$$

The profile values for all subsectors at the same angular location are added and build a combined profile $p_b(\theta)$ for that

$$p_b(\theta) = \sum_i p_b^i(\theta) \tag{4.20}$$

An object is hypothesized, and bordering angles for the region of significant change are determined as follows:

$$\theta_l = \theta_i | p_b(\theta_{i-1}) = 0 \wedge p_b(\theta_i) > 0 \tag{4.21}$$

$$\theta_r = \theta_i | p_b(\theta_i) > 0 \wedge p_b(\theta_{i+1}) = 0 \tag{4.22}$$

While this transformation dealt with change detection and ultimately generated the region of significant change for each sub-sector, we will later use a similar accumulative measure $\hat{M}_\theta$ for the foot position estimation. Since the underlying geometry is the same, we introduce this measure already at this point and define

$$\hat{M}_\theta = \sum_{r=r'_{\text{in}}}^{r'_{\text{out}}} \hat{d}^2(r', \theta) \tag{4.23}$$

For ${}^{i}\hat{M}_{\theta}$ we summed r'_i values along direction θ in the ith sub-sector (ring), while for calculating $\hat{M}_{\theta}$ we sum across the entire region of interest, and define $q' = r'_{\text{out}} - r'_{\text{in}}$ as the number of values summed between the interval borders for the region of significant change, r'_{in} and r'_{out}.

4.2.4 T4: Feature Generation

The following step is used to generate features for the location estimation. Therefore, we apply a projection operation orthogonal to the one explained in 4.2.3 followed by analyzing and evaluating this profile.

Let $L^{r,\theta}_{\theta^{\top}}$ be a radial line through point $(r\cos\theta, r\sin\theta)$, parameterized by angle $\theta^{\top}$:

$$L^{r,\theta}_{\theta^{\top}} = \{(x,y)|(r\cos\theta - x)\sin\theta^{\top} - (r\sin\theta - y)\cos\theta^{\top} = 0\} \tag{4.24}$$

Similar to transformation T3, an accumulative measure $\hat{M}^{\top}_{r,\theta}$ is calculated by summation of the values $\hat{d}^2$ along line $L^{r,\theta}_{\theta^{\top}}$ (which is orthogonal to line L^{x_c,y_c}_{θ} in T3):

$$\hat{M}^{\top}_{r,\theta} = \sum_{(x,y)} \hat{d}^2(x,y) \forall (x,y)|(x_c - x)\sin\theta^{\top} - (y_c - y)\cos\theta^{\top} = r$$

$$\text{with } r \in [r'_{\text{in}}, r'_{\text{out}}], \quad \theta^{\top} = \theta + \frac{\pi}{2}$$

We threshold $\hat{M}^{\top}_{r,\theta}$ similarly as proposed in section 4.2.3. The remaining region of significant change can be approximated by an ellipse, of which the major axis is oriented with angle θ and the minor axis with $\theta^{\top}$. The prior distribution of the object's two main-axis length is a function of the person's location R_p in world coordinates, or equivalently, a function of the corresponding pixel location r_p.

4.2.5 T5: Person Foot Location Estimation in 2D (image coordinates)

We have derived the distributions of the $\hat{d}^2$ image measurements, and have narrowed our hypotheses for people location and attributes. The next step is to perform the Bayesian estimation of person locations and attributes. This step uses the likelihood models $L(\hat{d}^2|background)$ and $L(\hat{d}^2|object)$ along with 2D prior models for person attributes induced by 3D object priors $P(H_p), P(R_p) and P(\theta)$. In our current application, we make use of the fact that the probability of occlusion by persons is small to assert that the probability of a sector containing multiple people is rather small.

The center angle θ_f of a given sector would in this instance give us the estimate of the major axis of the ellipse corresponding to the person. It is then sufficient to estimate the foot location of person along the radial line corresponding to θ_f. The center angle θ_f of the sector defines the estimate for the **angular component** of the foot position:

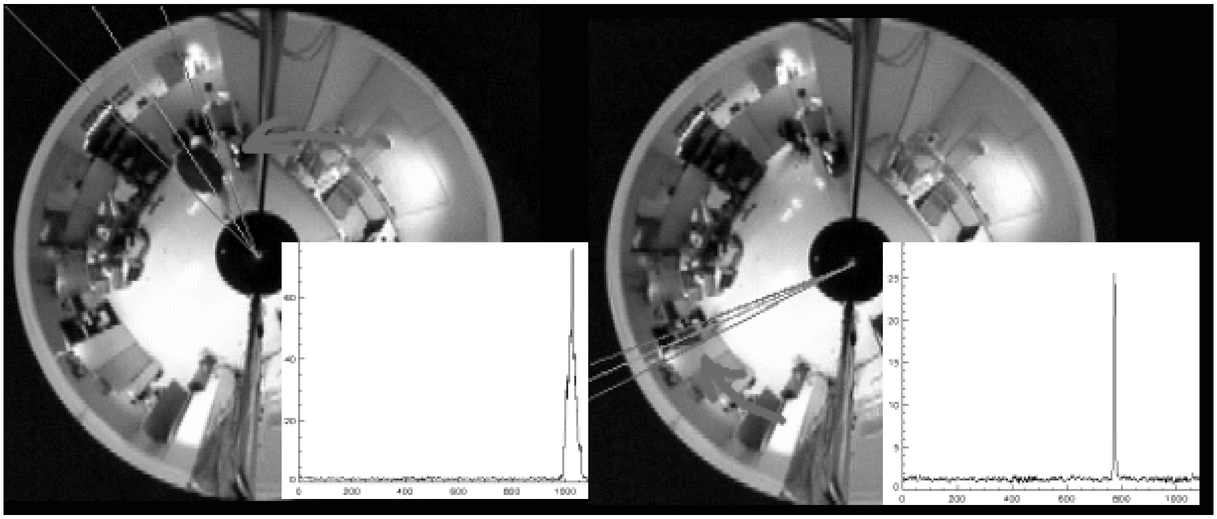

Figure 4.8: Area of significant change (Left and right lines correspond to θ_l and θ_r; Center line denotes the angular position $\hat{\theta}_f$.) Inserts show corresponding radial profile M_θ.

$$\theta_f = \frac{\theta_l + \theta_r}{2} \tag{4.25}$$

with θ_l and θ_r denoting the border angles of the hypothesized sector. See also Figure 4.8.

Given the line $L_\theta^{x_c,y_c}$ it is necessary to estimate the foot position of the person along this radial line. Therefore, the person is assumed to stand upright such that it is projected along radial lines in the omni-image. To find this estimate and variance of the **radial foot position** r_f we choose the best hypothesis for the foot position that minimizes the Bayes error. The prior distribution of person heights H_p is assumed to be Gaussian. We actually need to estimate the person's height on the projection by using the Bayesian formulation. However, we rather use the assumption that the variance of the height is small, and just fix H_p as constants. Errors introduced at this point will be analyzed in the analysis process (see 4.3.5) and accounted for in the final zoom setting. The geometric transformations are still taken into account to identify 2D projection lengths as a function of radial position along the radial line. Let $P(h_i|m)$ denote the posterior probability to be maximized, where h_i denotes the ith out of multiple foot position hypotheses footnoteEach hypothesis h_i maps directly to a potential radial foot position r and m the measurements $\hat{M}_{r,\theta_f}^\top$, that are statistically independent; hyper-script b or o denotes *background* respectively *object*. The radial foot position r_f in the image space can then be estimated as follows:

$$\begin{aligned}
&P(h_i|m) \\
&= P(h_i^b|m^b)P(h_i^o|m^o) = P(h_i^b|m^b)\left(1 - P(\bar{h_i^o}|m^o)\right) \\
&= \frac{p(m^b|h_i^b)P(h_i^b)}{p(m^b)} \frac{p(m^o) - p(m^o|\bar{h_i^o})P(\bar{h_i^o})}{p(m^o)}
\end{aligned} \tag{4.26}$$

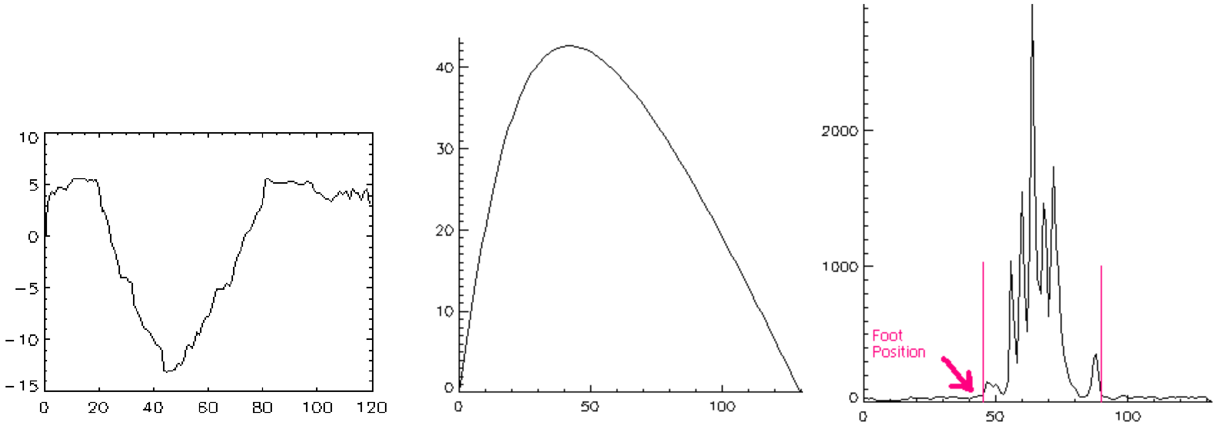

Figure 4.9: Left: Bayes error as function of hypothesized foot position r'_f, here: most probable foot position at position $r'_f = 47$. Center: Projected person length k as function of r_f. Note: $k(r'_f = 47) = 43$. Right: Profile $\bar{d}^2(r', \theta_f)$: by minimizing Bayes error, responses in interval [47...47+43=90] are classified as object responses.

where p denotes the density function. $P(h_i|m)$ becomes maximal for maximal $p(m^b|h_i^b)$ and minimal $p(m^o|\bar{h}_i^o)$, so that

$$r_f = \underset{r'_f}{\operatorname{argmin}} \log\left(\frac{p(m^o|\bar{h}_i^o)}{p(m^b|h_i^b)}\right) \tag{4.27}$$

Approximating $\hat{M}_{r,\theta_f}^\top$ by a Gaussian with same variance for background and object responses at this step, we can approximate

$$\log\left(\frac{p(m^o|\bar{h}_i^o)}{p(m^b|h_i^b)}\right) = \left(\sum_{r=0}^{r'_f-1} \hat{M}_{r,\theta_f}^\top + \sum_{r=r_h(r'_f)}^{r_m} \hat{M}_{r,\theta_f}^\top - \sum_{r=r_f}^{r_h(r'_f)-1} \hat{M}_{r,\theta_f}^\top\right) \tag{4.28}$$

Bayes error as a function of the radial index parameter please see Figure 4.9. It also illustrates the typical profile for the projected person-height as a function of index.

4.2.6 T6: Location Estimation in 3D (real-world coordinates)

Given the geometric projection model of the OmniCam [84] we can transform the foot position (θ_f, r_f) in the image space into 3D world coordinates.

Knowing the person's height H_p, the radial distance in 3D R_p between a person and the omni-camera can be calculated as follows (see Figure 4.2):

$$R_p = 2\frac{r_m r_f}{r_m^2 - r_p^2} H_o \tag{4.29}$$

The angle θ is invariant to the projection model.

4.2.7 T7: Foveal Camera Control Parameter Estimation (Pan, Tilt)

Once the estimate for the foot position in 3D is known, basic trigonometric transforms provide the solution for how to set the foveal camera control parameters. From Figure 4.3 it is easy to see that

$$\begin{aligned} \tan(\alpha) &= \frac{H_p - R_h - H_f}{D_p} && (4.30) \\ \sin(\beta) &= \frac{R_p}{D_p}\sin(\vartheta) \quad \text{with} && (4.31) \\ D_p &= \sqrt{D_c^2 + R_p^2 - 2D_cR_p\cos(\vartheta)} && (4.32) \end{aligned}$$

4.2.8 T8: Zoom Parameter — The Final Estimate

At this point, it is most evident how the analysis needs to go along with the algorithm and module design. We remember that for optimal zooming it is not enough to know the best estimate of a person in the scene. We do have to know as well *how* good this best estimate is. Otherwise we might end up zooming in to the maximal extent but still missing the center of the face by a bit, which might be large enough to not have the entire face in the foveal view. By knowing the uncertainty, we rather zoom in more conservatively to ensure the face being in the foveal frame in α_z% of the cases. To zoom in to the maximal extent given the just mentioned requirement, we assume that we know the uncertainties in the estimates already. How to derive these on-line will be shown by in the system analysis section 4.3.7.

Given the uncertainties in the estimates, we can derive the horizontal and vertical field of view for the foveal camera, $2\gamma_h$ respectively $2\gamma_v$, which map directly to the zoom parameter z. Zoom parameter Z is defined as

$$z = min(T_Z^h(2\gamma_h), T_Z^v(2\gamma_v)) \tag{4.33}$$

where the transformation $T_Z^h(2\gamma_h)$ and $T_Z^v(2\gamma_v)$ between the horizontal respectively vertical field of view and the zoom factor Z is foveal camera specific (see appendix A.4 for details on the camera used) .

Figure 4.10 shows the geometric relationship for the horizontal and vertical case. Following equations provide half the vertical, and horizontal field of view, γ_v, respectively γ_h .

$$\begin{aligned} \hat{\gamma}_h &= \arctan\left(\frac{\hat{R}_h + f_h\sigma_{\sin\hat{\beta}}\sqrt{\hat{R}_h^2 + \hat{D}_p'^2}}{\hat{D}_p'}\right) && (4.34) \\ \hat{\gamma}_v &= \arctan\left(\frac{\hat{R}_h + f_v\sigma_{\tan\hat{\alpha}}\hat{D}_p'}{\hat{D}_p'}\right) \quad \text{with} && (4.35) \end{aligned}$$

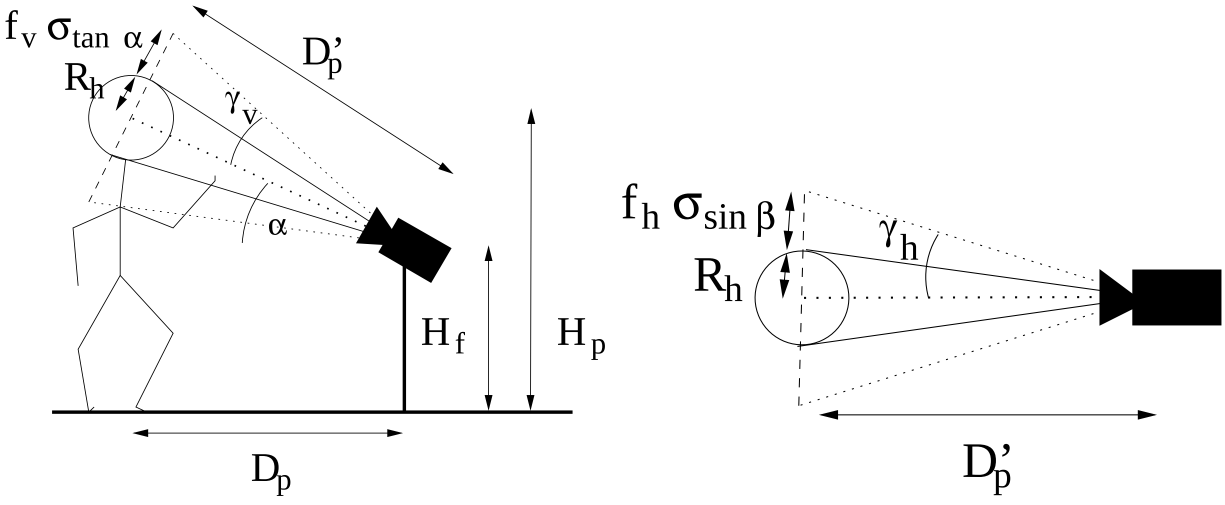

Figure 4.10: Geometric relations for vertical (left, γ_v), and horizontal field of view (right, γ_h) calculation. View from the side respectively from atop.

$$\hat{D}'_p = \frac{\hat{D}_p}{\cos\alpha} \tag{4.36}$$

where factor $f = f_h = f_v$ solves for $\int_0^f N(0,1)d\xi = \frac{\alpha_Z}{2}\%$ given user specified confidence percentile α_z that the head is display in the foveal frame.

For zoom factor Z, one can now calculate the corresponding pair of half the actual horizontal and vertical field of view, $\tilde{\gamma_h}$ and $\tilde{\gamma_v}$ by applying inverse transformations $(T_Z^h)^{-1}(Z)$ and $(T_Z^v)^{-1}(Z)$:

$$\tilde{2}\gamma_h = (T_Z^h)^{-1}(Z) \tag{4.37}$$

$$\tilde{2}\gamma_v = (T_Z^v)^{-1}(Z) \qquad \text{with} \tag{4.38}$$

$$\sin(\tilde{\gamma_h}) = r_a \sin(\tilde{\gamma_v}). \tag{4.39}$$

Due to a fixed aspect ratio of the foveal camera r_a=width:height, the ratio of the tan(.) of vertical and horizontal field of view is fixed as well. Given the radius R_h of the person's head and it's distance D'_p from the foveal camera, we can estimate the percentage of pixels in the foveal frame being covered by a face. Here, r_v describes the ratio in vertical direction, r_h in horizontal direction, and r_{2D} the ratio for the entire pixels of the foveal frame:

$$\hat{r}_h = \frac{\hat{R}_h}{\sin(\tilde{\gamma_h})\hat{D}'_p} \tag{4.40}$$

$$\hat{r}_v = \frac{\hat{R}_h}{\sin(\tilde{\gamma_v})\hat{D}'_p} \qquad \text{with} \tag{4.41}$$

$$\hat{r}_{2D} = \hat{r}_h \hat{r}_v \tag{4.42}$$

By knowing the best estimate, and the uncertainties in the foot position as well as the zoom factor one can easily calculate the range of the horizontal and vertical number of pixels, which are dominant in the foveal frame. That knowledge can then help to automatically adapt parameters in higher level image processing algorithms, e.g. kernel sizes in face recognition engines.

4.2.9 Tracker

To label location estimates and establish correspondences over time, we use a simple nearest neighborhood tracker, which compares location predictions $(\tilde{x}_i, \tilde{y}_i)$ for time t with the results $(\hat{x}_j, \hat{y}_j)$ from our location estimation routines at time t. $(\hat{x}_j, \hat{y}_j)$ denotes the estimates of the real-world Cartesian coordinates (x, y), which can be derived from the estimates (R_p, θ) in polar coordinates as follows:

$$\begin{aligned} x &= R_p \cos\theta \\ y &= R_p \sin\theta \end{aligned} \tag{4.43}$$

The predictions $(\tilde{x}_i, \tilde{y}_i)$ can be calculated by a Kalman filter at time $t-1$. In our application we use a constant velocity model that allows for white noise acceleration. Since evaluation and discussion of the Kalman filter is not part of this work, please refer to [8], page 82 pp. for further details. However, the following term describes which label L_j we assign for the jth person at estimated location $(\hat{x}_j, \hat{y}_j)$ and time t (for simplicity reasons the time index t is omitted):

$$L_j = \underset{i}{\text{argmin}} \{ \left(\begin{pmatrix} \tilde{x}_i \\ \tilde{y}_i \end{pmatrix} - \begin{pmatrix} \hat{x}_j \\ \hat{y}_j \end{pmatrix} \right) \left({}^{K}\Sigma_{\tilde{x},\tilde{y},i} + \Sigma_{\hat{x},\hat{y},j} \right)^{-1} \left(\begin{pmatrix} \tilde{x}_i \\ \tilde{y}_i \end{pmatrix} - \begin{pmatrix} \hat{x}_i \\ \hat{y}_i \end{pmatrix} \right) \} \tag{4.44}$$

where ${}^{K}\Sigma_{\tilde{x},\tilde{y},i}$ indicates the covariance matrix from the Kalman filter, that indicates for the ith person the uncertainty in the prediction. $\Sigma_{\hat{x},\hat{y},j}$ indicates for the jth person the covariance matrix that describes the uncertainty in the location estimate in real world x-y-coordinates.

It is worthwhile noting that even though we do not provide an analysis of the tracker in this work, the Kalman Filter needs to be initialize to work properly. For the initialization it is necessary to provide the uncertainty in the estimates R_p and θ. As we outline later, our online analysis provides estimates for $(\sigma^2_{R_p}, \sigma^2_{\theta})$. That shows how the analysis actually influences the design. With equations (A.9),(A.8), and (A.5) we use following expressions to initialize the Kalman filter:

$$\sigma^2_{\hat{x}} = \sigma^2_{\hat{R}\cos\hat{\theta}} = \sigma^2_{\theta} \sin^2\theta \left(R^2 + \sigma^2_R \right) + \cos^2\theta \sigma^2_R \tag{4.45}$$

$$\sigma^2_{\hat{y}} = \sigma^2_{\hat{R}\sin\hat{\theta}} = \sigma^2_{\theta} \cos^2\theta \left(R^2 + \sigma^2_R \right) + \sin^2\theta \sigma^2_R \tag{4.46}$$

$$\begin{aligned}
\sigma_{\hat{x},\hat{y}} &= \sigma_{\hat{R}\cos\hat{\theta},\hat{R}\sin\hat{\theta}} \\
&= E\{\Big(R^2\sin\theta\cos\theta - (R+\eta_R)^2\sin(\theta+\eta_\theta)\cos(\theta+\eta_\theta)\Big)\} \\
&\approx E\{\Big(R^2\sin\theta\cos\theta - (\sin\theta+\cos\theta\eta_\theta)\, left(\cos\theta-\sin\theta\eta_\theta\Big)\,(R+\eta_R)^2\} \\
&\approx \sin\theta\cos\theta\left(R^2\sigma_\theta^2 + \sigma_R^2\sigma_R^2 - \sigma_R^2\right) \qquad (4.47)
\end{aligned}$$

4.3 System Analysis

This section illustrates in detail how the system analysis methodology described in sections 2.1, and 3.1 is applied to this application. Figure 1.1 (right block) illustrates the design phase being followed by the analysis. As described above, and seen in Table 4.2, for analysis purposes the system is thought of as a sequence of transformations T1 through T8. Table 4.2 summarizes the abstract definition of each transformation step, while Table 4.3 illustrates how the statistical analysis proceeds for each transformation step. Please remember, that the output of a given transformation is the input to the successive transformation (except in the case of T4 and T5 that correspond to the situation where T5 is not operating on the output of T4). Given that the architecture and the transformations are fixed, tools outlined in the methodology section were used to derive the theoretical distributions for the perturbation models at the output of each stage, given the input perturbation model from previous stage, ideal model parameters and the thresholds employed (if any). Table 4.3 illustrates the type of analysis method used in each step. Please note that in most cases covariance propagation works well as long as the perturbation magnitude is small compared to the signal magnitude (steps: T1, T6, T7, T8 use linear covariance propagation). Steps T2, T3 and T4, are less involved so that preferred distribution propagation and/or distribution-approximation techniques could be applied. This is standard random variable algebra from statistics literature (Please see [89]). Step T5 corresponds to a non-linear estimation step that requires numerical computation of the uncertainty of the estimated value. This is done by parametric bootstrap [31]. Table 4.4 provides an overview of the prior distributions that influence each transformation and the thresholds related to the global criterion functions (e.g. probability of missing a hypothesis, probability of false hypothesis, and the probability that the zoomed up image contains the face). In the following sections a module-wise analysis is carried out.

Table 4.2: Abstract Model for Algorithm

Estimation	Trafo	Mapping
Illumination Invariance	T1	$\begin{pmatrix} \hat{R} \\ \hat{G} \\ \hat{B} \end{pmatrix} \longrightarrow \begin{pmatrix} \hat{r} \\ \hat{g} \end{pmatrix}$
Probability of Background	T2	$\left(\begin{pmatrix} \hat{r} \\ \hat{g} \end{pmatrix}_c \times \begin{pmatrix} \hat{r} \\ \hat{g} \end{pmatrix}_b, \mathbf{\Sigma}_{\hat{\mathbf{r}},\hat{\mathbf{g}}} \right) \longrightarrow \hat{d}^2$
Indexing	T3	$\hat{d}^2(r', \theta) \longrightarrow^i \hat{M}_\theta \longrightarrow (\hat{\theta}_l, \hat{\theta}_r)$ with ${}^i\hat{M}_\theta = \sum'_r \hat{d}^2(r', \theta), r' \in [r'_{\text{in},i}, r'_{\text{out},i}]$
Feature Estimation	T4	$\hat{d}^2(x,y) \times (\theta_l, \theta_r) \longrightarrow \hat{M}^\top_{r,\theta}$ with $\hat{M}^\top_{r,\theta} = \sum_{(x,y)} \hat{d}^2(x,y)$ $\forall (x,y) \vert (x_c - x)\sin(\theta + \frac{\pi}{2}) - (y_c - y)\cos(\theta + \frac{\pi}{/2}) = r,$ with $r' \in [r'_{\text{in}}, r'_{\text{out}}], \quad \theta \in \{\theta_l, \theta_r\}$
Location estimation 2D	T5	$\begin{pmatrix} \hat{M}_\theta \\ \hat{M}^\top_{r,\theta} \end{pmatrix} \longrightarrow \begin{pmatrix} \hat{r}_p \\ \hat{\vartheta} \end{pmatrix}$
Location estimation 3D	T6	$\hat{r}_p \longrightarrow \hat{R}_p$
Pan/Tilt Estimation	T7	$\begin{pmatrix} \hat{R}_p \\ \hat{\vartheta} \end{pmatrix} \longrightarrow \begin{pmatrix} \tan(\hat{\alpha}) \\ \sin\left(\hat{\beta}\right) \end{pmatrix}$
Zoom Setting	T8	$\begin{pmatrix} \tan(\hat{\alpha}) \\ \sin(\hat{\beta}) \end{pmatrix} \longrightarrow \text{zoom } Z$

4.3.1 Analysis of T1: Illumination Invariant Measure Estimation

The ideal sensor output for the three color-channels is the ideal input for the illumination invariant transformation T1:

$$\begin{pmatrix} \hat{R} \\ \hat{G} \\ \hat{B} \end{pmatrix} \tag{4.48}$$

The perturbation module chosen assumes Gaussian noise for all bands (η_R, η_G, η_B), while cross-correlation between bands is assumed to be zero:

$$\begin{pmatrix} \eta_R \\ \eta_G \\ \eta_B \end{pmatrix} \sim N \left(\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \mathbf{diag} \begin{pmatrix} \sigma^2_{\hat{R}} \\ \sigma^2_{\hat{G}} \\ \sigma^2_{\hat{B}} \end{pmatrix} \right) \tag{4.49}$$

such that the true input can be written as:

$$\begin{pmatrix} \hat{R} \\ \hat{G} \\ \hat{B} \end{pmatrix} \sim N \left(\begin{pmatrix} R \\ G \\ B \end{pmatrix}, \mathbf{diag} \begin{pmatrix} \sigma^2_{\hat{R}} \\ \sigma^2_{\hat{G}} \\ \sigma^2_{\hat{B}} \end{pmatrix} \right) \tag{4.50}$$

Table 4.3: Statistical Analysis

Trafo	I/P distribution	O/P distribution	Type of Propagation
T1	$N\left(\begin{pmatrix} R \\ G \\ B \end{pmatrix}, \mathbf{diag}\begin{pmatrix} \sigma^2_{\hat{R}} \\ \sigma^2_{\hat{G}} \\ \sigma^2_{\hat{B}} \end{pmatrix}\right)$	$N\left(\begin{pmatrix} r \\ g \end{pmatrix}, \mathbf{\Sigma}_{\hat{\mathbf{r}},\hat{\mathbf{g}}}\right)$	covariance propagation
T2	$\mathrm{N}\left(\begin{pmatrix} r \\ g \end{pmatrix}, \mathbf{\Sigma}_{\hat{\mathbf{r}},\hat{\mathbf{g}}}\right)$	Background pixel: $\chi^2_2(0)$ Object pixel: $\chi^2_2(c_\theta), c_\theta \neq 0$	distribution propagation
T3	$\chi^2_2(c_\theta), c \in [0 \ldots \infty]$	$(q' - k')\chi^2_{2(q'-k')}(0) + k\chi^2_{2k'}(c_\theta)$	distribution propagation
T4	$\chi^2_2(c_\theta), c_\theta \in [0 \ldots \infty]$	Background pixel: $\chi^2_{2s'}(0)$ Object pixel: $\chi^2_{2s'}(c_\theta), c_\theta \neq 0$	distribution propagation
T5	$n_b\chi^2_{2n_b)}(0) + n_o\chi^2_{2n_o}(c^\top_{r,\theta}),$ $c^\top_{r,\theta} \in [0 \ldots \infty]$	$N\left(\begin{pmatrix} r_p \\ \vartheta \end{pmatrix}, \begin{pmatrix} \sigma^2_{\hat{r}_p} \\ \sigma^2_{\hat{\vartheta}} \end{pmatrix}\right)$	bootstrap
T6	$N\left(r_p, \sigma^2_{\hat{r}_p}\right)$	$N\left(R_p, \sigma^2_{\hat{R}_p}\right)$	covariance propagation
T7	$N\left(\begin{pmatrix} R_p \\ \vartheta \end{pmatrix}, \begin{pmatrix} \sigma^2_{\hat{R}_p} \\ \sigma^2_{\hat{\vartheta}} \end{pmatrix}\right)$	$N\left(\begin{pmatrix} \tan(\alpha) \\ \sin(\beta) \end{pmatrix}, \begin{pmatrix} \sigma^2_{\tan(\hat{\alpha})} \\ \sigma^2_{\sin(\hat{\beta})} \end{pmatrix}\right)$	covariance propagation
T8	$N\left(\begin{pmatrix} \tan(\alpha) \\ \sin(\beta) \end{pmatrix}, \begin{pmatrix} \sigma^2_{\tan(\hat{\alpha})} \\ \sigma^2_{\sin(\hat{\beta})} \end{pmatrix}\right)$	$N\left(Z, \sigma^2_Z\right)$	covariance propagation

Table 4.4: Criterion Functions and Priors Influencing the Choice of Transforms

Trafo	Threshold/Criterion Fct.	Priors
T1	n/a	P(Illumination)
T2	n/a	n/a

T3	P(missing hypothesis),P(false hypothesis)
	–on object level–
		P(Projection. Geometry)
		P(Object Height), P(Object Location)

T4	n/a	P(Object Height)
T5	n/a	P(Object Radius)
T6	n/a	P(Geometry)
T7	n/a	P(Object Pose)
T8	P(head in foveal frame)	P(Object Head Size)

The ideal output after the normalization transformation T1 would be

$$r = \frac{R}{R+G+B}, \qquad g = \frac{G}{R+G+B} \tag{4.51}$$

At this point, we apply linear covariance propagation techniques to estimate the uncertainty in the output as a function of ideal input parameters and the input noise model parameters. For the following estimation of variance and covariance entries for the covariance matrix $\mathbf{\Sigma}_{\hat{\mathbf{r}},\hat{\mathbf{g}}}$ we apply equations (A.17) and (A.20).

$$\begin{pmatrix} \eta_r \\ \eta_g \end{pmatrix} N\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \mathbf{\Sigma}_{\hat{\mathbf{r}},\hat{\mathbf{g}}} \right) \qquad \text{with} \tag{4.52}$$

$$\begin{aligned} \mathbf{\Sigma}_{\hat{\mathbf{r}},\hat{\mathbf{g}}} &= \frac{\sigma_S^2}{S^2} \begin{pmatrix} \frac{\sigma_R^2}{\sigma_S^2}(1-\frac{2R}{S}) + \frac{R^2}{S^2} & -\frac{\sigma_G^2 R + \sigma_R^2 G}{\sigma_S^2 S} + \frac{RG}{S^2} \\ -\frac{\sigma_G^2 R + \sigma_R^2 G}{\sigma_S^2 S} + \frac{RG}{S^2} & \frac{\sigma_G^2}{\sigma_S^2}(1-\frac{2G}{S}) + \frac{G^2}{S^2} \end{pmatrix} && (4.53) \\ &\approx \frac{\sigma^2}{S^2} \begin{pmatrix} 1-\frac{2R}{S}+3\frac{R^2}{S^2} & -\frac{R+G}{S}+3\frac{RG}{S^2} \\ -\frac{R+G}{S}+3\frac{RG}{S^2} & 1-\frac{2G}{S}+3\frac{G^2}{S^2} \end{pmatrix} \quad \text{for } \sigma_{\hat{R}}^2 = \sigma_{\hat{G}}^2 = \sigma_{\hat{B}}^2 = \sigma^2 && (4.54) \end{aligned}$$

with $S = R + G + B$, and $\sigma_S^2 = \sigma_R^2 + \sigma_G^2 + \sigma_B^2$[5]. The true output in the normalized color space becomes

$$\begin{pmatrix} \hat{r} \\ \hat{g} \end{pmatrix} = \begin{pmatrix} \frac{\hat{R}}{\hat{R}+\hat{G}+\hat{B}} \\ \frac{\hat{G}}{\hat{R}+\hat{G}+\hat{B}} \end{pmatrix} \sim N\left(\begin{pmatrix} r \\ g \end{pmatrix}, \mathbf{\Sigma}_{\hat{\mathbf{r}},\hat{\mathbf{g}}} \right) \tag{4.55}$$

can be approximated as normal distributed with mean (r, g) and pixel-dependent covariance matrix $\mathbf{\Sigma}_{\hat{\mathbf{r}},\hat{\mathbf{g}}}$.

Please note, that even though the noise across the input bands could assumed to be independent, we can not assume noise in the normalized channels to be uncorrelated as well. The analysis suggests to maintain a full 2 × 2 covariance matrix.

To account for discretization noise in the sensor we lower bound the variances for each channel by $\sigma_n^2 = 0.09$. The values of $\sigma_{\hat{r}}^2$ are shown in Figure 4.11 for an entire OmniCam frame. Note, in the normalized space the covariance matrix for each pixel is different: Bright regions in the covariance image correspond to regions with high variance in the normalized image. These regions correspond to dark regions in RGB space. In saturated areas, where values appear to be stable due to limited dynamic range of the camera, they are rather uncertain in reality. To account for this effect and treat saturated values similar

[5] We assume a sufficiently large signal to noise ratios larger 3 in each band.

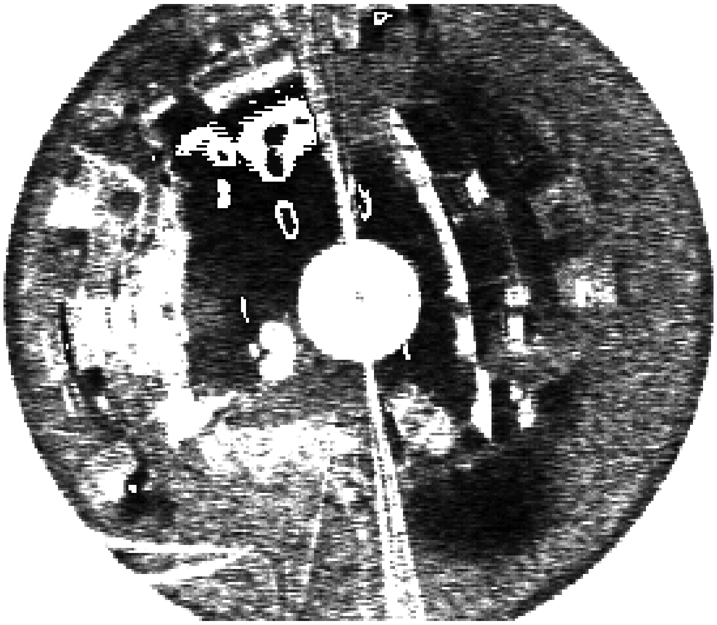

Figure 4.11: Covariance image for $\sigma^2_{\hat{r}}$. Bright regions in the normalized space denote high variance; these regions correspond to dark areas in the RGB image.

to dark values in terms of uncertainty, sensor responses $C_i > 220$[6] are "mirrored" on the center value of the dynamic range and for calculating corresponding entries in $\mathbf{\Sigma}_{\hat{\mathbf{r}},\hat{\mathbf{g}}}$ only, they are replaced by $\tilde{C}_i = 255 - C_i$.

4.3.2 Analysis of T2: Probability of Background

The transformation T1 provided an illumination-invariant pixel representation in the normalized space. Since the pixel can represent two different classes (background, indicated by subscript b, or object indicated by subscript o), each pixel value is either distributed as

$$\begin{pmatrix} \hat{r}_b \\ \hat{g}_b \end{pmatrix} \sim N\left(\begin{pmatrix} r_b \\ g_b \end{pmatrix}, \mathbf{\Sigma}_{\hat{\mathbf{r}}_\mathbf{b},\hat{\mathbf{g}}_\mathbf{b}}\right) \tag{4.56}$$

or

$$\begin{pmatrix} \hat{r}_o \\ \hat{g}_o \end{pmatrix} \sim N\left(\begin{pmatrix} r_o \\ g_o \end{pmatrix}, \mathbf{\Sigma}_{\hat{\mathbf{r}}_\mathbf{o},\hat{\mathbf{g}}_\mathbf{o}}\right) \tag{4.57}$$

Since the transform can be thought of a summation of two squared independent normal distributed random variables (in the background case with identical mean 0), the distribution of the output is well known [89]: For background pixels, $\hat{d}^2$ is approximately

[6] The CCD camera uses 8 bits per channel.

χ^2 distributed with two degrees of freedom under the hypothesis, $\hat{\mu}_{\mathbf{b}}, \hat{\mu}_c$ are normal distributed. For object pixels $\hat{d}^2$ can be approximated by a non-central χ^2 distributed with two degrees of freedom, and non-centrality parameter c_θ. $\hat{d}^2$ is exactly non-central χ^2 distributed with two degrees of freedom if the covariance for background and foreground are identically.

$$\begin{array}{ll} \textbf{Background pixel:} & \hat{d}^2 \sim \chi_2^2(0) \\ \textbf{Object pixel:} & \hat{d}^2 \sim \chi_2^2(c_\theta), \quad c_\theta \neq 0 \end{array} \tag{4.58}$$

4.3.3 Analysis of T3: Indexing for Hypothesis Generation

For the indexing the distance image $\hat{d}^2$ serves as input. It represents for each pixel a metric between background and current image. Depending on the class a pixel belongs to it is distributed as follows, given the noise introduced in the sensor and propagated through each previous model:

$$\begin{array}{ll} \textbf{Background pixel:} & \hat{d}^2 \sim \chi_2^2(0) \\ \textbf{Object pixel:} & \hat{d}^2 \sim \chi_2^2(c_\theta), c_\theta \neq 0 \end{array} \tag{4.59}$$

Since the transform T3 is essentially a summation of k' χ^2 distributed values, where k' can be derived from the geometry model as follows: Let q' be the total number of pixels along a radial line $L_\theta^{x_c,y_c}$ onto which the radius of the parabolic mirror is projected, and k' be the expected number of object pixels projected onto the same line.

For later analysis one need to know the prior probability density function of H_p, R_p, H_o, q', and c: The prior distribution for non-centrality parameter c_θ can be assumed uniformly distributed, while the priors for the other parameters can be assumed normal distributed.

Given the geometric relations induced by the omni-camera model we derived

$$k' = r'_h - r'_f = r'_m \left(\frac{H_p}{R_p} + \sqrt{\left(\frac{H_o - H_p}{R_p}\right)^2 + 1} - \sqrt{\left(\frac{H_o}{R_p}\right)^2 + 1} \right) \tag{4.60}$$

The distribution of the sum is well known. With non-centrality parameter $c_\theta > 0$ the distribution of $\hat{M}_\theta$ after summing q' values, out of which k' were object pixels[7], is as follows:

$$\begin{array}{ll} \textbf{all background pixels:} & \hat{M}_\theta = \hat{M}_\theta^b \sim \chi_{2q'}^2(0) \\ \textbf{k}' \textbf{ object pixels:} & \hat{M}_\theta = \hat{M}_\theta^o \sim (q' - k')\chi_{2(q'-k')}^2(0) + k'\chi_{2k'}^2(c_\theta) \end{array} \tag{4.61}$$

[7] remaining pixels are background pixels

Having derived these probability density functions, expressions for the probabilities of false alarm α_f, and miss-detection α_m in respect to the task of people detection follow: Due to the way intermediate results were propagated (please note, that output distributions of the previous transform is input for the following) they are direct functions of the input distributions for $\hat{d}^2(r,\theta)$, the prior distribution for the expected fraction of the pixels along a given radial line belonging to the object, and the non-centrality parameter of $\hat{d}^2(r,\theta)$ in object locations.

Applying Canny's hysteresis-thresholding technique ([18]) on $\hat{M}_\theta$, provides the sectors of significant change bounded by left and right angles θ_l respectively θ_r. The hysteresis-thresholding technique partitions the signal (here profile $\hat{M}_\theta$) into two subsets of intervals, where one subset denotes areas of significant change in the angle space. This interval $[\theta_l...\theta_r]$ can be defined as follows based on an upper and a lower threshold T_u respectively T_l: $\hat{M}_{\theta_i} < T_l \forall \theta_i \in [\theta_l...\theta_r] \wedge \hat{M}_{\theta_j} > T_u \exists \theta_j \in [\theta_l...\theta_r]$. That means, all values within the interval are larger than the lower threshold *and* at least one value is larger than the upper threshold. Obviously, following restriction applies: $T_u > T_l$.

Thresholds: Thresholds used for the hysteresis thresholding can be setup by using the scene and object priors as follows. To guarantee a false-alarm rate for false sectors of equal or less than $\alpha_f\%$ (background case) we can set the lower threshold T_l so that

$$\int_0^{T_l} p(\hat{M}_\theta^b)dM_\theta^b = \int_0^{T_l} \chi^2_{2q'}(0) = 1-\alpha_f\% \tag{4.62}$$

with p(.) denoting the probability function that describes the statistical behavior of the argument.

To guarantee a miss-detection rate of equal or less than $\alpha_m\%$, theoretically, we can similarly solve for an upper threshold T_u by evaluating the distribution in equation (4.61) for the object case:

$$\begin{array}{l} \int_0^{T_u}\int_{c_\theta}\int_{H_o}\int_{R_p}\int_{H_p} p(\hat{M}_\theta^o)p(H_p)p(R_p)p(H_o)p(c_\theta)\,dH_p\,dR_p\,dH_o\,dc\,d\xi = \alpha_m\% \\ \text{with} \quad p(\hat{M}_\theta^o) = (q'-k')\chi^2_{2(q'-k')}(0) + k\chi^2_{2k'}(c_\theta) \end{array} \tag{4.63}$$

where k' indicates the number of object-pixels along a line. Note, that M_θ^o is a function of q', k', c, and k' itself is a function of R_p, H_p, H_o, and r'_m (see equation (4.60)). Unfortunately, we cannot make any assumptions about the distribution of non-central parameter c_θ, so we have to resort to the use of a LUT $T_u(\alpha_m)$ generated by simulations instead.

4.3.4 Analysis of T4: Hypothesis Generation

The input for this transformation is the same as for the previous module:

$$\begin{array}{ll} \textbf{Background pixel:} & \hat{d}^2 \sim \chi^2_2(0) \\ \textbf{Object pixel:} & \hat{d}^2 \sim \chi^2_2(c), \quad c \neq 0 \end{array} \tag{4.64}$$

In fact, the transform itself is quite similar, too. From previous steps s' is known: It denotes the number of pixels summed along line $L_{\theta^\top}^{r,\theta}$ (see equation (4.24)) between the hypothesized bordering angles θ_l and θ_r.

$$\begin{array}{ll} \textbf{Background pixels:} & \hat{M}_{r,\theta}^\top \sim \chi_{2s'}^2(0) \\ \textbf{Object pixels:} & \hat{M}_{r,\theta}^\top \sim \chi_{2s'}^2(c_{r,\theta}^\top)), \quad c_{r,\theta}^\top \neq 0 \end{array} \tag{4.65}$$

Of course, s' is a function of person size S and its location in the scene. Given s' and $M_{r,\theta}^\top$, and under the assumption that all pixels along the line segment between θ_l and θ_r are object pixels one can numerically calculate the corresponding non-centrality parameter $c_{r,\theta}^\top$ by solving $\hat{M}_{r,\theta_f}^\top = \int_0^\infty \chi_{2s'}^2(c_{r,\theta}^\top, \xi))d\xi$ for $c_{r,\theta}^\top$.

4.3.5 Analysis of T5: Person Foot Location Estimation in 2D

Finally, we estimate the uncertainty in the foot position coordinates (θ_f, r_f). As seen above in section 4.2.5 we neither have a close form to estimate θ_f from profile $\hat{M}_\theta$, nor r_f from the input profile $\hat{M}_{r,\theta_f}^\top$. Nevertheless, our approach provides us with the corresponding *pdf*s up to the previous step in the algorithm. At this point, it is affordable to simulate the distribution of r_f, respectively θ_f and estimate $\sigma_{\hat{r}_f}^2$ and $\sigma_{\hat{\theta}_f}^2$ by parametric bootstrap. This sampling technique is feasible, since the space is small enough, and only few estimates with known distributions are involved in few operations.

To estimate the **angular position** we approximate $\hat{\theta}_f$ to be normal distributed with unknown θ_f and variance $\sigma_{\hat{\theta}_f}^2$. Once r_f and the corresponding projection length k' (see equation (4.15)) are estimated and $\hat{M}_\theta$ is calculated, the non-centrality parameter c_θ corresponding to the cumulative measure $\hat{M}_\theta$ in radial direction can numerically be estimated by solving the following integral for c_θ

$$M_\theta = \int_0^\infty (q' - k')\chi_{2(q'-k')}^2(0, \xi)d\xi + \int_0^\infty k'\chi_{2k',}^2(c_\theta, \xi)d\xi \tag{4.66}$$

To equally account for the uncertainty in the data as well as for the fact, that the head may naturally move [8] within the envelope restricted by θ_l and θ_r we set the uncertainty in $\hat{\theta}_f$ such that

$$\sigma_{\hat{\theta}_f}^2 = \max\left(\frac{|\theta_l - \theta_r|}{4}, \sigma_{\theta_f,bootstrap}^2\right) \tag{4.67}$$

This reflects the fact that the zoom factor is not only a function of the uncertainty in the estimates. It should rather be a function of both, the error introduced by deviating from the cylindrical model when a person moves her head (reflected by the first term) *and*

[8] no matter how certain the foot position estimation is

of the uncertainty in the foot position estimation (reflected by the second term). Given the proportions of a human being and given the restricted range of head motion relative to the main axis of the person, it is reasonable to assume that the head is in 95% of the cases within the envelope defined by $[\theta_l ... \theta_r]$.

Experiments show, that the **radial position** $\hat{r}_f$ can be approximated by a normal distribution with unknown mean r_f, and variance $\sigma^2_{\hat{r}_f}$. The variance $\sigma^2_{\hat{r}_f}$ is estimated by parametric bootstrap, using 100 samples.

4.3.6 Analysis of T6: Location Estimation in 3D

We have seen that the foot position estimate error can be approximated as a zero mean Gaussian random variate. For the following error propagation steps we will assume that $\hat{r}_m$, $\hat{r}_p$, $\hat{D}_p$, and $\hat{D}_c$ are Gaussian random variables with true unknown means r_m, r_p, D_p, and D_c, and variances $\sigma^2_{\hat{r}_m}, \sigma^2_{\hat{r}_p}, \sigma^2_{\hat{D}_p}$, and $\sigma^2_{\hat{D}_c}$ respectively ($\sigma^2_{\hat{r}_m}$ and $\sigma^2_{\hat{D}_c}$ are estimated during the calibration phase). By applying linearization techniques in the geometric transformations, and by making independence assumptions on variables where applicable, it is easy to show how the estimates and its uncertainties propagate through the geometric transformations

$$\begin{aligned} R_p &= 2\frac{r_m r_f}{r_m^2 - r_p^2} H_o \\ D_p &= \sqrt{D_c^2 + R_p^2 - 2 D_c R_p \cos(\vartheta)} \end{aligned}$$

that were outlined in detail in section 4.1.2.

Derivations are straight forward by cascading intermediate results for covariance propagation applied to elementary transforms as addition, subtraction, multiplication and division as well as sin(.) and tan(.). Details can be found in appendix A.1.

The results for the uncertainties $\sigma^2_{\hat{R}_p}$ and $\sigma^2_{\hat{D}_p}$ in distance $\hat{R}_p$ respectively D_p are as follows. With equation (A.5) and (A.15) we drive

$$\sigma^2_{\hat{R}_p} = {H_0}^2 \sigma^2_{\hat{a}} + 4\,(a^2 + \sigma^2_{\hat{a}})\sigma^2_{\hat{H}_p} \qquad \text{with} \qquad a := \frac{r_m r_f}{r_m^2 - r_f^2} \tag{4.68}$$

$$\begin{aligned} \sigma^2_{\hat{a}} = {} & \frac{{r_m}^2 {r_f}^2 \left({r_m}^2 {\sigma_{\hat{r}_f}}^2 + {\sigma_{\hat{r}_m}}^2 {r_f}^2 + {\sigma_{\hat{r}_m}}^2 {\sigma_{\hat{r}_f}}^2\right)}{\left({r_m}^2 - {r_f}^2\right)^4} + \frac{4\,{r_m}^2 {\sigma_{\hat{r}_m}}^2 + {\sigma_{\hat{r}_m}}^4 + 4\,{r_f}^2 {\sigma_{\hat{r}_f}}^2 + {\sigma_{\hat{r}_f}}^4}{\left({r_m}^2 - {r_f}^2\right)^2} + \\ & \frac{\left({r_m}^2 {\sigma_{\hat{r}_f}}^2 + {\sigma_{\hat{r}_m}}^2 {r_f}^2 + {\sigma_{\hat{r}_m}}^2 {\sigma_{\hat{r}_f}}^2\right)}{\left({r_m}^2 - {r_f}^2\right)^4} \left(4\,{r_m}^2 {\sigma_{\hat{r}_m}}^2 + {\sigma_{\hat{r}_m}}^4 + 4\,{r_f}^2 {\sigma_{\hat{r}_f}}^2 + {\sigma_{\hat{r}_f}}^4\right) \end{aligned} \tag{4.69}$$

$$\begin{aligned} \sigma^2_{\hat{D}_p} = {} & \frac{N}{D} \quad \text{By concatenating equation (A.17) and (A.13) we find} \\ & D := 4\,\left({R_p}^2 + {D_c}^2 - 2\,R_p\,D_c\,\cos(v)\right) \quad \text{and} \\ & N := 2\,{\sigma_{\hat{R_p}}}^2 {\sigma_{\hat{D_c}}}^2 + 4\,{D_c}^2 {\sigma_{\hat{D_c}}}^2 + {\sigma_{\hat{R_p}}}^4 - \end{aligned} \tag{4.70}$$

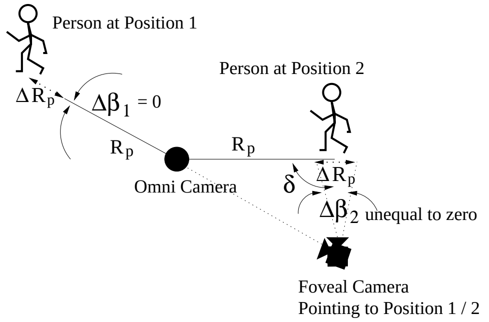

Figure 4.12: Local dependency - same uncertainty in R_p, different $\Delta\beta$. For $\delta = 90°$, $\Delta\beta$ and ΔR_p become maximal.

$$\begin{aligned}
&-8\,R_p\,D_c\,\cos(v){\sigma_{\hat{D_c}}}^2 + 4\,{R_p}^2{D_c}^2{\sigma_{\hat\vartheta}}^4\left(\sin(v)\right)^4 - \\
&-8\,R_p\,D_c\,\cos(v){\sigma_{\hat{R_p}}}^2 + {\sigma_{\hat{D_c}}}^4 + 4\,{R_p}^2{\sigma_{\hat{R_p}}}^2 + \\
&+4\,{\sigma_{\hat{R_p}}}^2{\sigma_{\hat{D_c}}}^2{\sigma_{\hat\vartheta}}^4\left(\sin(v)\right)^4 + 4\,{\sigma_{\hat{R_p}}}^2{\sigma_{\hat{D_c}}}^2\left(\cos(v)\right)^2 + \\
&+4\,{R_p}^2{\sigma_{\hat{D_c}}}^2\left(\cos(v)\right)^2 + 4\,{R_p}^2{\sigma_{\hat{D_c}}}^2{\sigma_{\hat\vartheta}}^4\left(\sin(v)\right)^4 + \\
&+4\,{\sigma_{\hat{R_p}}}^2{D_c}^2\left(\cos(v)\right)^2 + 4\,{\sigma_{\hat{R_p}}}^2{D_c}^2{\sigma_{\hat\vartheta}}^4\left(\sin(v)\right)^4
\end{aligned}$$

Figure 4.12 illustrates how uncertainties in 3D radial distance R_p influence the foveal camera control parameters.

4.3.7 Analysis of T7: Foveal Camera Control Parameter Estimation

For the following error propagation step we will assume that $\hat{H}_o$, $\hat{H}_p, \hat{R}_h$, and $\hat{H}_f$ are Gaussian random variables with true unknown means H_o, H_p, R_h, and H_f, and variances $\sigma^2_{\hat{H}_o}$, $\sigma^2_{\hat{H}_p}$, $\sigma^2_{\hat{R}_h}$, and $\sigma^2_{\hat{H}_f}$ respectively (all estimated in the calibration phase). As described above, $\hat{D}_p$ is assumed to be normal distributed with mean D_p, and variance $\sigma^2_{\hat{D}_p}$ (see equation (4.70)).

Similar to the analysis for T6 we can approximate tilt $\tan\hat\alpha$, and pan $\sin\hat\beta$ to be normal distributed with mean $\tan\alpha$ respectively $\sin\hat\beta$ and covariance $\sigma^2_{\tan\hat\alpha}$ respectively $\sigma^2_{\sin\hat\beta}$. As we will show in the following experiments, for

$$\begin{aligned}
\tan(\alpha) &= \frac{H_p - R_h - H_f}{D_p} \\
\sin(\beta) &= \frac{R_p}{D_p}\sin(\vartheta)
\end{aligned}$$

the independence assumptions hold, such that the final results for the uncertainties in tilt $\tan\hat\alpha$, and pan $\sin\hat\beta$ can be estimated by applying equation (A.3) and (A.7) respectively

(A.8) and (A.7) :

$$\sigma^2_{\tan\hat{\alpha}} = \frac{\sigma^2_{\hat{D}_p}}{D_p^4}\left((H_p - R_h - H_f)^2 + \sigma^2_{\hat{H}_p} + \sigma^2_{\hat{R}_h} + \sigma^2_{\hat{H}_f}\right) + \frac{\sigma^2_{\hat{H}_p} + \sigma^2_{\hat{R}_h} + \sigma^2_{\hat{H}_f}}{D_p^2} \tag{4.71}$$

$$\sigma^2_{\sin\hat{\beta}} = \frac{{R_p}^2 {\sigma_{\hat{\vartheta}}}^2 \cos^2\vartheta}{{D_p}^2} + \left(\sin^2\vartheta + {\sigma_{\hat{\vartheta}}}^2\cos^2\vartheta\right)\left(\frac{{R_p}^2 {\sigma_{\hat{D}_p}}^2}{{D_p}^4} + \frac{{\sigma_{\hat{R}_p}}^2}{{D_p}^2} + \frac{{\sigma_{\hat{R}_p}}^2 {\sigma_{\hat{D}_p}}^2}{{D_p}^4}\right) \tag{4.72}$$

It is clear that the systems design and analysis phases involve the choice of various models. For example, one has to make choices for the input perturbation model in the first step (T1) (e.g. the CCD noise model in this application), the prior distributions that influence various transformation stages, and hypotheses generation strategy. Moreover, the systematic propagation of the statistical distributions through the various transformations (T1 through T8) may involve approximations. There is a critical need to verify that these approximations indeed are realistic ones, and that the errors introduced by cascading several approximations do not render the analysis useless in practice. Thus, the systems analysis phase needs to be coupled with validations of three kinds:

- Verification of the correctness of the theoretical expressions under the given assumptions using simulated data.

- Verification of the correctness of the models themselves from real data ("Model Validation")

- The ultimate test that a given system is ready for commercial use is through large-scale experiments to verify that the designed system meets the requirements set. This is done by devising a careful experimental design to measure the performance of the system under various operating conditions.

In the following chapter on experiments it is illustrated how these steps are typically carried out. For a detailed discussion on empirical evaluation, performance characterization protocols, and experimental design issues please see [17],[54],[134], etc.

4.4 Experiments and Validation

As pointed out in 4.3 there is a critical need to verify that approximations in the modelling and transformation processes are indeed realistic ones, and that the errors introduced by cascading several approximations do not render the analysis useless in practice. This chapter illustrates validation of theoretical expressions and assumptions, model validation and long-term experiments to verify that the designed system meets the requirements set.

4.4.1 Validation of Assumptions by Simulation

We verify the correctness of our theoretical expressions and approximations through extensive simulations (Monte-Carlo simulation). In the following we show plots validating expressions for illumination normalization equation (4.53), Figure 4.13, and for foveal camera control parameters pan and tilt eqn. 4.71, 4.72 , Figure 4.14). The validation is performed in two steps: Verification of the theoretical results using simulations, and model validation using real data (Please see section 4.4.2). In the following, we include plots showing theoretically predicted answers and the differences between these predictions and simulated results, based on 10000 samples of normal distributed parameters. For demonstration purpose, parameters that represent the worst-case system behavior were chosen, based on our range of application settings (see 4.4.3). The figures show results obtained by using the following parameters: $\hat{H}_o = 2.46m$ $\hat{H}_p = 1.75m$ $\hat{H}_f = 1.82m$ $\hat{D}_c = 2.38m$. Following values for standard deviations were used: $\sigma_{\hat{H}_o} = 1cm$, $\sigma_{\hat{H}_p} = 5cm$, $\sigma_{\hat{H}_f} = 1cm$, $\sigma_{\hat{\vartheta}} = 1^\circ$, $\sigma_{\hat{D}_c} = 10cm$, $\sigma_{\hat{r}_f} = 3$ pixels, $\sigma = 2.5$ gray-level.

For validation of the distribution of the normalized color values, we fixed B at different values between 1 and 255 while varying R and G values in the range of 0 through 255. Figure 4.13 illustrates results for $B = 50$. In reality uncertainties are calculated on-line from the current data and are functions of the object, background and location of the object as well as the sensor noise.

Plots show the correctness of the derivations and approximations, give insights of the system limitations depending on user-defined tolerances, and show, where the assumptions hold. By examining parametric expressions for uncertainties (see equations (4.71), an (4.72)) the differences between simulation, and derived predictions can be explained by the error due to the linearization step at low levels of signal to noise ratio.

4.4.2 Validation of Models by Systematic Experiments

The correctness of the models is verified by comparing ground truth values of the control parameters of the camera against module estimates for mean and variance of the running system. First, we marked eight positions $P1 - P8$ of different radial distances and pan

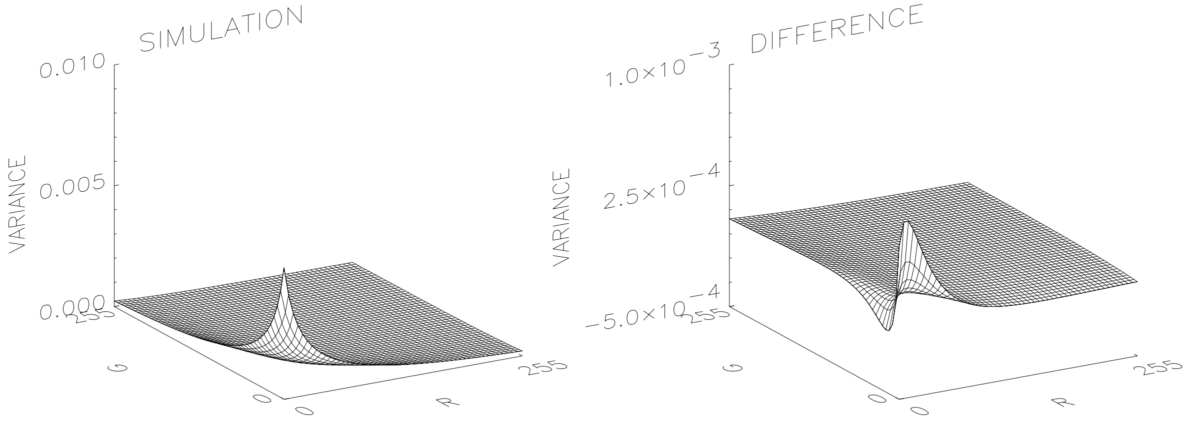

Figure 4.13: Color normalization: Variance $\sigma^2_{\hat{r}} + \sigma^2_{\hat{g}} + \sigma^2_{\hat{r}\hat{g}}$. Simulated values (left) and difference between simulation and theory (right). Standard deviation in the RGB bands were chosen to be $\sigma_R = \sigma_R = \sigma_R = 2.5$

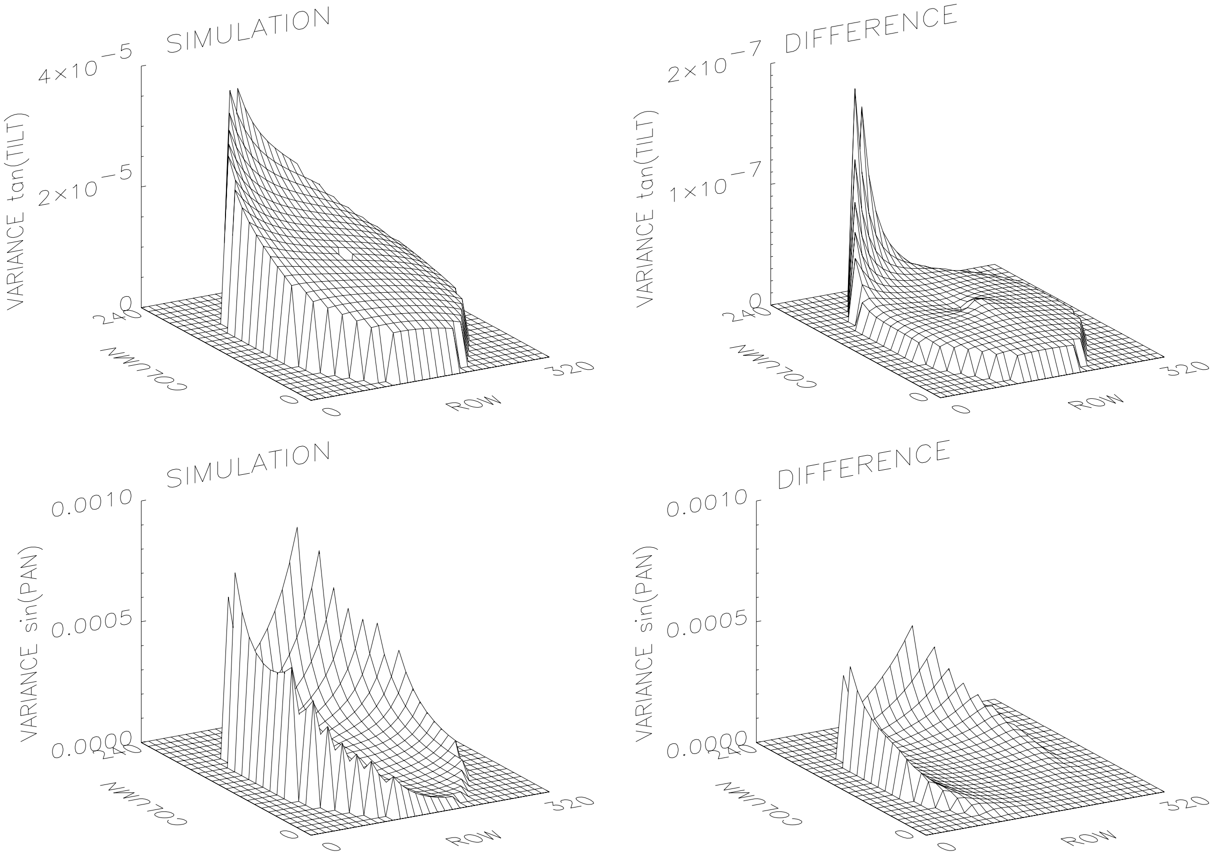

Figure 4.14: Variances of $\sin(\hat{\beta})$ and $\tan(\hat{\alpha})$ plotted as a function of person foot position in omni-image coordinates. Left: Simulation. Right: Difference between simulation and theory. Note different scale.

angles, see Figure 4.16. Positions, and test persons were chosen to simulate different positions, illumination, and contrast. In the following Table 4.5, we show the final foveal camera control parameters for one person. Ground truth values for the mean values were taken by measuring tilt angle α, and pan angle β by hand, and are compared against the corresponding mean of system measurements estimated from 100 trials per position and person.

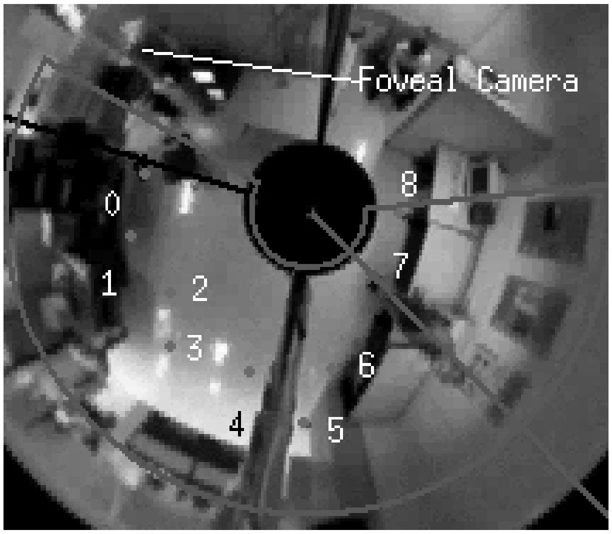

Figure 4.15: Omni-image; test positions at which snapshots in Figure 4.16 were taken.

The variances calculated by the system for pan and tilt angles are compared against the corresponding variance-estimates calculated based on the theoretical analysis. The comparison between system output and ground truth demonstrates the closeness between theory and experiment, see Table 4.5[9].

Table 4.5: Model Validation: First two lines show the predicted (hat̂), and experimental (tildẽ) variances for the tilt angle $\hat{\alpha}$. Next two lines correspond to pan angle $\hat{\beta}$.

$\times 10^{-5}$	P1	P2	P3	P4	P5	P6	P7	P8
$\hat{\sigma}^2_{\tan\hat{\alpha}}$	2.10	2.12	1.57	1.40	1.35	1.31	1.31	1.32
$\tilde{\sigma}^2_{\tan\hat{\alpha}}$	2.05	2.04	1.60	1.34	1.36	1.32	1.40	1.31
$\hat{\sigma}^2_{\sin\hat{\beta}}$	28.9	26.1	21.3	17.9	15.3	15.2	18.4	20.1
$\tilde{\sigma}^2_{\sin\hat{\beta}}$	25.9	24.1	19.5	15.1	14.9	15.0	18.1	19.3

A similar approach was taken to validate the zooming setting. Confidence percentile α_z was set to 95%. For 100 arbitrary positions in the room the foveal images were manually classified into two groups.

[9] Unfortunately, when the system was installed in the lab, we did only record the variances but not the standard deviation of the predicted variances. However, we repeat the experiment later under even relaxed constraints in an quasi-outdoor setting At that point we will present these data (see section 5.5).

- Group A: “Entire head visible; no part of the head cut off.”
- Group B: “Rest.”

In 92 of the 100 trials, assignment for *group A* was made by the system. The trials included having the person stand at several locations and wearing different clothing to simulate various contrasts and sizes. For performance in terms of percentage of pixels in the foveal frame being covered by the face, please refer to experimental results in Figure 4.17, 4.22, and Table 6.1.

4.4.3 System Performance Evaluation

In this section, we illustrate the performance of the running system under various conditions. Figures 4.17 through 4.22 demonstrate how the system can precisely locate a person and zoom onto its head while guaranteeing that the face is in the frame. The output of the foveal camera proved sufficient, as input for face detection algorithms (not part of this work).

Since this work does not analysis of the tracker, we zoom only in when the person stops moving, and zoom out if the location variation over time is larger than a threshold[10].

Figures 4.17 through 4.22 show snapshots of the running system. The foveal camera control parameters as well as the zoom parameter are functions of the geometry, as well as of the current uncertainties in the estimates. The more certain the estimate the more we can afford to zoom in. As described earlier the uncertainties are functions of the current scene, quality of segmentation, geometry, and calibration uncertainties. In these figures, the foot position estimate is displayed as a cross. Where the cross does not sit on the top of the toes, the camera does not zoom in too much. Precise estimation is characterized by stable positioning of the cross on the shoe. We tested the system without changing parameters in different settings (office, conference room) under different conditions. Varying object/background contrast was obtained by having the person move in front of different background areas and letting him wear different clothing with varied color and texture. The results in all experiments were obtained with user specified probability α_Z that the detected person’s face is completely contained in the image while zoomed in to the maximal extent.

In each figure 4.19 through 4.20 the left column shows the Omni-image: The red segment defines region of interest, the line through the center corresponds to the angular component of the position estimate. The inner cross corresponds to the radial foot position estimate while the outer cross shows the estimate for the head position. The right column shows the corresponding foveal frame.

[10] This is of course different from the uncertaity in the estimates based on the analysis discribed earlier, which investigates into uncertainties of the estimates.

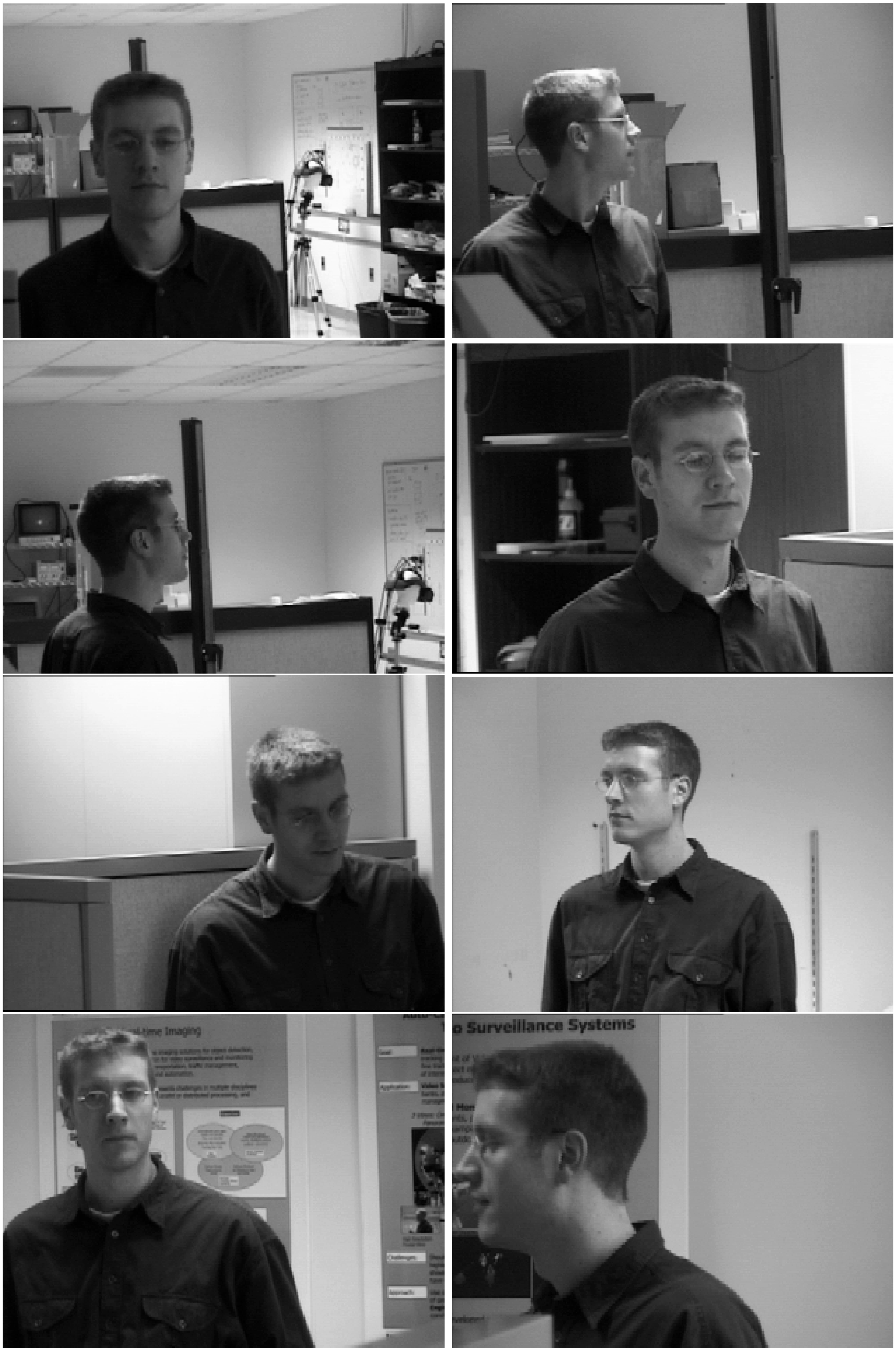

Figure 4.16: Sequence 1. Left right, top down: Foveal image corresponding to positions P1–P8 in Figure 4.15.

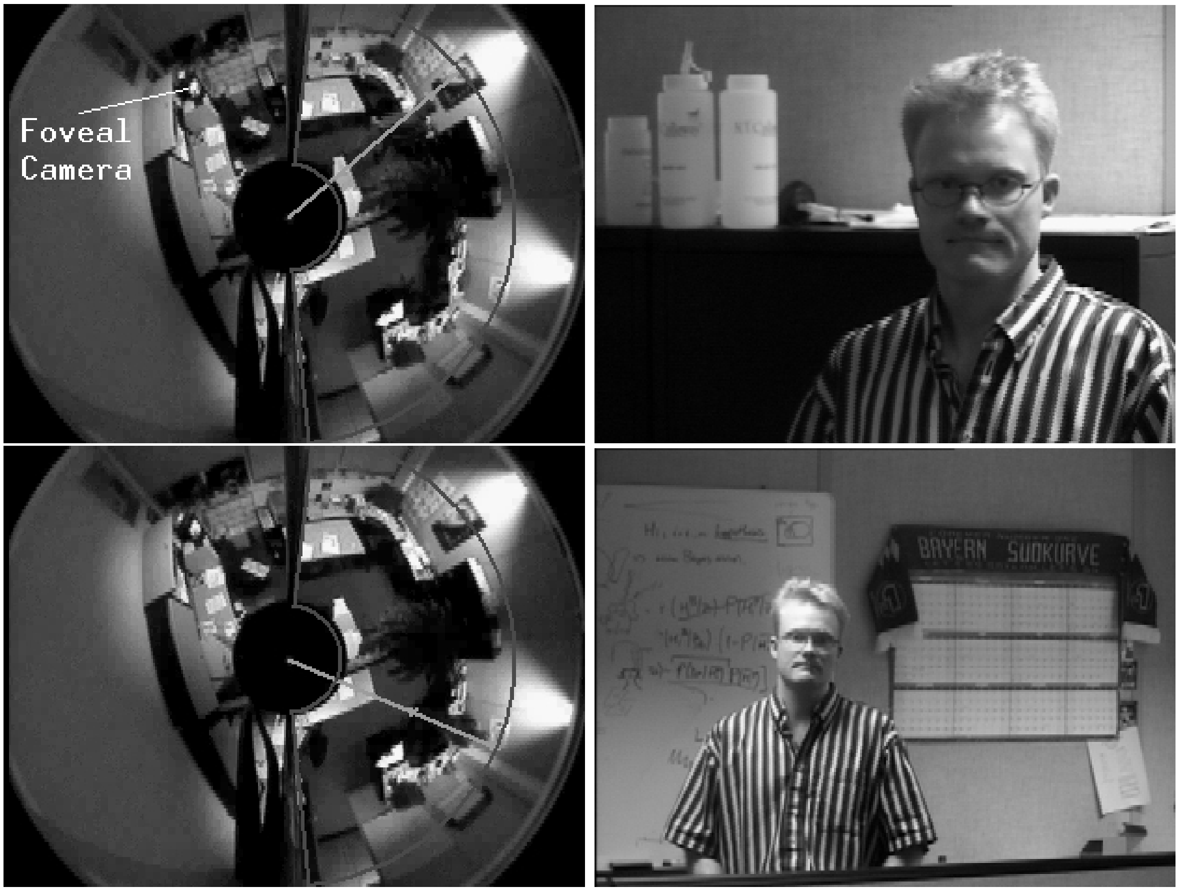

Figure 4.17: Snap shots office O1-O2. Top: Even though feet are occluded by the edge of the table, the foot location is estimated precisely and with high confidence: Foveal camera zooms in. Bottom: Foot position is estimated quite precisely, nevertheless the uncertainty is quite high, since the background is extremely dark behind the trousers such that only parts of the person are segmented and therefor, the position estimate remains unreliable; foveal camera does not zoom in much.

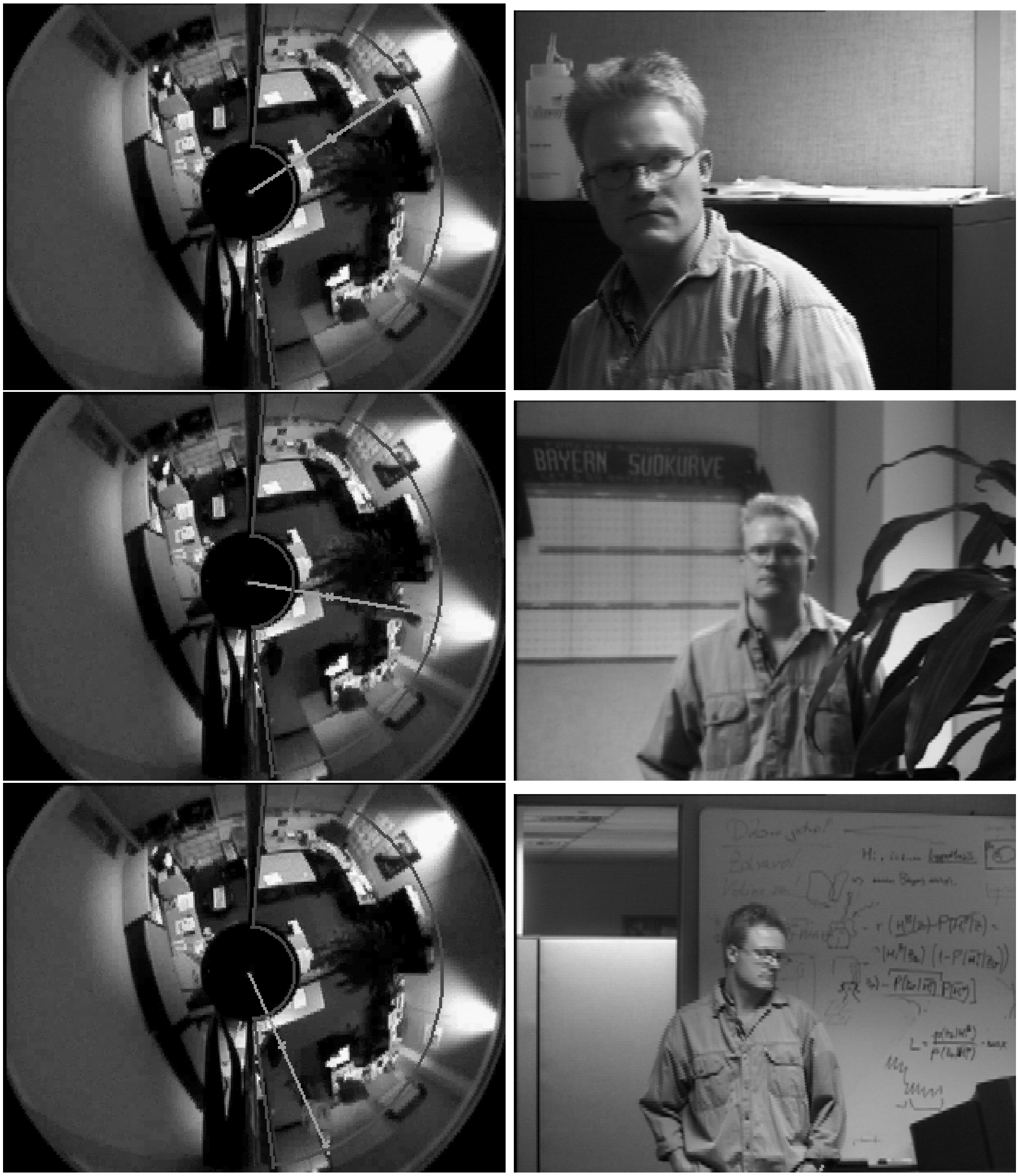

Figure 4.18: Snap shots office O3-O5. Top: High contrast, reliable segmentation; zoomed in. Center: Partial occlusion, therefor increased uncertainty; not too much zoomed in. Bottom: Low contrast, foot position quite off / unreliable; zoomed out. Face is only centered because estimated foot position, omni camera and foveal camera are aligned.

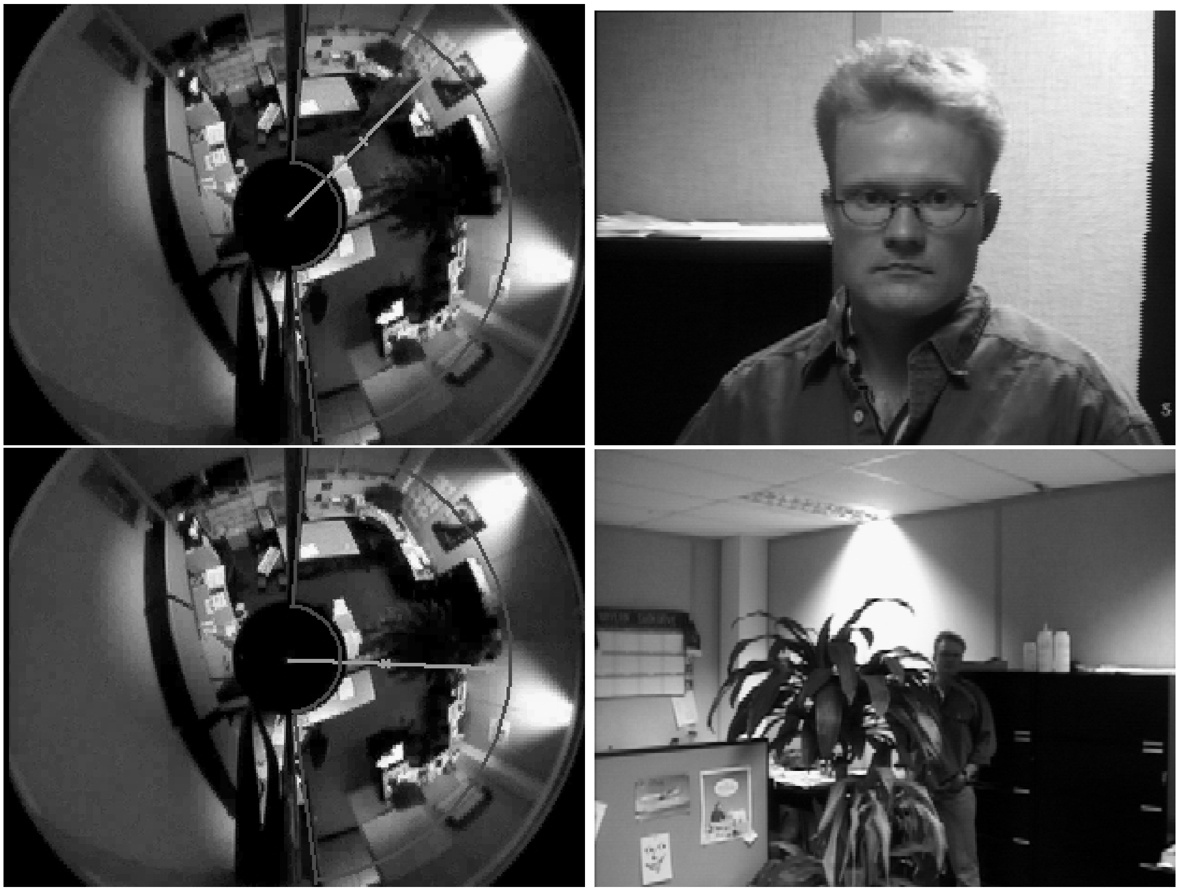

Figure 4.19: Snap shots office O6-O7. Top: Precise location estimation, zoomed in. Bottom: Extreme occlusion, prior model for projected person does not match data such that the uncertainty in the location estimation is very large and the system zooms out.

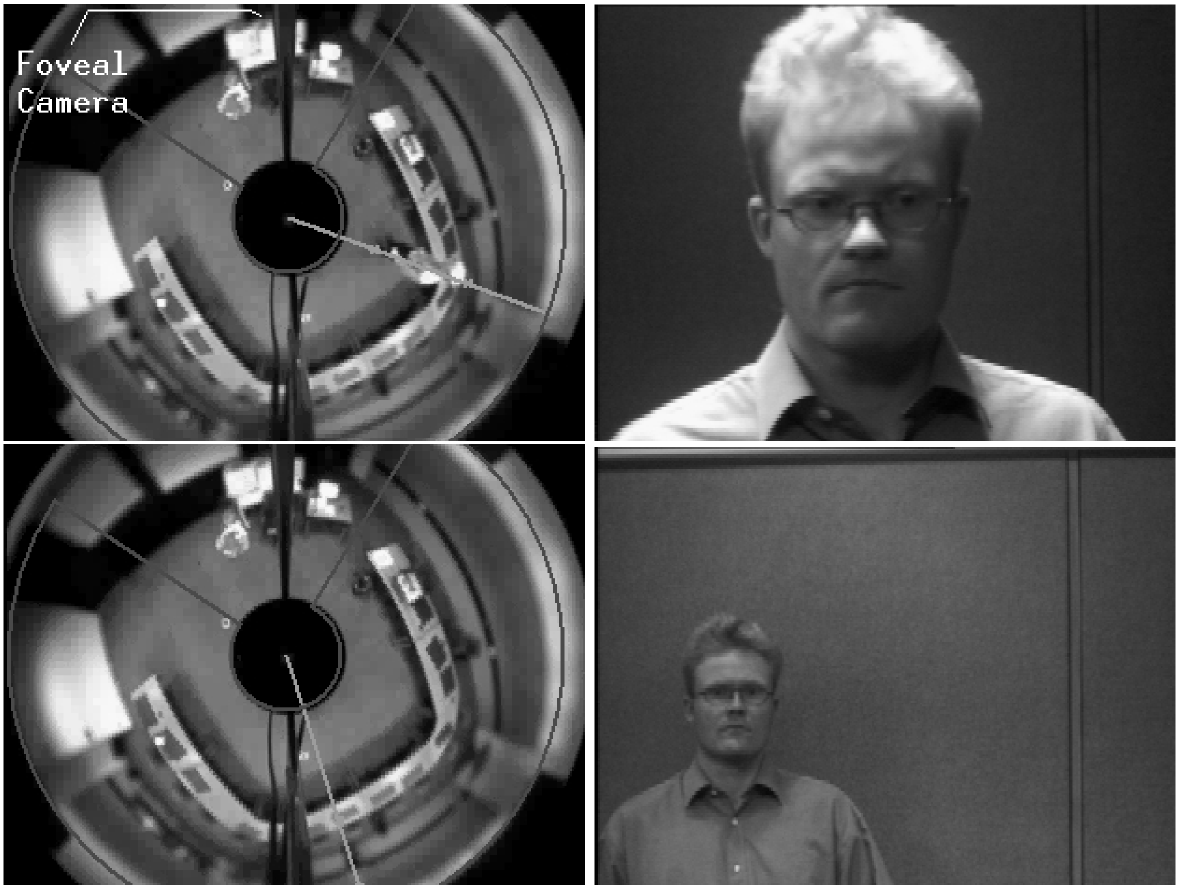

Figure 4.20: Snap shots conference room C1-C2. Top: Precise and reliable localization; zoomed in to the maximal extend. Bottom: Person only partially captured; segmented region does not correspond to expected region given the geometry model, high uncertainty; zoomed out. Since foot position is quite off, face is not centered.

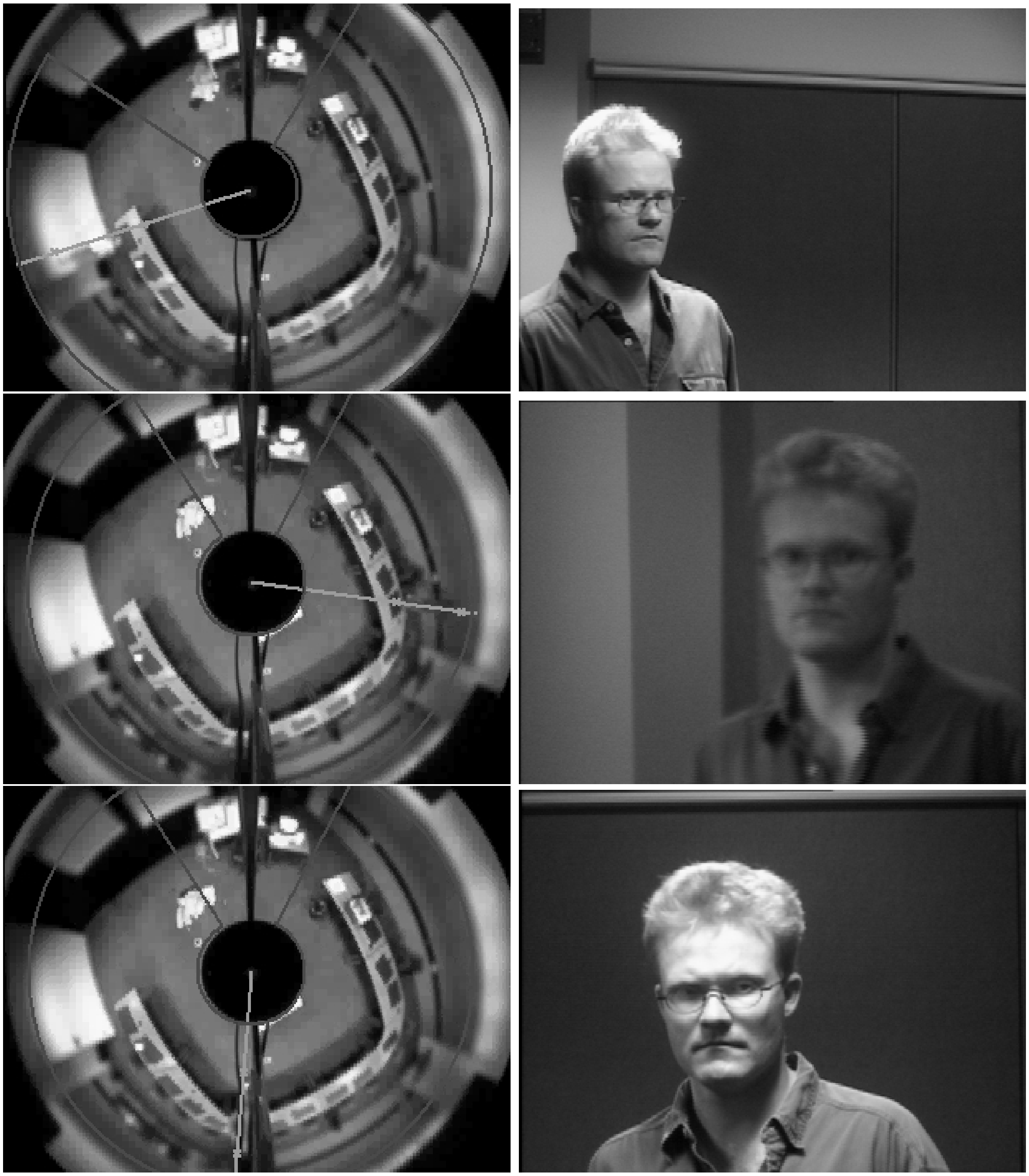

Figure 4.21: Snap shots conference room C3-C5. Top: Low contrast between white trousers and saturated background. Strong segmentation only for parts of the object such that given the geometry model, many foot positions are possible, hence the estimate is unreliable, and it is zoomed out. Indeed, the foot position is off and the face is not centered. Center: Precise and reliable estimate; zoomed in. Bottom: Even though partially occluded, the contrast at that location is high enough to compensate for limited number of pixels: Precise and reliable segmentation, zoomed in.

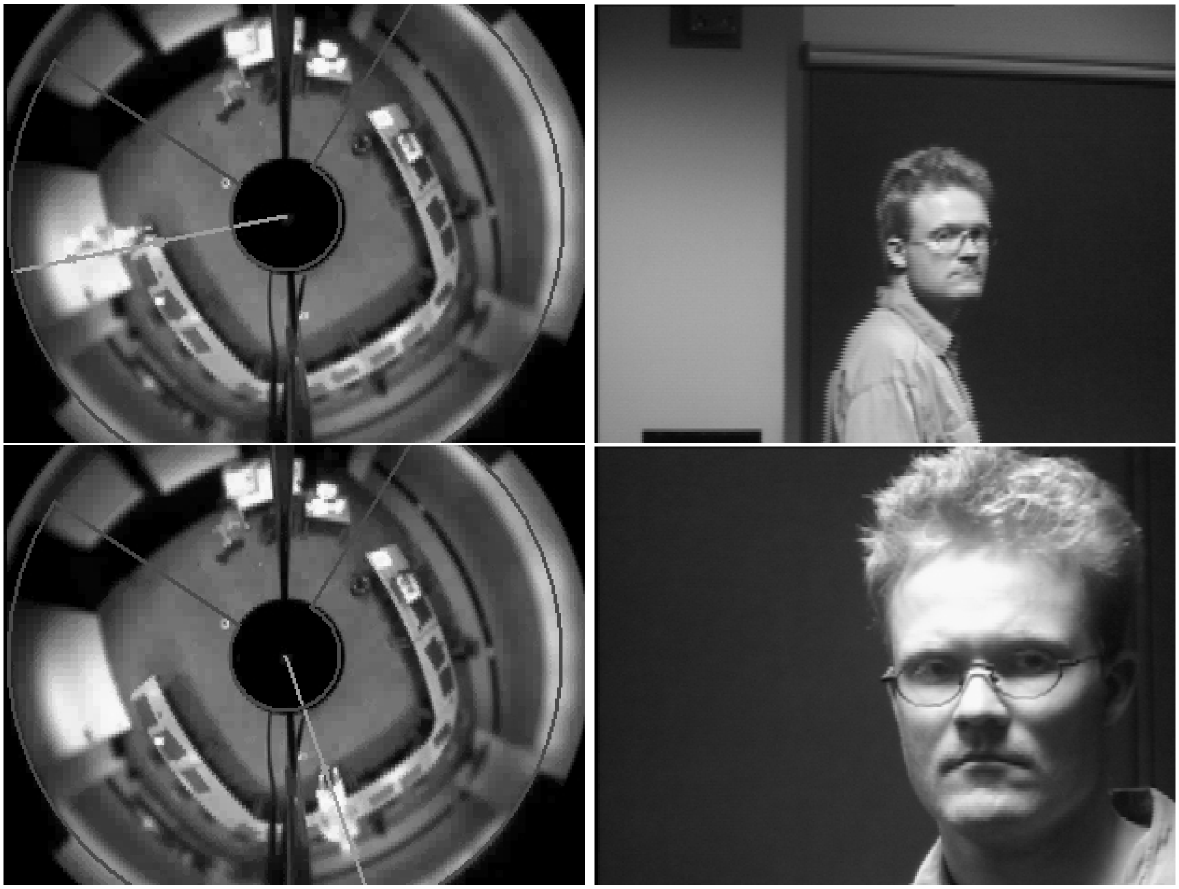

Figure 4.22: Snap shots conference room C6-C7. Top: Saturated background provides unreliable segmentation, hance zoomed out, even though the foot position is estimated quire precisely in this particular case.

Chapter 5

Evolution of the System

In this chapter it will be illustrated how the existing system, which was designed and analyzed as described above can be extended to relax the system operating conditions with minimal re-design and analysis efforts. The key conclusion is that by choosing appropriate modules and suitable statistical representations, we are able to re-use existing system design and performance analysis results. In the following will reinforce the methodology described earlier that proposes a design and analysis. Originally, the system was designed for indoor (static illumination) settings. Now the goal is to extend the system to deal with dynamic illumination changes such that extensive re-use of the original system and its performance characterization results can be achieved.

5.1 Approach to Maximal Re-Use of Modules

Assuming that the system has been designed, analyzed, and tested for a given restricted application scenario (as previously described), the question is how one can adapt the system to operate under a less restrictive input condition. In order to adapt the system configuration to meet extended system requirements one has to identify how a change in requirement influences the existing modules and the system architecture. A redesign of all modules affected is one option. Another way is to utilize third party modules to replace existing ones. To retain the advantages derived by following a systematic methodology during the design and analysis phases of the original system one has to choose external modules that satisfy the following constraints:

- They should be amenable to statistical analysis so that a probabilistic fusion of the component with the existing system is feasible, and

- They should facilitate ease of re-use of previous modules and their performance analysis.

The first point is important in order to be within the systems engineering formalism. Moreover it is necessary to identify how the probabilistic fusion of two different modules addressing the same task can be fused to derive hybrid solutions that meet the new requirements. If one represents the system as an execution graph wherein the nodes are data structures and the edges correspond to the transformations, a statistical characterization is associated with each edge in the graph. The total system analysis, [97], essentially provides the relationship between the final output statistics and the input statistics as a function of all the system parameters. The replacement of a module within the larger system corresponds to the change of an edge in the graph. In order for the total systems analysis performed in the previous design cycle to be re-usable, one has to choose a new module that satisfies the statistical distribution conditions for the input and output data types. Thus, the second point is needed to not re-do the entire system analysis phase as a result of modification of the input/output distribution in one module of the system. In situations where no existing external module satisfies this constraint, we propose to devise a *wrapper transformation* that essentially molds the external module and makes its statistical interfaces consistent with the existing framework[1]. Our approach therefore consists of:

- Identifying modules influenced by the translation of the relaxation in the application constraints.

- Finding replacement modules that satisfy the engineering constraints mentioned above.

- Performing statistical characterization of the replacement modules (or their hybrid design variations obtained by fusing existing modules with the replacement modules)

- Developing the wrapper that enables us to integrate it with the existing system with no re-analysis phase. Once the design is accomplished, the validation and test phase follows.

5.2 Relaxed Constraints on Scene and Illumination

In chapter 4 the statistical modeling and performance characterization under indoor conditions of our dual-camera surveillance system was discussed. The objective in this chapter is to describe how this system can be refined to handle less restrictive input conditions while retaining most of the original system design intact. The goal is to apply the system

[1] This is the statistical equivalent of wrapping legacy code written in Fortran to be usable in C++, for instance.

to quasi-outdoor setting (i.e. lobbies with external illumination) with minimal re-design and analysis effort. Table 5.1 shows the relaxed constraints. The original system was designed to handle shadow effects and changes in camera gain, the new system must also handle slow changes in the background and fast changes between multiple background modes. The system needs to operate 24 hours a day, 7 days a week in an office building entrance lobby, that is lit by artificial light during night and primarily natural light during day time.

Table 5.1: Comparison of old and new system requirements

Previous System Constraints	New System Features
constant background	changing background
constant illumination	variable illumination
single mode background	multi modal background
restricted operational in black and saturated background areas	operational on full sensor input range

Since the constraint being relaxed has to do with illumination, we need to replace the relevant illumination invariant module in our original system. As described in section 4.2.1 the original system used the prior knowledge that the scene consisted of light sources with same spectra but arbitrary intensities. No background adaptation to handle the changes in spectra was done. The relaxation of this application constraint necessitates the use of a background adaptation module. In our review of the literature, we could not find a module that satisfies the requirements that the output distribution of the background adaptation module is of the form suitable for input into our people detection module. Therefore, we propose a method, which combines the advantages of our existing change detection module with the advantages of the background adaptation algorithm by Stauffer-Grimson [111]. We will see that this fusion itself presents challenges due to complexity in analysis and due to subtle differences in the output feature space: Shadows are assigned object labels by [111] while our existing change detection algorithm assigns background labels to the shadow pixels. We address this by adding an augmentation module that alters the output data to be in the same feature space.

5.3 Module Description: Design and Analysis

This section provides a description of the background adaptation and change detection modules being fused. Since section 4.2.1 and 4.2.2 already describe in detail the static change detection module operating on normalized color the emphasis in this section is put on introducing the third party module.

5.3.1 Gain and Shadow Invariant Change Detection Module

In this subsection we briefly review our illumination invariant change detection method based on normalized color as described in the design section 4.2.1 and 4.2.2. The module takes as input a vector $(\hat{R}, \hat{G}, \hat{B})^T$ which is assumed to be normal distributed with mean $(R, G, B)^T$ and covariance matrix $\Sigma = \text{diag}(\sigma_R^2, \sigma_G^2, \sigma_B^2)$, normalizes it by $\hat{S} = \hat{R} + \hat{G} + \hat{B}$ and provides a distance metric $\hat{d}^2$ between the current values $\hat{\mu}_c = (\hat{r}_c, \hat{g}_c)^T$ and a background representation $\hat{\mu}_b = (\hat{r}_c, \hat{g}_c)$ in the normalized space, where $\hat{r} = \frac{\hat{R}}{\hat{S}}$, and $\hat{g} = \frac{\hat{G}}{\hat{S}}$, when subscripts b, c for background respectively current image are omitted. The probability of a pixel being background corresponds to the distance measure

$$\begin{aligned} \hat{d}^2 &= (\hat{\mu}_\mathbf{b} - \hat{\mu}_\mathbf{c})^T \left(2\mathbf{\Sigma}_{\hat{\mathbf{r}}_\mathbf{b},\hat{\mathbf{g}}_\mathbf{b}}\right)^{-1} (\hat{\mu}_\mathbf{b} - \hat{\mu}_\mathbf{c}) \qquad \text{with} \\ \mathbf{\Sigma}_{\hat{\mathbf{r}}_\mathbf{b},\hat{\mathbf{g}}_\mathbf{b}} &= \\ & \frac{\sigma_S^2}{S^2} \begin{pmatrix} \frac{\sigma_R^2}{\sigma_I^2}(1 - \frac{2R}{S}) + \frac{R^2}{S^2} & -\frac{\sigma_G^2 R + \sigma_R^2 G}{\sigma_I^2 S} + \frac{RG}{S^2} \\ -\frac{\sigma_G^2 R + \sigma_R^2 G}{\sigma_I^2 S} + \frac{RG}{S^2} & \frac{\sigma_G^2}{\sigma_I^2}(1 - \frac{2G}{S}) + \frac{G^2}{S^2} \end{pmatrix} \end{aligned} \tag{5.1}$$

where $\sigma_S^2 = (\sigma_R^2 + \sigma_G^2 + \sigma_B^2)$. For details please see [41].

Note, that in the normalized space the covariance matrix $\mathbf{\Sigma}_{\hat{\mathbf{r}}_\mathbf{b},\hat{\mathbf{g}}_\mathbf{b}}$ for each pixel is different. This method was proved to perform very precisely and accurately in indoor situations with static illumination within our module framework, but it is not suitable for situations of varying light conditions and changes in the background. Due to the nature of normalization, this module ignores cues provided by the signal intensity. Key-features are invariance against shadows and changes in camera gain, and the notion of sensor uncertainty. Nevertheless, the miss-detection rate is high in areas that are dark and saturated due to the large uncertainty in the normalized color space. It also fails when the input has no color information.

Analysis: We briefly review the results from the analysis section 4.3.1 and 4.3.2. The statistical characterization for the normalized color segmentation module can be summarized as follows: For normal distributed input parameters $\hat{R}, \hat{G}, \hat{B}$, the output statistic $\hat{d}^2$ (see equation (5.1)) is χ^2 distributed with two degrees of freedom for background pixels. For object pixels, $\hat{d}^2$ can be approximated by a non-central χ^2 distribution with two degrees of freedom, and non-centrality parameter c. $\hat{d}^2$ is exactly non-central χ^2 distributed with two degrees of freedom if the covariance matrix for background and foreground are identical.

5.3.2 Background Adaptation Module

In [111] Stauffer and Grimson propose a background adaptation scheme, that adapts to slowly drifting multi modal background intensities. They model each pixel value X_t

as a mixture of K Gaussians with weights w_i, means μ_i, variances σ_i^2 and use an online approximation to update the model. For details, please refer to appendix A.2. For simplification reason we omit time index t from here on. They introduce a pixel labeling process, which is primarily based on the assumption that the least frequently occurring component in the mixture with large variance is more likely to be objects. This is done by the use of a threshold T that is based on the prior probability of a pixel being background. A pixel is labeled "background", if it is closest in distance to one of the top B distribution components in the mixture, where $B = \text{argmin}_b(\sum_{k=0}^{b} w_k > T)$. The model parameters for the mode μ_i, variance σ_i^2, and weight w_i, which represents the current data best are updated following an exponential forgetting scheme with learning constant α. Pixels which are outside an $2.5\sigma_i$-interval around each of the K modes are labeled "object" and modeled by a new Gaussian distribution. The mean of this Gaussian corresponds to the current pixel value, the variance is initialized with a high value ${\sigma^2}_{\text{init}}$, its weight w_{init}, with a small value. The new mode replaces the background mode with least supporting evidence. This is the mixture component with the smallest ratio of weight to the standard deviation.

Statistical Analysis: To characterize the statistical behavior of the background adaptation module, we conducted numerous experiments on real data as well as on simulated data with similar results. We generated random samples from a mixture distribution with model parameters w_i, μ_i, σ_i^2 with $i \in \{0, 1, 2, 3\}$. Table 5.2 shows the parameter settings used along with the ideal model parameters. Initialization for the modes of the components were done randomly or in a deterministic fashion. For instance, in the example shown, we initialized all modes with the same parameters: $w = 1.88, \mu = 0, \sigma_i^2 = 25$ except for the first mode's mean, which we initialized with 10. In Figure 5.1 and 5.2 we show for simulated data how the model parameters typically evolve over 10,000 time intervals.

Table 5.2: Parameter setting for background adaptation.

T	w_{init}	σ^2_{init}	α	w_0	w_1	w_2	w_3
0.9	0.05	25.0	0.03	$0.0\bar{5}$	$0.1\bar{6}$	$0.\bar{3}$	$0.\bar{4}$

μ_0	μ_1	μ_2	μ_3	σ_0^2	σ_1^2	σ_2^2	σ_3^2
10	50	150	205	0.3	0.3	0.3	0.3

The experimental analysis shows, that only the modes of the mixture distribution are estimated and tracked correctly. The variance and the weights are unstable and unreliable. They do not track the data and do not converge. Even though the experiment uses a random sample from a stationary mixture distribution, the variance and weights tend to oscillate arbitrarily and are frequently re-initialized.

Even if we assumed that the parameters do converge, for every sample that falls outside all of the 2.5σ-intervals, a new mode is introduced and initialized with a high variance and a small weight. For samples from the background mixture this occurs in approximately 1% of the cases. In other words, for every pixel the introduction of the new mode occurs on an average of less than every 10 seconds if the system processes more than 10 frames per second. Depending on the update factor α the model parameters $\sigma^2_{k,t}$, and $w_{k,t}$ are somewhere between the initial and the true value but do not represent the parameters of the true underlying background distribution. Since the variances are not constrained to have a lower-bound, variances from most frequently occurring modes are constantly reduced and become significantly smaller then the true variances, such that this new mode introduction happens even more often.

This is not a problem in the original implementation because the weights and the variances are not used in any way in a subsequent processing step. They use connected components followed by a region size threshold to prune false detection. On the contrary, in our methodology, we need to characterize the stability of these estimates in order to determine how they can be fused to another module and do the systems analysis. Since the experiments show the mean estimates are stable, we explicitly use this feature to develop the hybrid algorithm. We further note, that the estimated modes are approximately normal distributed such that the difference between the current measurement and the closest background mode is approximately normal distributed with zero-mean and a covariance that is different from model parameter ${\sigma_i}^2$ (see Fig. 5.4).

5.4 Fusion of Modules

In this section, we will show how to fuse two modules statistically correctly to obtain a modified system to meet the old and newly added requirements simultaneously. Figure 5.3 shows the block diagram. The main essence in the fusion algorithm is as follows. The change in background is modelled as two separate effects:

- Change due to the illumination spectrum and non-linear dynamics.
- Change due to sudden camera gain/shadow changes.

The Stauffer-Grimson (SG) algorithm is ideally suited to deal with changes in the illumination spectrum and slow dynamics, while the normalized color change detection algorithm is invariant to gain and shadows. By using SG algorithm first and feeding its internal state to the normalized color change detection algorithm we gain the advantages of both. Nevertheless, we still have two issues to contend with:

- The SG algorithm does not discriminate between shadows and objects and

- The normalized color module fails when the input signal has no color and its discrimination power diminishes in dark and saturated areas.

We will show that these problems can be solved by augmenting the SG algorithm to handle shadow information and by proper statistical fusion of the two algorithm outputs. The added requirement is that after we have done the fusion the output of the fused algorithm must have the same output distribution suitable for integration into the original system.

5.4.1 Updating Normalized Color Model

We have seen that an analysis of the Stauffer-Grimson module demonstrated that the modes of the mixture [2] are stable within a time window. Further, the distribution of estimated modes $\hat{\mu}_i$ for each color band can be approximated by normal distributions thus matching the input distributional assumptions for the normalized color module. Summarizing the estimated modes $\hat{\mu}_i$ for each color band RGB in a single vector $\mathbf{v_{b,i}}$ we define $\mathbf{v_{b,i}} = (\hat{\mu}_{R,i}, \hat{\mu}_{G,i}, \hat{\mu}_{B,i})^T$. Therefore, we can use the components of $\mathbf{v_b}$ as estimates to compute $\mu_\mathbf{b}$ and $\mathbf{\Sigma}_{\hat{\mathbf{r}}_\mathbf{b},\hat{\mathbf{g}}_\mathbf{b}}$ in our normalized-color background-model as laid out in equation (4.53,4.11).

Let X denote a current pixel value representing one of the three color bands RGB. We define the corresponding background mode X_b as the one closest to any of the B background modes $\hat{\mu}_i$:

$$\hat{X}_b = \mu_j | j = \underset{k}{\text{argmin}} (\hat{\mu}_k - \hat{X})^2 \forall k \in \{0...\} \tag{5.2}$$

Summarizing the estimated background values $\hat{X}_b$ for each color band RGB in a single vector $\mathbf{v_b}$ we define $\mathbf{v_b} = (\hat{R,} b, \hat{G,} b, \hat{B,} b)^T$.

With $\mathbf{v_b} = (\hat{R,} b, \hat{G,} b, \hat{B,} b)^T$ and equation (4.53,4.11) we then update covariance matrix $\mathbf{\Sigma}_{\hat{\mathbf{r}}_\mathbf{b},\hat{\mathbf{g}}_\mathbf{b}}$ and mean $\hat{\mu}_\mathbf{b}$ of the normalized color background.

The Mahalanobis distance $\hat{d}^2$ between the current normalized color value obtained from $\hat{\mu}_\mathbf{c}$ and $\hat{\mu}_\mathbf{b}$ is used as the shadow/gain invariant change detection measure. The distribution of this statistic is approximately chi-squared distributed with 2 degrees of freedom.

5.4.2 Augmented Shadow-Invariant SG-Algorithm

As discussed above, the normalized color information should be augmented with intensity information to deal with dark/saturated areas and when no color information is present in the signal. Therefore, we apply the Stauffer-Grimson algorithm also to gray scale values

[2] We apply the algorithm to RGB bands independently such that for simplification reasons X denotes a pixel value for one of the three channels RGB.

$I = (R+G+B)/3$ and search for a representation which allows statistically correct fusion of the two representations. This second kind of fusion differs from the one just discussed in the previous section. Table 5.3 compares both representations. The SG algorithm is not designed to distinguish between shadow and object pixels. They are both labeled as object, since they occur simultaneously and have large variance. Therefore, we add a computational test that augments the gray scale background state model in SG to include a shadow component. Please note, that for RGB space, the normalized color representation automatically takes care of the camera gain effect as well as the shadow effect and is more time efficient, so we don't maintain a mixture density in normalized color space. Under the assumption that shadow pixel values are multiplicative factors (identical in each color band) of the corresponding background color, pixels that are labeled as non-background pixels by the original SG algorithm are further classified as being shadow or object. This is done based on a classical statistical hypothesis test [75] for the current pixel being shadow (please refer to appendix A.3 for further details and derivations). Along with a label, this method also provides a probability that the given label is correct. Formally, the number of background modes is augmented to be $B + 1$ where the last mode is the additional shadow mode. Let $\mu_{I,i}, i = 1, \ldots, B + 1$ denote the means of background mixture model for a given gray pixel.

Table 5.3: Pros and Cons in different feature spaces.

	Normalized color	Intensity
Pro	eliminates shadow	similar discrimination power on full sensor input range
	very stable representation	operates in poorly lit environments as well
Con	reduced discrimination power in dark and saturated background areas	tends to assign shadow pixels to object group

The distance between the current intensity value $\hat{I}_c$ and the closest $\mu_{I,i}$'s, denoted by $\hat{I}'_b$, is used as the change detection measure. We denote this minimum distance by $\hat{\Delta}$. Similar to equation (5.2)we derive with

$$\hat{I}'_b \quad = \quad \mu_{I,j} | j = \underset{k}{\text{argmin}} (\hat{\mu}_{I,k} - \hat{I}_c)^2 \forall k \in \{0...B + 1\} \tag{5.3}$$

$$\hat{\Delta} \quad = \quad \hat{I}'_b - \hat{\mu}_{I,j} \sim N(0, \sigma_{\hat{I}'_b}\sqrt{2}) \tag{5.4}$$

The distribution of $\hat{\Delta}$ is also approximated by a Gaussian distribution[3]. Note that the σ_i

[3] The actual distribution is a mixture that can be well approximated by a mixture of two Gaussians

values provided by SG algorithm are not used in this distance measure because of their instability. Moreover, the variance of the estimated mode is not the same as the variances of the components of the mixture. The variance of the estimated mode is obtained through a analysis of the empirical distribution of $\hat{\Delta}$ for the background pixels. The local fluctuations in the modes of the mixture distribution (within a small window of time) are primarily assumed to be due to a global illumination effect such as camera gain. Therefore, the trimmed standard deviation of the histogram of $\hat{\Delta}$ is used as an estimate of the standard deviation $\sigma_{\hat{I'}_b}$ of $\hat{I'}_b$.

5.4.3 Fused Change Detection Measure

The final goal is to statistically fuse the normalized color feature with the shadow invariant intensity feature such that the new feature statistic has the same characteristic as the old one. This is important in order to ensure, that the modules which follow in the original system and take this new statistic as input can still be used, and the systems analysis conducted earlier remains valid. Therefore, we need to find a feature for the augmented intensity representation that will provide a change detection measure that is χ^2 distributed, since the original test statistic $\hat{d}^2$ was also χ^2 distributed, see section 4.3.2. Knowing from the analysis in 5.3.2 that the means are stable and approximately normal distributed, we define a Mahalanobis distance $\hat{d'}^2$ similar to $\hat{d}^2$ as proposed in equation (4.58):

$$
\begin{aligned}
\hat{d'}^2 = \quad & \begin{pmatrix} \hat{r}_b - \hat{r}_c \\ \hat{g}_b - \hat{g}_c \\ \hat{I}'_b - \hat{I}_c \end{pmatrix}^T \left(2\mathbf{\Sigma}_{\hat{\mathbf{r}}_\mathbf{b},\hat{\mathbf{g}}_\mathbf{b},\hat{\mathbf{I}}_\mathbf{b}}\right)^{-1} \begin{pmatrix} \hat{r}_b - \hat{r}_c \\ \hat{g}_b - \hat{g}_c \\ \hat{I}'_b - \hat{I}_c \end{pmatrix} \\
\text{with} \quad & \begin{pmatrix} \hat{r}_b - \hat{r}_c \\ \hat{g}_b - \hat{g}_c \\ \hat{I}'_b - \hat{I}_c \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_\mathbf{b} - \mu_\mathbf{c} \\ I'_b - I_c \end{pmatrix}, 2\mathbf{\Sigma}_{\hat{\mathbf{r}}_\mathbf{b},\hat{\mathbf{g}}_\mathbf{b},\hat{\mathbf{I}}_\mathbf{b}}\right)
\end{aligned}
$$

with I_c denoting the current intensity value and I'_b denoting the mode in the background mixture that is closest to I_c. Experiments show that the correlation between normalized color representation and the intensity based mixture model is negligible such that the new change detection measure:

$$\hat{d'}^2 = \hat{d}^2 + \frac{(\hat{I}'_b - \hat{I}_c)^2}{2\sigma^2_{\hat{I}_b}}$$

where $\hat{d}^2$ is identical with the output of the change detection module for normalized color in section 4.3.2 equation (4.14). Thus, we know that under the given conditions $\hat{d'}^2$ is approximately central χ^2 distributed with 3 degrees of freedom for background pixels

both with zero mean, but one with very small variance due to the *min* operation.

(including shadow pixels) and non-central χ^2 distributed with 3 degrees of freedom for object pixels. This shows that the distributional form of the statistic which serves as input for the next module in our original system remains the same and the modules may remain untouched. The only difference is in the parameter of the distribution, i.e. the number of degrees of freedom changes from 2 to 3 in this case. Figure 5.5 illustrates the cumulative distribution functions for the intensity based distance measure, the normalized color-based measure, and the fused measure when there are no objects. It shows that the approximations are reasonable.

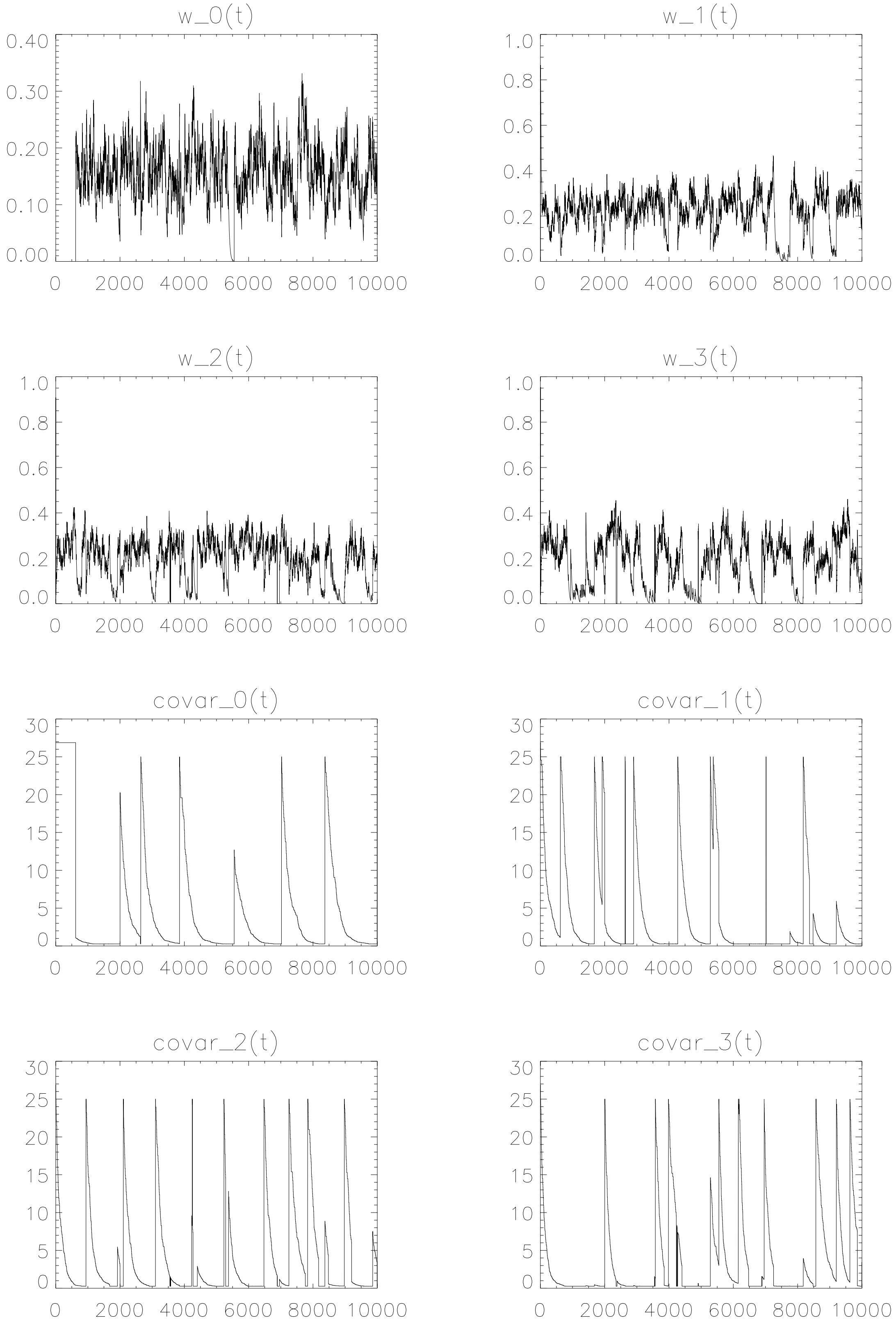

Figure 5.1: Model parameters w, σ^2 evolving over time for $K = 4$.

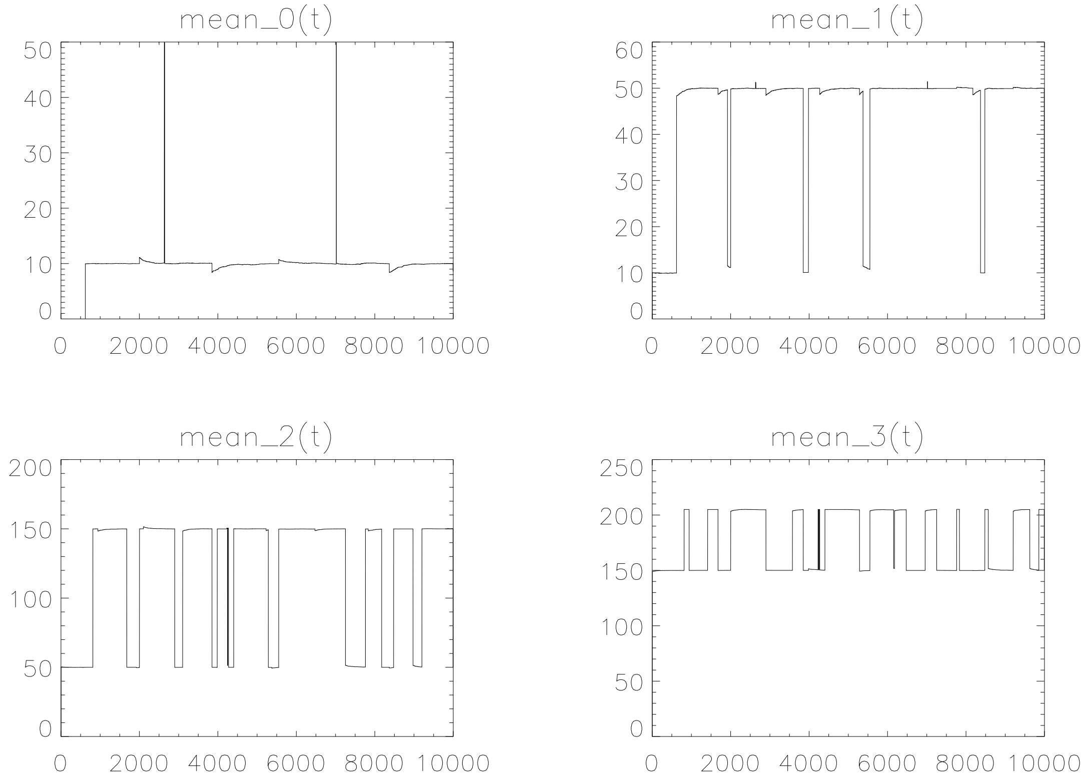

Figure 5.2: Model parameter w evolving over time for $K = 4$. Please note, that the modes are stable, only labels alternate.

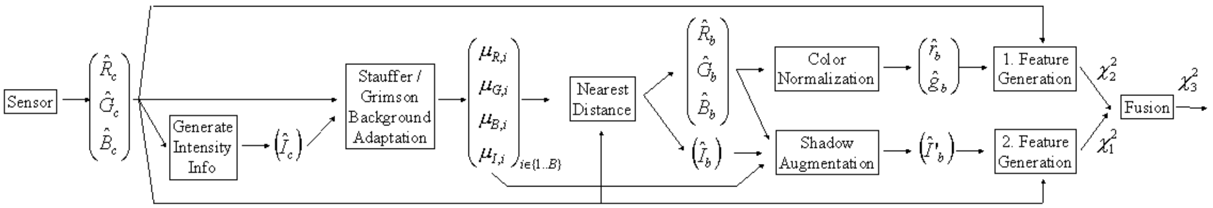
Sensor
$\begin{pmatrix}\hat{R}_c\\ \hat{G}_c\\ \hat{B}_c\end{pmatrix}$
Generate Intensity Info
$(\hat{I}_c)$
Stauffer / Grimson Background Adaptation
$\begin{pmatrix}\mu_{R,i}\\ \mu_{G,i}\\ \mu_{B,i}\\ \mu_{I,i}\end{pmatrix}_{i\in\{1..B\}}$
Nearest Distance
$\begin{pmatrix}\hat{R}_b\\ \hat{G}_b\\ \hat{B}_b\end{pmatrix}$
$(\hat{I}_b)$
Color Normalization
Shadow Augmentation
$\begin{pmatrix}\hat{r}_b\\ \hat{g}_b\end{pmatrix}$
$(\hat{I}'_b)$
1. Feature Generation
2. Feature Generation
χ_2^2
χ_1^2
Fusion
χ_3^2

Figure 5.3: Fusion of SG and Shadow/Gain invariant change detection modules. The output statistic after fusion is chi-square distributed as per our requirement for the systems integration.

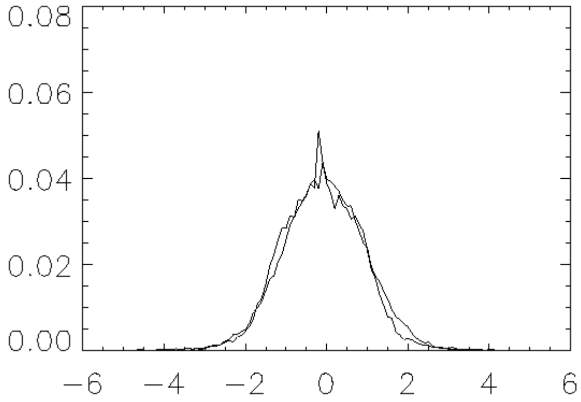

Figure 5.4: Difference between the current measurement and the closest background mode: True measurements overlayed by the Gaussian approximation.

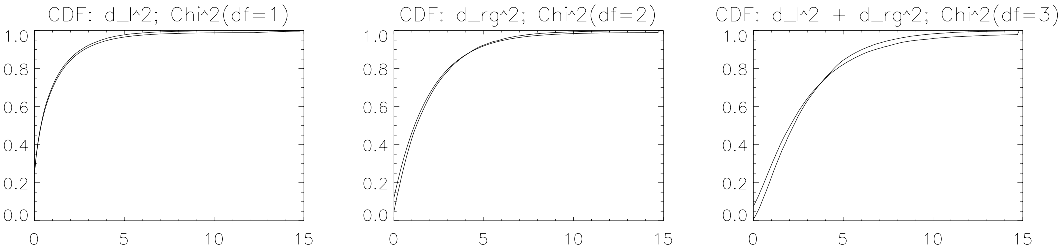

Figure 5.5: Cumulative densities for distances when there is no object in the scene; system output is overlaid by theoretical values. Left to right: Intensity band. Normalized color band after fusion with background adaptation module. Combined distance measure after fusion of intensity and normalized color distance measure as used for input in subsequent modules.

5.5 Experiments and Validation

We have verified that new modifications to the system result in output statistics that are the same as that required by other components in our original system. Therefore, the systems analysis for the original system remains untouched and there is no need to re-do the theoretical analysis and validation experiments. However, we do need to check whether the approximation error introduced in the new module affect the final performance of the system and verify this in real experiments.

Again, our real experiments follow a similar protocol as in section 4.4. The correctness of the pan, tilt, and zoom parameters estimated by our modules are compared against ground truth values of these control parameters to estimate the mean and variances of the running system. First, we marked eight positions $P1 - P8$ of different radial distances and pan angles as shown in Figure 5.6. Positions, and test persons were chosen to simulate different positions, illumination, and contrast. In the following table, we show the final foveal camera control parameters for one person. Ground truth values for the mean values were taken by measuring tilt angle α, and pan angle β by hand, and are compared against the corresponding mean of system measurements estimated from 100 trials per position and person. The variances calculated by the system for pan and tilt angles are compared against the corresponding variance-estimates calculated based on the theoretical analysis. The comparison between system output and ground truth demonstrates the closeness between theory and experiment. Please see Table 5.4 for tilt angles and Table 5.5 for pan angles.

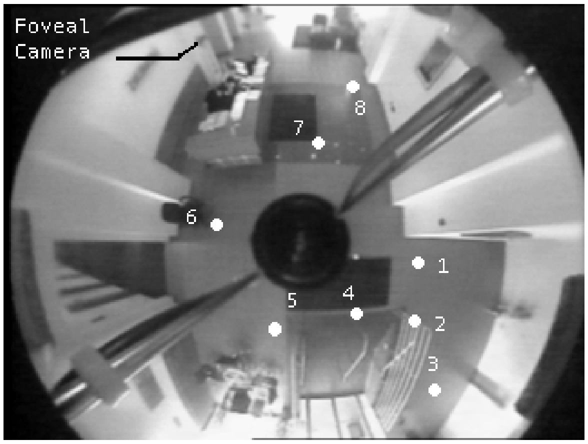

Figure 5.6: . Positions P1–P8, corresponding to measurements in Table 5.4 for tilt angles and in Table 5.5 for pan angles.

We repeat the experiment from section 4.4.2 to validate the zooming setting for the extended system. Confidence percentile α_z was again set to 95%. For 100 arbitrary positions in the room the foveal images were manually classified into two groups.

Table 5.4: Validation of tilt control for new system at positions P1–P8. First line shows experimental variance $\tilde{\sigma}^2_{\tan\hat{\alpha}}$ for $\tan\hat{\alpha}$, second line for the predicted equivalent $\hat{\sigma}^2_{\tan\hat{\alpha}}$. third line provides the corresponding standard deviation of the prediction.

$\times 10^{-6}$	P1	P2	P3	P4	P5	P6	P7	P8
$\tilde{\sigma}^2_{\sin\hat{\beta}}$	2.02	2.03	1.50	1.73	10.1	12.6	1.97	1.67
$\hat{\sigma}^2_{\sin\hat{\beta}}$	2.02	2.06	1.29	1.73	9.99	12.5	1.91	1.60
$\sigma_{\hat{\sigma}^2_{\sin\hat{\beta}}}$	0.36	0.27	0.21	0.37	2.28	10.5	0.40	0.24

Table 5.5: Validation of pan control for new system at positions P1–P8. First line shows experimental variance $\tilde{\sigma}^2_{\sin\hat{\beta}}$ for $\sin\hat{\beta}$, second line for the predicted equivalent $\hat{\sigma}^2_{\sin\hat{\beta}}$. third line provides the corresponding standard deviation of the prediction.

$\times 10^{-4}$	P1	P2	P3	P4	P5	P6	P7	P8
$\tilde{\sigma}^2_{\tan\hat{\alpha}}$	3.72	5.13	6.01	8.27	14.7	26.9	2.27	2.44
$\hat{\sigma}^2_{\tan\hat{\alpha}}$	2.93	5.11	13.9	8.12	18.3	20.4	2.24	2.10
$\sigma_{\hat{\sigma}^2_{\tan\hat{\alpha}}}$	0.48	0.95	5.29	2.20	9.81	5.82	1.04	0.26

- Group A: “Entire head visible; no part of the head cut off.”
- Group B: “Rest.”

In 90 of the 100 trials, assignment for *group A* was made by the system. The trials included having the person stand at several locations and wearing different clothing to simulate various contrasts and sizes.

Results of people detection and zooming under various conditions including day, night, day to night transitions, low contrast between object and background, and various positions of the object are demonstrated in Figures 5.7- 5.9. Figures 5.10- 5.12 demonstrate how the system zooms out in occlusion situations, when the prior model of a projected person does not match the data and therefore the uncertainty in the estimates increases. They clearly show the robustness of the people detection and zooming system. The camera control parameters as well as the zoom parameter are functions of the geometry, as well as of the current uncertainties in the position estimates of the person. The more certain the estimate the more we can afford to zoom in. The uncertainties are functions of the current scene, quality of segmentation, geometry, and calibration uncertainties.

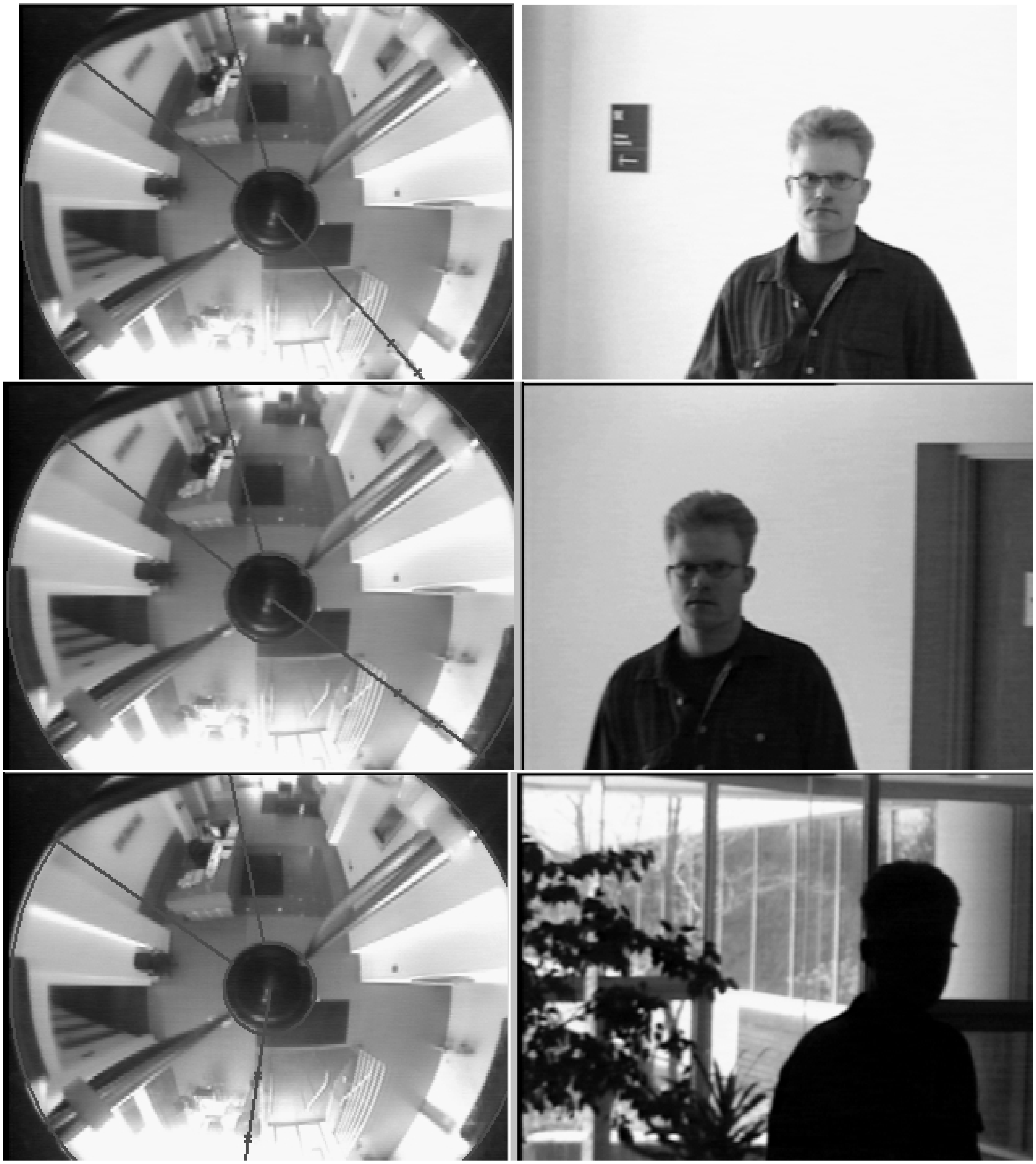

Figure 5.7: New System: Performance during day: Lines indicate angular position of person and crosses indicate foot and estimated head positions, top to bottom: a) Object far away, partially saturated background, b) Object closer, c) Object closer, partially saturated background.

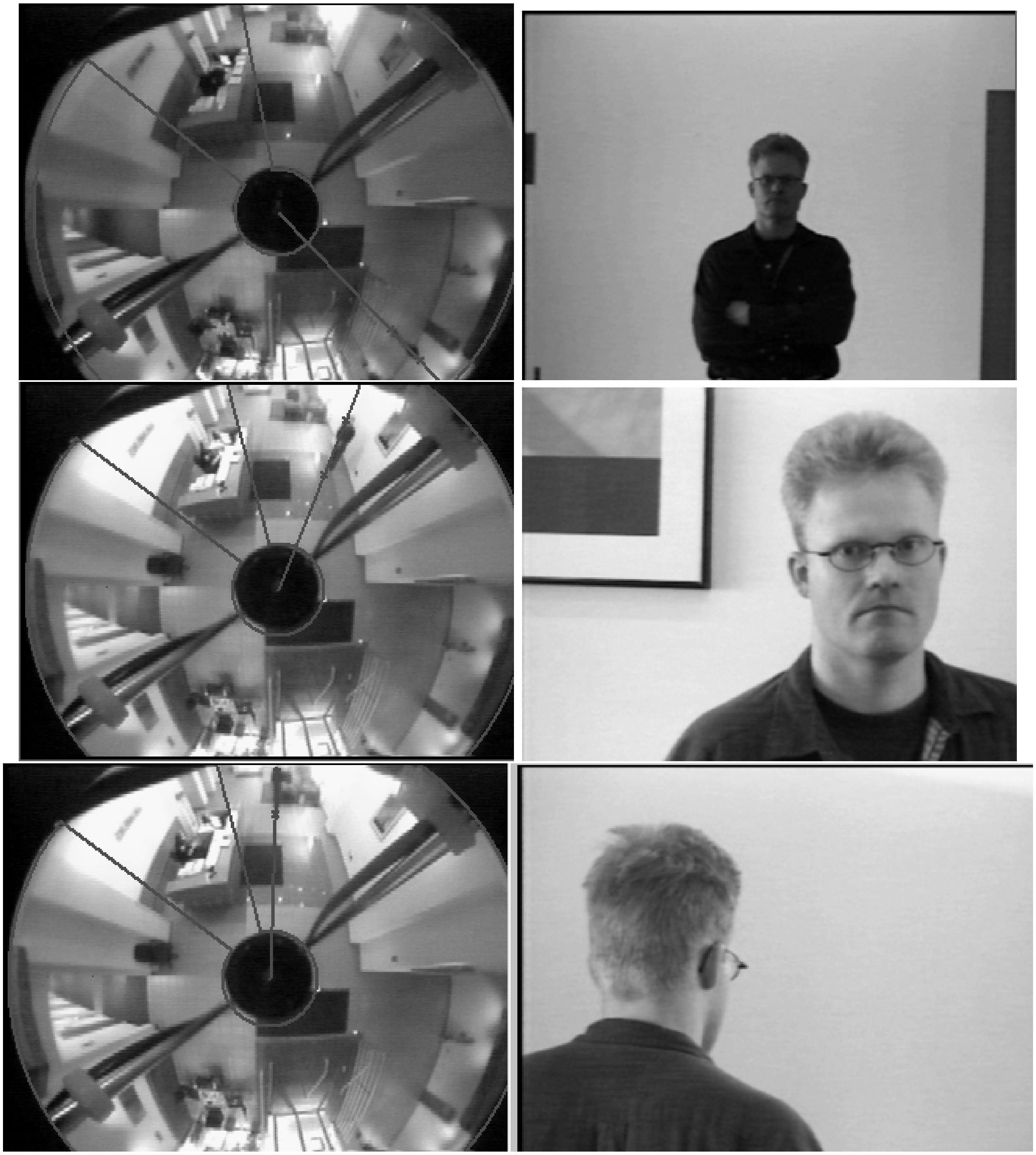

Figure 5.8: New System: Performance during transition from day to night illumination: Lines indicate angular position of person and crosses indicate foot and estimated head positions, top to bottom: a) Object far away, b) Object closer, very precise and reliable foot position estimation c) Object far away from omni-camera center.

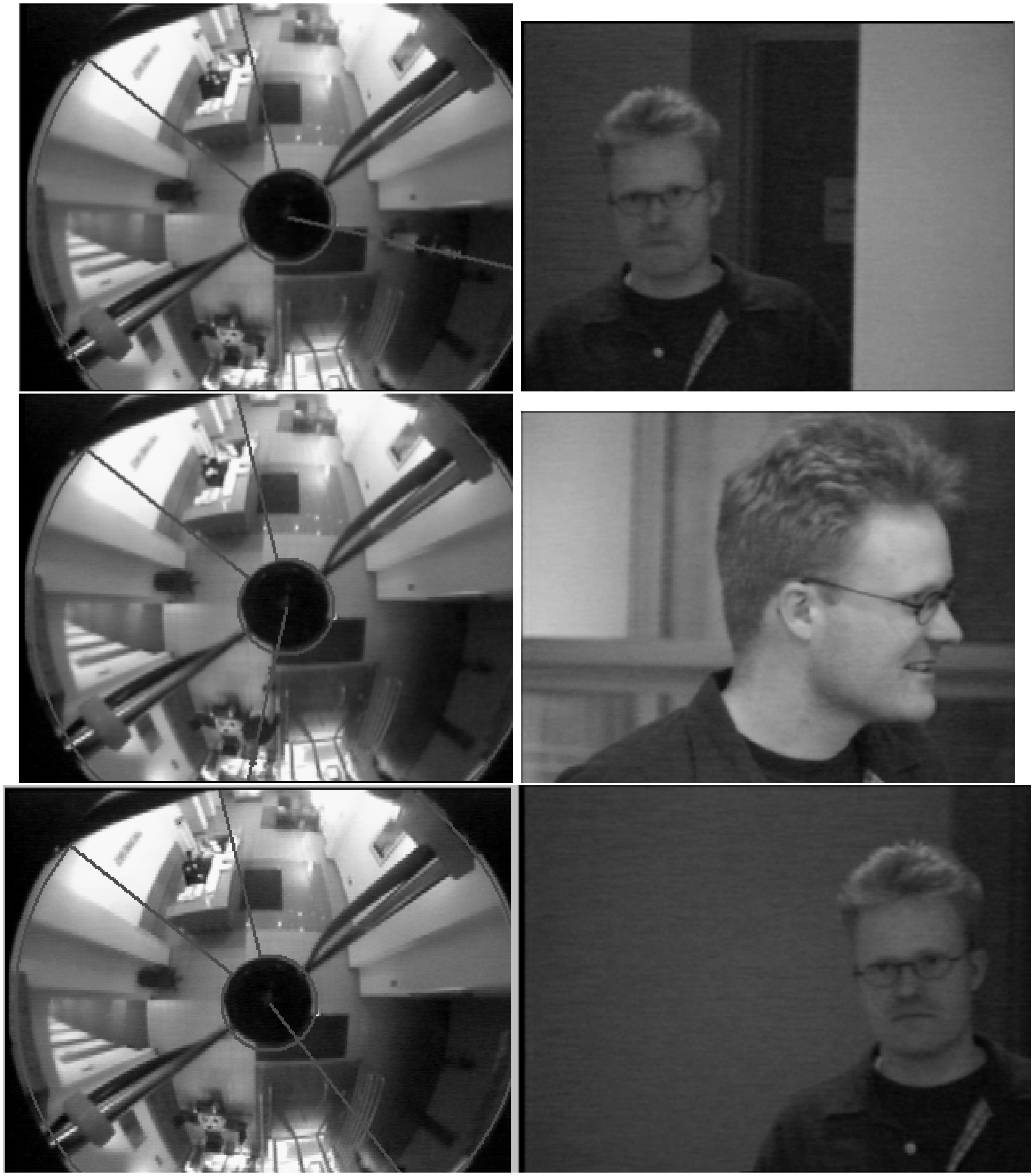

Figure 5.9: New System: Performance during night: Lines indicate angular position of person and crosses indicate foot and estimated head positions, top to bottom: a) Object in front of dark background, partially low contrast, b) Precise and reliable foot position estimation, zoomed in close, c) Object far away, low contrast, zoomed further out.

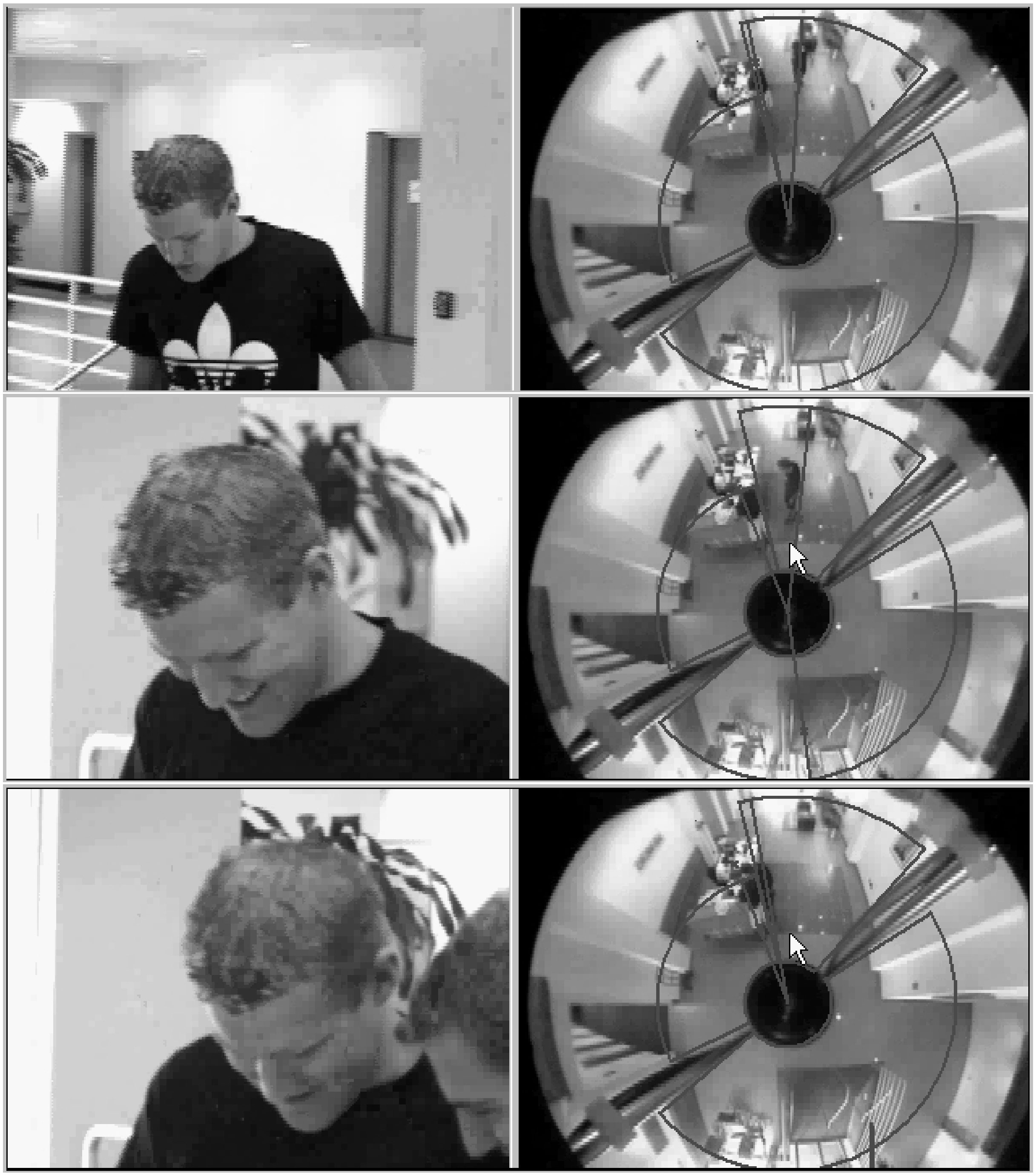

Figure 5.10: Zooming during occlusion: Lines indicates angular position of person and crosses indicate foot positions. An additional cross shows the head position of the person being tracked by the foveal image. Three active sectors. Top to bottom: a) Person 1 being tracked registers with guard, person 2 enters scene , b) Person 1 tracked by foveal camera, person 2 and entering person 3 only tracked in omni view c) Person 1 still tracked solely, person 2 close but still separated, person 3 left scene.

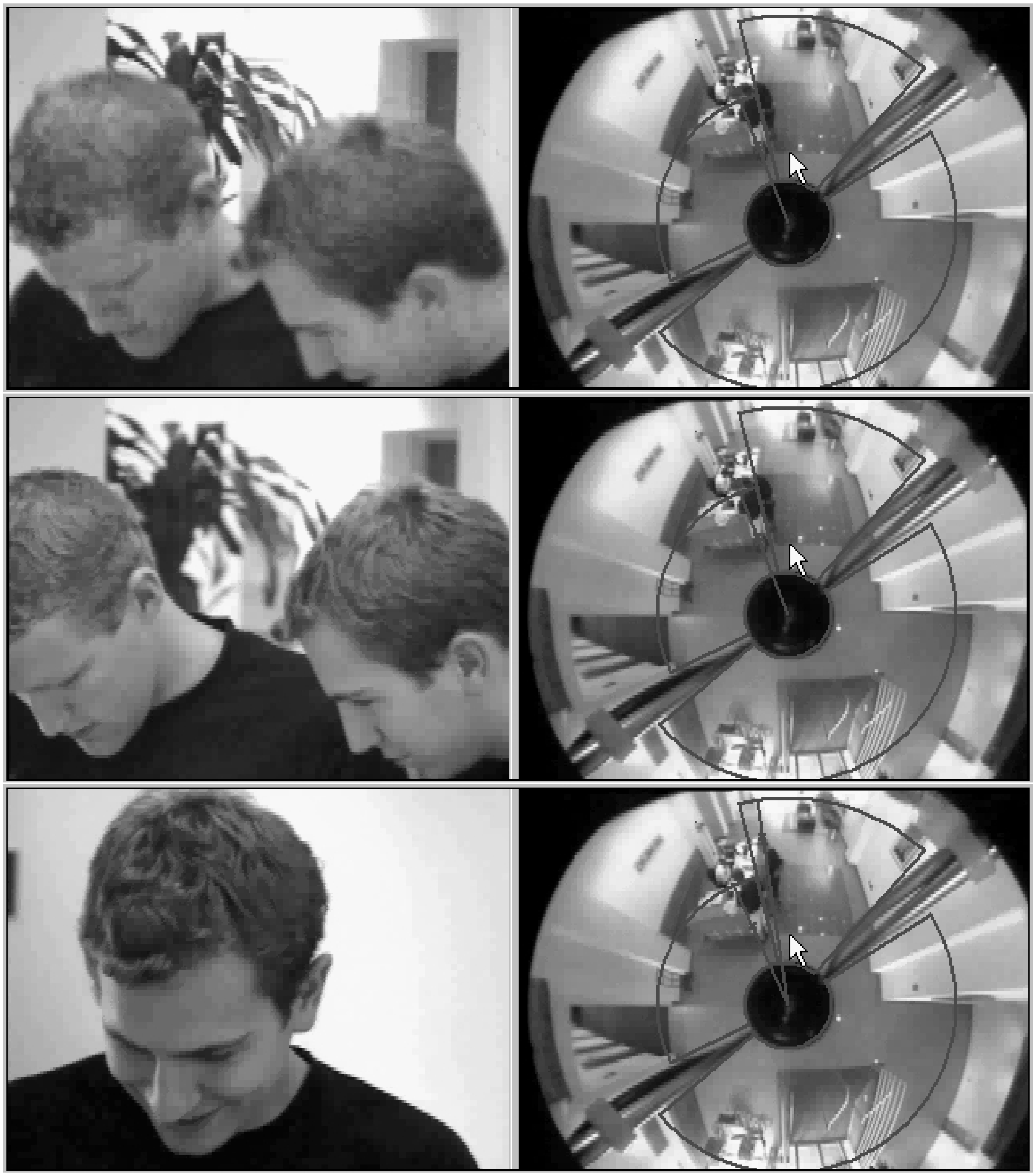

Figure 5.11: Zooming during occlusion: Lines indicates angular position of person and crosses indicate foot positions. An additional cross shows the head position of the person being tracked by the foveal image. Three active sectors. Top to bottom: a,b) Person 1 and 2 are occluded, system interprets this as one person that does not closely match the prior on expected projection length. Therefore, the position estimation becomes unreliable and the system zooms that much out that all possible foot locations are covered and both person's heads are captured by the foveal camera. c) Occlusion ended, 2 different objects detected, focus back on one person, zoomed in. Unfortunately, the focus changed erroneously from person 1 to person 2.

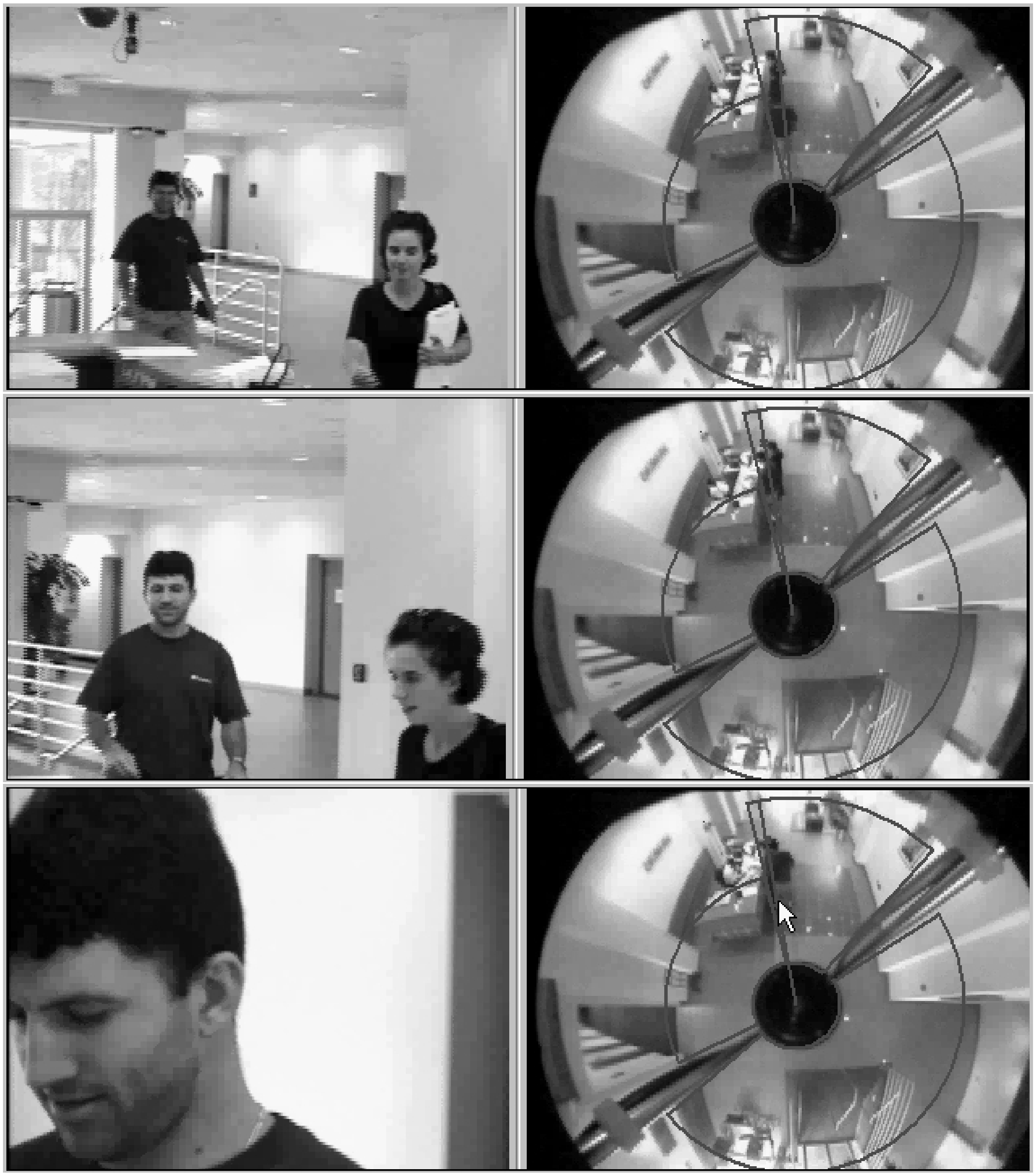

Figure 5.12: Zooming during occlusion: Lines indicates angular position of person and crosses indicate foot positions. An additional cross shows the head position of the person being tracked by the foveal image. Three active sectors. Top to bottom: a,b) Two people occluded along same radial line, high foot position uncertainty, since model assumption for projection does not match closely, system zooms out, captures both persons simultaneously. c) Person 1 occludes person 2 entirely in the omni-view, precise zoom onto person 1.

Chapter 6

Results and Insights

In the following sections the results obtained as well as the insights gained from the analysis and experiments are presented and discussed.

The essence of the message is that by carefully decomposing the global task into sub-pieces, by statistical characterizing the system, and by incorporating application specific priors in various stages of the system, it is possible to build a computationally efficient, but yet statistically well motivated system.

The main two result under the system engineering aspect is that by carefully decomposing the global task into sub-pieces, by statistical characterizing the system, and by incorporating application specific priors in various stages of the system, it is possible to build a computationally efficient, but yet statistically well motivated system. Following these systematic engineering principles rigorously, one can minimize re-design and analysis efforts required when extend functionality of a vision system. The key conclusion is that by choosing appropriate modules and suitable statistical representations, we are able to re-use existing system design, software, and performance analysis results.

From the application point of view, we obtained the following results. The system operates reliable during day and night operations. The final system is installed in an office and an office-building lobby that is lit by a mix of natural and artificial light during the day, while during night it is only lit by artificial light. Tests showed that the system successfully handles camera gain changes and shadows, as well as dynamic illumination changes.

In terms of zooming, we set the probability that the head is entirely contained in the foveal frame to $\alpha_Z = 0.95$. Experiments confirmed in 90 out of 100 trials that the entire head was contained in the foveal view .

6.1 Quantitative Results

The experiments demonstrate that the zoom parameter estimated by the system is a function of three factors: the contrast, the angle δ (corresponding to the relative position of the object between center of the foveal camera and the omni camera center), as well as the radial distance between the object and the omni-cam center, R_p. For low, medium, and moderate signal to noise ratios (specified by the median contrast measure between object and background), the zoom factor is a function of δ, while for high contrast the zoom factor is mainly a function of the distance R_p. Finally, the zoom factor is directly related to the median signal to noise ratio, i.e. the higher the signal to noise ratio the larger the zoom factor. Table 6.1 illustrates for the original system the face sizes in pixels obtained under various operation conditions depicted in Figures 4.17 through 4.22. While the new system is able to operate on a much larger range of settings (see Figures 5.7-5.9) and in a much less restricted environment, the zooming results remain stable given comparable segmentation. Of course, in regions where the segmentation is improved such that it affects the reliability in the foot position estimation, the improved system zooms in closer.

It illustrates the number of face pixels in the foveal frame as a function of the segmentation quality (specified by the median contrast measure between object and background), and the angle $\delta = 180° - \beta - \vartheta$ (see Figure 4.3). The alphanumeric entry after the face size corresponds to the images in Figure 4.17 through 4.22: *C* for conference room, *O* for office sequence, numbers top down. Note, that the maximal number of face pixels is approximately 160×240; due to the fact that a face has inverse aspect ratio relative to the image frame (portrait (face) vs. landscape (image)).

Figure 6.1 illustrates the *theoretical* values for the percentage of pixels in the foveal frame being covered by the face for a given signal to noise ratio of > 3 and assumed uncertainty of $\sigma_{r_f} = 3$ pixel at any pixel location. Note that the outer circle maps to infinity in 3D. For an area of approximately $220m^2$, the ratio of face pixel area to total image pixel area is $> 1:10$. (e.g. 90 x 90 pixels in a 320 x 240 image). The center image clearly shows the relative geometry influencing the results: For positions along the line passing through both cameras and the person location simultaneously the uncertainties in the pan estimate is minimal ant therefor zooming in further is possible[1].

Table 6.1 summarizes the results from *real* experiments. Here, the zooming is a function of the *actual* segmentation quality and therefor neither a fixed signal to noise ratio nor a fixed uncertainty in the foot position estimation is guaranteed for any position equally. We see that the experimental results match the expectation: In areas of low segmentation quality ($< 50\%$) the zoom setting is rather low and a function of angle δ. For segmentation

[1] Please note, that this overweighs the influence that the distance between the two cameras has.

Table 6.1: Number of face pixels in foveal frame as function of segmentation quality (specified by the median contrast measure between object and background), and angle $\delta = 180° - \beta - \vartheta$ (see Figure 4.3). Second entry (alphanumeric) refers to scenarios such as in Figure 4.19 through 4.20: C for conference room, O for office sequence, numbers top down. Note, that the maximal number of face pixels is approximately 160 × 240; due to the fact that a face has inverse aspect ratio relative to the image frame (portrait (face) vs. landscape (image)).

δ	low	medium	moderate	high
5		52 × 80(C2)		156 × 218(C7)
10	38 × 60(O5)			97 × 147(C5)
15		45 × 73(O4)		
20		37 × 58(O2)		
25	15 × 22(O7)			
35			89 × 138(O3)	96 × 145(C4)
45			82 × 130(O1)	127 × 189(C1)
50			74 × 109(C3)	97 × 153(O6)
55			67 × 87(C6)	

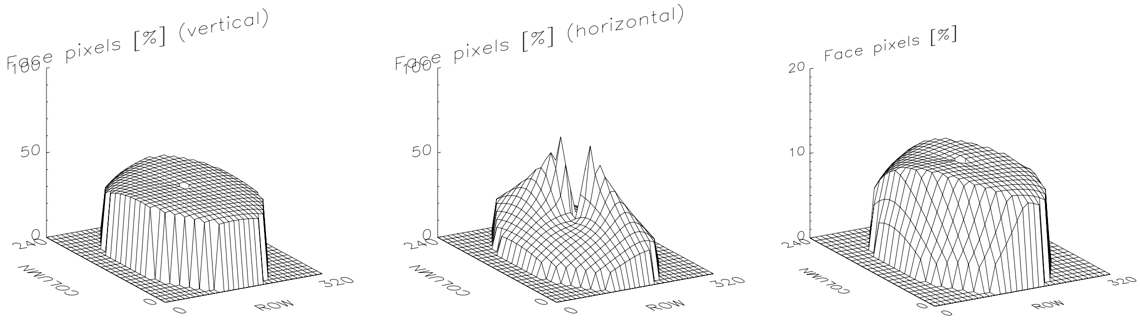

Figure 6.1: Influence of foot position in the space on the percentage of pixels in the foveal frame being covered by a face. For demonstration purpose fixed signal to noise ratio > 3), uncertainty of $\sigma_{r_f} = 3$ pixels at any position (in reality, smaller at most positions). Ratio of 10% corresponds to 90 x 90 pixels in a 320 x 240 image. Left: Horizontal. Center: Vertical. Right: Entire image.

results of > 80% one can not find the zoom being a function of δ; it rather seems to be a function of r_f respectively R_p. This is anticipated, since for good segmentation results, the uncertainty in the foot position estimation $\sigma^2_{r_f}$ becomes small, while we know from equation (4.68) that for large r_f uncertainty in 3D value R_p grow. This is due to the character of the transformation equation (4.6), where for small r_f one pixel maps to a few centimeter, while for large r_f one pixel can map to multiple decimeters. Please note, that even very small values for the uncertainty in the angle estimates may at a large distance R_p not necessarily map to high amounts of head pixels in the foveal frame: due to geometry, the amount of head pixels in the frame is a function of the angle *times* the distance R_p .

This discussion shows that theoretical derivations alone do not buy anything, unless we get a handle on the uncertainties of the *current* estimates. Even though the geometry strongly influences the results, the final zooming quality equally has to be a function of the segmentation quality (encoded in foot position uncertainties), and is therefor a function of the current data. That exactly is the strength of the system: The uncertainties are calculated online from the actual data, and the parameters are set in real-time accordingly. In our case, the uncertainties are very much a function of the segmentation results and therefor not predictable in advance; we have to derive them from the actual data. These results, which we generated online by propagating the uncertainties at run time, helped us to zoom properly for the current situation. We have seen, that we can afford to zoom in even further where segmentation results improve due to improved signal to noise ratio in the current signal. Some Results obtained in the experiments exceed the results shown in the plots, which were generated on the base of fixed values as described earlier. In cases of poorer signals, the results are worse, and we are able to zoom further out to guarantee performance in terms of ensuring that the head is still in the foveal frame.

6.2 Optimization and Customization of Setup

We now illustrate, how the statistical analysis is used to optimize the camera setup. The formulas 4.71, and 4.72 suggest that the configuration that minimizes these uncertainties is the one with large inter-camera distance D_c and foveal camera height H_f equal to the mean person eye-level height H_p. Figure 6.2 illustrates a comparison of the uncertainties in the pan angle for this setup (right plots), versus a camera position setup with lower distance D_c (left plots). In Figure 6.3 we compare the uncertainties in the tilt angle for this setup ($H_f \approx H_p$, right plots), versus a foveal camera mounting height of $H_f = 3.75m$(left plots).

Figures 6.2, and 6.3 illustrates how the setup of the system (here placement/mounting of foveal camera)influences precision globally and locally. Especially in Figure 6.2, note preferred zones with low uncertainties. During installation, these results can be used to adapt the system to have optimized operational performance in certain areas of the room. One can note that the plots for the uncertainties are truncated for (row,col) coordinates that are farther away from the omni-image center. These points represent areas where the uncertainties are beyond an acceptable threshold and hence the zoom parameter is never adapted.

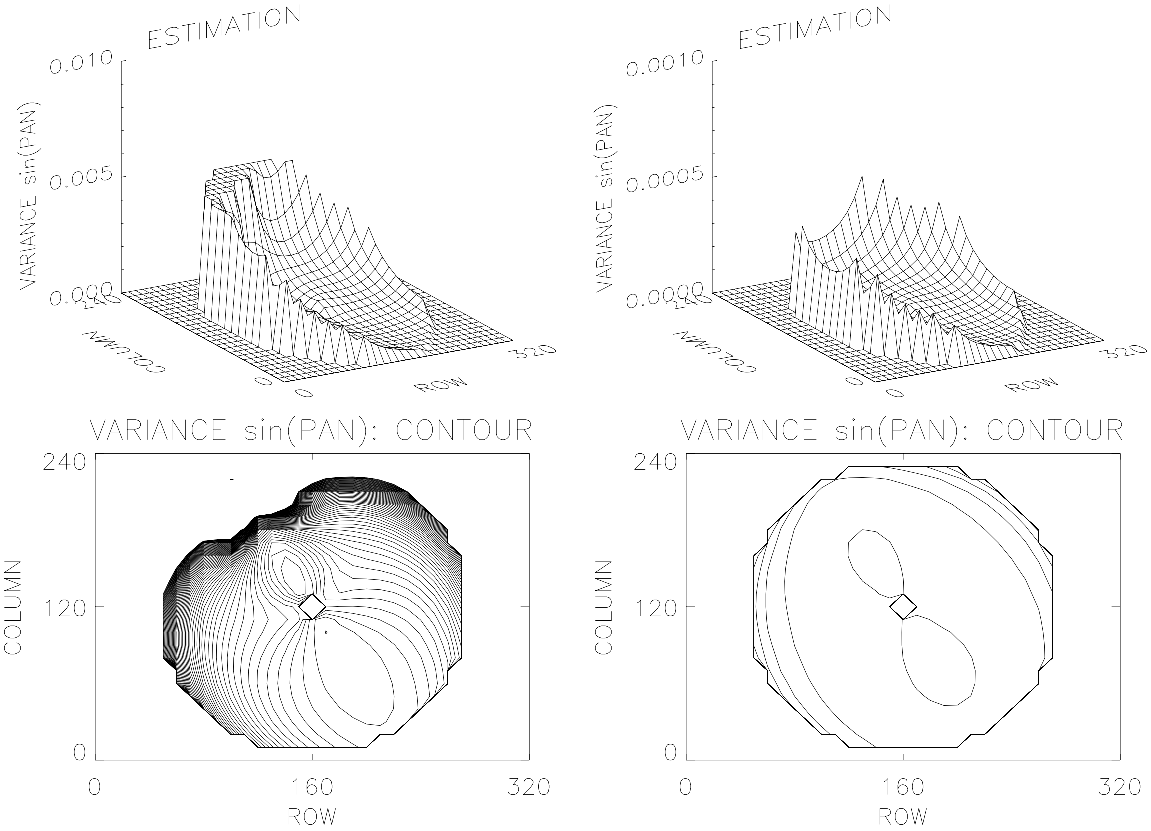

Figure 6.2: Influence of camera positioning on global and local performance. Top: variance $\sigma_{\hat{\sin\beta}}$ (pan). Bottom: corresponding contours. Left: close distance D_c between foveal camera and OmniCam. Right: larger distance, better performance.

6.3 Discussion

In this section we list and discuss some details and insights gained while engineering, testing and refining the current system. The following discussion is meant to provide starting points for further module and system improvements.

Discrimination: The current system performs reliable in indoor and quasi outdoor settings. However, since the pixel based miss detection rate is space variant in the normalized color space (indicated by spacial varying covariance matrix $\Sigma_{\hat{r},\hat{g}}$), it seems worthwhile to explore, how to weight the terms contributing to the accumulated feature representation accordingly.

Modelling Person Height $\mathbf{H}_p$ by a single Gaussian: As described earlier, we model a person's height $\hat{H}_p$ as normal distributed with mean H_p and variance $\sigma^2_{\hat{H}_p}$. Nevertheless, in section 4.2.4 where we estimate the radial foot location in the omni-image, we use the assumption that the variance of the height and size is small, and just fix H_p as constants

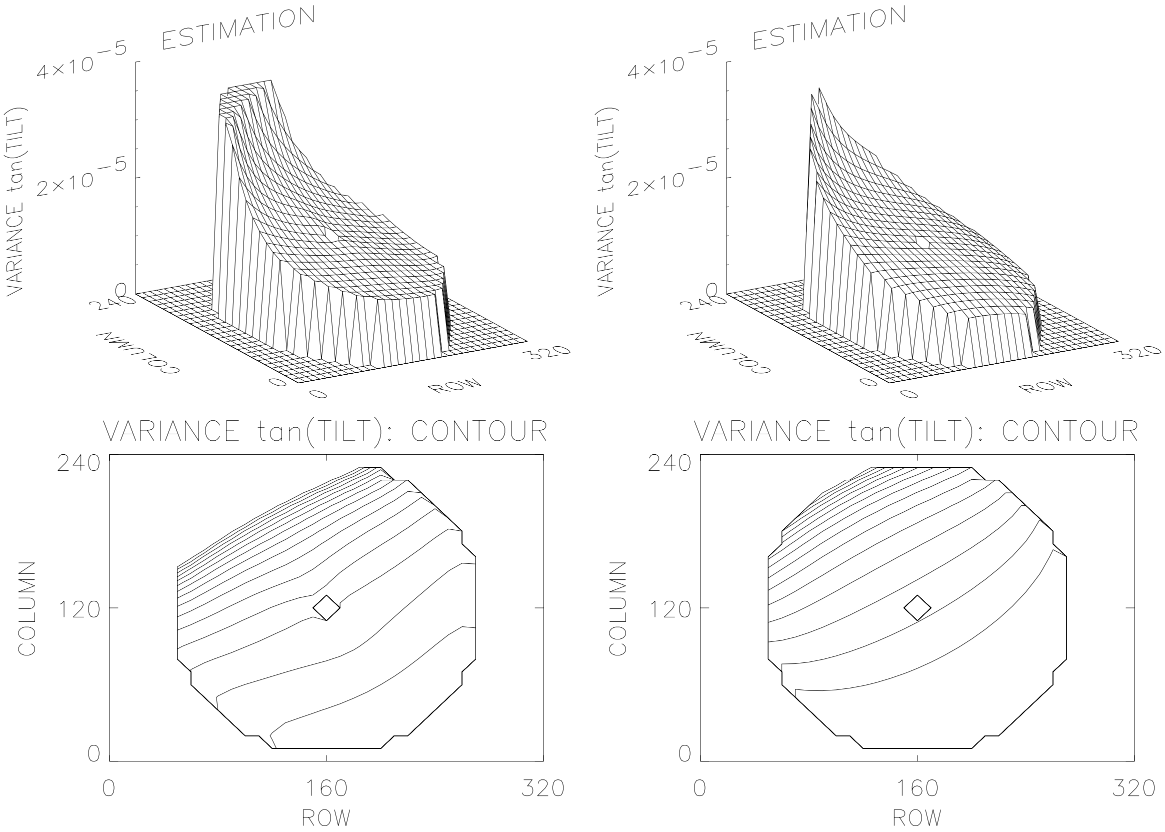

Figure 6.3: Influence of foveal camera mounting height on global and local performance. Top: variance $\sigma_{\widehat{\tan}\,\alpha}$ (tilt). Bottom: corresponding contours. Left: H_f approximately H_p. Right: Mounting height increased by 2m.

in equation (4.26). Though the current system is designed for persons of average height $1.75m$, extension to a multi-modal approach for groups of e.g. children, people of small, average, and tall size can be done within the proposed framework. Allowing for a different variance in the height distribution could adapt the system to application needs. Currently, a larger variance would result in more conservative zooming. Maintaining close zooming and allowing for different people heights (e.g. children, small, average and tall people) would make necessary a different prior model for $\hat{H}_p$, e.g. a mixture of Gaussian.

Occlusion: The system and the analysis needs to be further adapted to deal with scenarios where we have multiple persons along a given radial line along with occlusions. However, the current system would interpret occluded persons as a single person that does not match the prior assumption well. This will result in high uncertainty in the foot position estimation such that the foveal camera zooms out and captures both persons in the same frame (see Figures 5.10- 5.12).

Out of Focus Images: The system adapts to variations in camera setup like sub-optimal focusing and automatically account for the varying uncertainty in the measurements. An out of focus situation results in a larger uncertainty in the measurement allowing the system not to zoom in too close. We found that with a sharper focused setup, the zoom factors were consistently higher than with a blurred focus.

Radial Foot-Position Estimation: In the estimation step for the radial foot position, we replace the likelihood term for $p\,(\text{measurment}\ |\text{object})$ by the term $p\,(\text{measurment}\ |\text{hypothesis non-background})$, see equation (4.26). Since we know that the distribution $\hat{M}^{\top}_{r,\theta_f}$ for objects $p\,(\text{measurment}\ |\text{object})$ is $\chi^2_{2k'}(c^{\top}_{r,\theta})$ distributed with known degrees of freedom $2k'$ and known non-centrality parameter $c^{\top}_{r,\theta}$ (see section 4.3.5) an estimation based on this distribution had been even more precise, since it included knowledge of the object data itself. Nevertheless, it would be more time consuming in the parametric sampling process that tries to analyze the uncertainty in the estimate.

Tracking: Although we mainly discussed person detection and location estimation alone, the actual video surveillance system has tracking algorithms implemented. However precise tracking is only implemented for slowly moving persons. The current system uses temporal uncertainty in the location estimation to switch between adaptive zooming when the object moves slowly and zoomed-out mode if the person is moving arbitrarily in the scene. Please note the difference between temporal uncertainty estimated over a time window and uncertainty in the location estimation.

Interface to Higher Level Algorithms: The fact that our approach does not only provide best estimates but along with it also uncertainties in these estimates, the out put can be used as input for higher level algorithms. For example, a face detection/recognition engine would know the expected size of the head and can initialize its filters and kernel sizes adaptively.

Illumination Adaptation: When we put the original system in a stationary environment illuminated by sunlight, we found that it would only operate successfully for approximately 20 minutes. This is the time, during which the sun travels 5°. The change in illumination caused a shift in the camera gain, large enough that the background learned during the initialization phase changed significantly and produced 100% false alarms. In cloudy weather conditions the system failed even sooner, since the underlying illumination model was also violated sooner. As described earlier, the final system has an additional modules implemented that models these effects such that the revised system combines advantages of the original and the added illumination module, while their individual limitations were compensated by each other.

Chapter 7

Summary

The objective in this work was to study the systematic engineering, design and test cycle while building a dual-camera video surveillance system for people detection and zooming.

There are two main contributions of this thesis:

- One contribution is the demonstration of a systematic design methodology for building a complete real-time video surveillance system.
- The other Contribution deals with the adaptation of the existing system to show how one can incrementally evolve the current system design to meet added requirements.

A system was developed, which goal it was to continuously provide a high resolution zoomed-in image of a persons head at any location of the monitored area. An omni-directional camera video is processed to detect people and to precisely control a high-resolution foveal camera, which has pan, tilt and zoom capabilities. The pan and tilt parameters of the foveal camera and its uncertainties are shown to be functions of the underlying geometry, lighting conditions, background color/contrast, relative position of the person with respect to both cameras as well as of sensor noise and calibration errors. The uncertainty in the estimates is used to adaptively estimate the zoom parameter that guarantees with a user specified probability α_Z that the detected person's face is completely contained in the image while zoomed in to the maximal extent. The higher the probability α_Z the more conservative the zoom factor would be. With α_Z set to 95% we achieved zooming results that in average provided foveal images that contained 80×115 face pixels out of 320×240 in an entrance lobby area of about $400m^2$. Experiments confirmed in 92 out of 100 trials for the original system and in 90 out of 100 trials for the extended system that the entire head was contained in the foveal view.

It was shown how application specific constraints impact the choice of the system configuration. The process of making the right choice of feature representations is still an art and not a science. However, it was demonstrated that once a system configuration has been chosen it is possible to analyze the system behavior and quantify its performance relative to the application at hand. Further research is needed to understand what feature representations are appropriate for a given task and to identify how the representation correlates with application specific priors. The work demonstrated how by careful statistical modelling it is possible to develop and quantify a visual surveillance system. The essence of the message is that by carefully decomposing the global task into sub-pieces, by statistical characterizing the system, and by incorporating application specific priors in various stages of the system, it is possible to build a computationally efficient, but yet statistically well motivated system.

To present this essence, it was necessary to make certain simplifying prior assumptions and illustrate a working system in a constraint environment. Following a systematic methodology during the design and analysis phases we were able to relax the constraints in a second development cycle/phase.

The second point we wish to make is that by following these rigorous systematic engineering principles one can minimize re-design and analysis efforts required to extend functionality of a vision system. The key conclusion is that by choosing appropriate modules and suitable statistical representations, we are able to re-use existing system design, software, and performance analysis results. A new change detection algorithm fusing two different change detection algorithms was devised. One dealt with camera gain changes and shadows, while another dealt with dynamic illumination changes. The strengths of both these algorithms were retained, while their individual limitations were compensated by each other. The integration was done by paying attention to how the change detection component interfaces with the rest of the existing system.

Extensive amount of real and synthetic data experiments was used to validate the models derived. The new system was successfully tested in a conference room, in an office and an office-building lobby that was lit by a mix of natural and artificial light during the day, while during night it was only lit by artificial light. It operated successfully during day and night operations. The system proved robust when tested under a variety of situations without modification of system parameters: It was able to deal with various backgrounds, shadows, camera gain changes, dark and saturated measurements, and varying illumination conditions. The system high sensitivity was achieved in detection while retaining precise and data-driven adaptive zooming of the person head. The system could adapt to variations in camera setup like sub-optimal focusing/blurring, and automatically account for the varying uncertainty in the measurements. For example: With a sharper focused setup, the zoom factors were consistently higher than with a blurred focus.

The analysis and the experiments point out the following:

- The statistical analysis enable us to optimize the system setup to obtain minimal variance in the control parameters over a large area.

- The system control parameters can be derived online as a function of the current data measurements and used to setup the zoom parameter adaptively.

- The amount of zoom is a function of median contrast between the object and the background (as well as the form of the change detection measure profile along the radial line), the relative positioning of the person with respect to the omni-camera and the foveal camera, and the distance of the person from the omni-camera.

- The original system is real-time and operates at approximately 10 frames per second on a Pentium III, 600MHz CPU, with prototype code being non-optimized. The new system is significantly slower. However, a software analysis tool revealed that this is primarily due to inefficient coding. After recoding, we expect similar performance, since the performance is mainly restricted by the slow frame-grabberA.4 with a frame-rate of maximal 10 frames per second, when operating on a 320 × 240 RGB image, 8 bit color depth.

- Without using intensity information the detection step in the original system failed when the background area is completely black or saturated, due to the nature of the illumination invariant used in the normalized color change detection step. The combination of two different background adaptation and change detection methods eliminates this problem and allows for robust operation indoor as well as in quasi-outdoor settings, where the light conditions vary from natural sunlight through a mix of natural and artificial light to pure artificial light conditions.

- The probability of capturing the entire head in the foveal frame is user defined, while the system performance is determined by the number of head pixels in the foveal image. The higher this number, the better the system performs. Given that the aspect ratio for a head and for the foveal image are approximately inverse[1], approximately maximal 50% of the pixels in x-direction can be head pixels, which corresponds to a maximal head pixel region of approximately 160 × 240 pixels for a 320 × 240 foveal image. Typical numbers for the static background case can be found in Table 6.1. We set the probability of capturing the entire head in the foveal view to $\alpha_Z = 95\%$. For a 320×240 foveal image, the size of head pixel regions range from 15×22 for low segmentation quality to 156×218 for high segmentation quality, averaging overt time and location approximately 80 × 115 pixels. This is equivalent to filling approximately half of the y-axis, and quarter of the x-axis in the foveal view with head-pixels. While the new system is able to operate on a much larger range of settings and in a much less restricted environment, the zooming results remain

[1] depending on each persons individual head dimensions

stable given comparable segmentation. Of course, in regions where the segmentation is improved such that it affects the reliability in the foot position estimation, the improved system zooms in closer. We repeated the experiments for the extended system and received similar results: In 90 out of 100 trials the entire head was contained in the foveal view.

While this are partially qualitative statements, the analysis provides quantitative tables and plots that quantify the operating limits of the system.

Chapter 8

Outlook

In the following an outlook on future work is provided. Although, in the work presented, we mainly discussed person detection and location estimation alone, the actual video surveillance system has tracking algorithms implemented. However, an analysis of the tracking module and an analysis of the active camera dynamics is not provided. Systematic characterization of the tracking algorithm is a subject of further research. A proper analysis of the motion prediction module and camera dynamics would allow to also zoom in to the maximal extent while the person is moving. The current system uses temporal uncertainty in the location estimation to switch between adaptive zooming when the object stops moving and zoomed-out mode if the person is moving arbitrarily in the scene.

Furthermore, the system and the analysis needs to be adapted to deal with scenarios where we have multiple persons along a given radial line along with occlusions. The framework allows for straight forward modification of the prior used. However, the current system would interpret occluded persons as a single person and zoom out such that both persons are captured by the same foveal camera frame.

To further improve precision and resolve ambiguities, future research will deal with evaluating the foveal image and feeding results back to the system. Since the current system uses only priors on the geometry of persons, investigation in the foveal frame could serve in an verification step to reject detection of moving objects that are not people. This approach can be combined with a setting that uses multiple properly positioned foveal cameras, such that the system guarantees to provide face images and not only images of the head[1].

While the system tracks multiple people simultaneously in the omni-directional view, only one person can be tracked by the single foveal camera. Future research will introduce multiple foveal cameras and address the issue of sophisticated control strategies such that optimal zooming can be achieved for all persons in the scene.

[1] Which might be captured from behind.

Finally, future research topics may address how applications as face recognition engines can benefit from the quantitative performance measures our system provides along with its output.

Bibliography

[1] D. A. Adjeroh and M. C. Lee, “Robust and Efficient Transform Domain Video Sequence Analysis: An Approach from the Generalized Color Ratio Model”, in *Journal of Visual Communication and Image Representation* , volume 8, number 2, pages 182–207, June 1997.

[2] D. A. Adjeroh and M. C. Lee, “Robust and Efficient Transform Domain Video Sequence Analysis: An Approach from the Generalized Color Ratio Model”, in *Journal of Visual Communication and Image Representation* , volume 8, number 2, pages 182–207,June1997.

[3] P. Allen and R. Bajcsy, “Two sensors are better than one: example of vision and touch”, in *Proceedings of 3rd International Symposium on Robotics Research*, pages 48–55, Gouvieux, France, 1986.

[4] Y. Amit and D. Geman, “Shape quantization and recognition with randomized trees”, *Neural Computation*, Vol. 9, pages 1545–1588, 1997.

[5] Y. Amit and D. Geman, “A computational model for visual selection”, *Neural Computation*, 1999.

[6] S. Ayer, P. Schroeter, and J. Bigun, “Segmentation of moving objects by robust motion parameter estimation over multiple frames”, in *Computer Vision - ECCV*, vol. 2, (Stockholm), pages 316–327, May 1994.

[7] S. Baker and S. Nayar, “Global Measures of Coherence for Edge Detector Evaluation”, in *Proceedings of the IEEE CVPR*, Fort Collins, Vol. II, pages 373–379, 1999.

[8] Yaakov Bar-Shalom, Thomas E. Fortmann, *Tracking and Data Association, Mathematics in Science and Engineering*, Academic Press, London, Volume 179, 1998.

[9] M. Bichsel, “Illuminant Invariant Object Recognition”, in *Proceedings of IEEE International Conference on Image Processing* , volume 3, pages 620–623, October 1995.

[10] M. Bichsel, “Illumination Invariant Motion Segmentation of Simple Connected Objects”, University of Zurich, Dept. of Computer Science, Zurich, CH.

[11] S. Blostein and T. Huang, “Detecting small moving objects in image sequences using sequential hypothesis testing”, in *IEEE Transactions Signal. Processing*, vol. 39, no. 7, pages 1611–1629, 1991.

[12] M. Boshra and B. Bhanu, “Predicting Performance of Object Recognition”, in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, No. 9, pages 956–969, September 2000.

[13] T. Boult, A. Erkin, P. Lewis, R. Micheals, C. Power, C. Qian, and W. Yin, “Frame-rate multi-body tracking for surveillance”, in *Proceedings of the DARPA IUW*, 1998.

[14] T. E. Boult, R. Micheals, X. Gao, P. Lewis, C. Power, W. Yin, and A. Erkan, “Frame-rate omnidirectional surveillance and tracking of camouflaged and occluded targets”, in *Second IEEE International Workshop on Visual Surveillance*, pages 48–55, IEEE, 1999.

[15] T. E. Boult, R. Micheals, X. Gao, A. Erkan, W. Yin, and C. Power, “Omnidirectional frame-rate detection and tracking of camouflaged and occluded targets”, *Technical Report*, Lehigh, September 1999. (Submitted.)

[16] T. Boult, R. Micheals, X. Gao, M. Eckmann, “Into the woods: visual surveillance of non-cooperative and camouflaged targets in complex outdoor settings”, in *Proceedings of the IEEE, Special Issue on Video Surveillance*, 2001.

[17] Bowyer, K. W. and Phillips, P. J., editors, “Empirical Evaluation Techniques in Computer Vision”, *IEEE Press*, 1998.

[18] F. J. Canny, “Finding edges and lines in images”, *Tech.Rep. 720*, MIT AI Lab, June 1983.

[19] V. Chalana and Y. Kim, “A Methodology for Evaluation of Segmentation Algorithms on Medical Images”, in *Image Processing, SPIE Medical Imaging*, Vol. 2710, pages 178–189, 1996.

[20] S. E. Chen, “QuickTime VR - an Image Based Approach to Virtual Environment Navigation”, in *Computer Graphics: Proceedings of SIGGRAPH 95*, pages 29–38, August 1995.

[21] K. Cho, P. Meer, J. Cabrera, “Performance assessment through bootstrap”, in *IEEE Transactions in Pattern Analysis Machine Intelligence*, 19, pages 1185–1198, 1997.

[22] D. Comaniciu, V. Ramesh, P. Meer, “Real-Time Tracking of Non-Rigid Objects using Mean Shift”, *IEEE Conf. on Comp. Vis. and Pat. Rec.*, Hilton Head Island, South Carolina, Vol. 2, pages 142–149, 2000.

[23] P. Courtney, N. Thacker, A. Clark, “Algorithmic Modeling for Performance Evaluation”, *Special Issue on Performance Characterization, Machine Vision & Applications Journal*, Springer Verlag, 1998.

[24] L. Csink, D. Paulus, U. Ahlrichs, B. Heigl, "Color Normalization and Object Localization", in *4. Farbworkshop*, Koblenz, Germany, 1998.

[25] Y. Cui, S. Samarasekera, Q. Huang, M. Greiffenhagen, "Indoor Monitoring Via the Collaboration Between a Peripheral Sensor and a Foveal Sensor," *IEEE Workshop on Visual Surveillance*, Bombay, India, pages 2–9, 1998.

[26] R. Cutler and L. Davis, "Robust real-time periodic motion detection, analysis and applications", in *IEEE Transactions On Pattern Analysis and Machine Intelligence*, pages 781–796, August 2000.

[27] J. Davis and A. Bobick, "The representation and recognition of human movements using temporal templates", in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 928–934, 1997.

[28] J. Denzler, B. Heigl, H. Niemann, "An Efficient Combination of 2D and 3D Shape Descriptions for Contour Based Tracking of Moving Objects", in Proceedings of European Conference on Computer Vision , Freiburg, Germany, pages 843–856, 1998.

[29] B. A. Draper, "Learning Object Recognition Strategies", *Ph.D. Dissertation*, University of Massachusetts, 1993.

[30] M. - P. Dubuisson-Jolly, C. - C. Liang, and A. Gupta, "Optimal polyline tracking for artery motion compensation in coronary angiography", in *Proceedings International Conference on Computer Vision*, Bombay, India, pages 414–419.

[31] B. Efron, R. Tibishirani, "An Introduction to the Bootstrap", *Chapman & Hall*, New York, 1993.

[32] A. Elgammal, D. Harwood, and L. Davis, "Non-parametric model for background subtraction", in *FRAME-RATE Workshop*, IEEE, 1999. (Electronic (only) proceedings at www.eecs.lehigh.edu/FRAME.)

[33] G. D. Finlayson and S. S. Chatterjee and B. V. Funt, "Color Angular Indexing", in *Proceedings of European Conference on Computer Vision* , volume 2, pages 16–27, 1996.

[34] F. Fleuret and D. Geman, "Graded Learning for Object Detection", in *Proceedings of the IEEE Workshop on Statistical and Computational Theories in Vision*, Fort Collins, CO, June 1999, (Published on the Web).

[35] "Machine Vision & Applications", *International Journal, Special Issue on Performance Evaluation*, ed. by W. Forstner, Vol. 9, nos. 5/6, pages 229–239, 1997.

[36] N. Friedmann and S. Russell, "Image segmentation in video sequences: A probabilistic approach", Computer Science Division, University of California, Berkley, CA.

[37] X. Gao, T. Boult, F. Coetzee and V. Ramesh, "Error Analysis of Background Adaption", in *Proceedings of the IEEE CVPR*, Hilton Head, CA, pages 503–510, 2000.

[38] A. Gebhard and D. Paulus. "Active System to Generate Views of Facial Features with Selectable Resolution", in B. Girod, H. Niemann, and H.-P. Seidel, editors, *Vision Modeling and Visualization 99*, pages 179–186, Erlangen, November 1999.

[39] C. Goad, "Special Purpose Automatic Programming for 3D Model-Based Vision", in *DARPA IU Workshop Proceedings*, pages 94–104, 1983.

[40] M. Greiffenhagen and V. Ramesh, "Real-Time Video Analysis at Siemens Corporate Research: Systems Research and Statistical Performance Characterization", *Special Session on Advanced Video-Based Surveillance Systems, International Conference on Image Analysis and Processing, ICIAP*, Venice, September 1999.

Real-Time Video Analysis at Siemens Corporate Research Systems Research and Statistical Performance Characterization

[41] M. Greiffenhagen, V. Ramesh, D. Comaniciu, H. Niemann, "Statistical Modeling and Performance Characterization of a Real-Time Dual Camera Surveillance System", in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2000), Hilton Head Island, South Carolina, IEEE Computer Society (publisher)*, Volume 2, pages 335–342, June 13-15, 2000.

[42] M. Greiffenhagen, V. Ramesh, D. Comaniciu, H. Niemann, "Design, Analysis, and Engineering of Video Monitoring Systems: An Approach and a Case Study", in *Proceedings of the IEEE, Special Issue on Video Surveillance*, 2001.

[43] W. E. L. Grimson, D. Huttenlocher, "On the Sensitivity of Geometric Hashing", *IEEE International Conference on Computer Vision*, pages 334 –338, 1990.

[44] W. E. L. Grimson, D. Huttenlocher, "On the Verification of Hypothesized Matches in Model-Based Recognition", *Lecture notes in computer science*, 427, O. Faugeras (Ed.), Springer-Verlag, 1990.

[45] R. M. Haralick, "Computer Vision Theory: The Lack Thereof", *CVGIP 36*, pages 272–286, 1986.

[46] R. M. Haralick, "Performance assessment of near-perfect machines", *Machine Vision and Applications 2*, pages 1–16, 1989.

[47] R. Haralick, "Overview: Computer Vision Performance Characterization", *Proceedings of the DARPA Image Understand Workshop*, Vol.1, pages 663–665, 1994.

[48] G. Healey and D. Slater, "Using Illumination Invariant Color Histogram Descriptors for Recognition", in *Proceedings of International Journal Conference on Computer Vision* , Seattle, WA, pages 355–360, 1994.

[49] G. Healey and D. Slater, "Global color constancy: recognition of objects by use of illumination-invariant properties of color distributions", *J. Opt. Soc Am. A*, volume 11, number 11, pages 3003–3010, November 1994.

[50] G. Healey and L. Wang, "Illuminant-invariant recognition of texture in color images", *Optical Society of America*, volume 12, number 9, pages 1877–1883, September 1995.

[51] G. Healey and L. Wang, "Using Linear Models for the Illumination-Invariant Classification of Color Textures", *Proceedings of the IS&SID*, pages 123–125, 1995.

[52] G. Healey and D. Slater, "Computing Illumination-Invariant Descriptors of Spatially Filtered Color Image Regions", in *IEEE Transaction on Image Processing* , volume 6, number 7, pages 1002–1013, July 1997.

[53] "Dialogue: Performance Characterization in Computer Vision. With contributions from R. M. Haralick; L. Cinque, C. Guerra, S. Levialdi, J. Weng, T. S. Huang, P. Meer, Y. Shirai; B. A. Draper, J. R. Beveridge", *CVGIP: Image Understanding 60*, pages 245–265, 1994.

[54] R. Haralick, "Performance Characterization Protocol in Computer Vision", *ARPA IUW94 Proceedings*, Vol.1, pages 667–674.

[55] R. M. Haralick, "Propagating Covariance In Computer Vision", *IJPRAI, Vol. 10*, pages 561–572, 1996.

[56] I. Haritaoglu, D. Harwood, and L. Davis, "w^4: A real-time system for detecting and tracking people in 2.5d", in *European Conference on Computer Vision - ECCV*, 1998.

[57] I. Haritaoglu, D. Harwood, and L. Davis, "w^4: Real-time surveillance of people and their activities", *IEEE Transactions On Pattern Analysis and Machine Intelligence*, pages 809–830, August 2000.

[58] Heath, M. D., Sarkar, S., Sanocki, T., and Bowyer, K.W, "A Robust Visual Method for Assessing the Relative Performance of Edge Detection Algorithms", *PAMI (19)*, No. 12, pages 1338-1359, December 1997.

[59] J. Hong, "Image Based Homing", in *Proceedings of IEEE International Conference on Robotics and Automation*, May 1991.

[60] D. Huttenlocher, P. Noh, J. Jae, and J. William, "Tracking on non-rigid objects in complex scenes", in *International Conference on Computer Vision*, (Berlin), Sept. 1995.

[61] S. Huwer, H. Niemann, "2D-Object Tracking Based on Projection-Histograms", in Proceedings of European Conference on Computer Vision , Freiburg, Germany, pages 861–876, 1998.

[62] S. Huwer, H. Niemann, "Adaptive Change Detection for Real-Time Surveillance Applications", in *Third IEEE Workshop on Visual Surveillance* , Dublin, Ireland, pages 37–45, 2000.

[63] "Dialogue: Ignorance, Myopia, & Naivete in Computer Vision Systems, R. C. Jain and T. Binford, with contributions from: M.A.Snyder, Y.Aloimonos, A. Rosenfeld, T. S. Huang, K. W. Bowyer, and J. P. Jones", *CVGIP, Image Understanding*, Vol. 53, No. 1, Jan 1991.

[64] H. Joo, R. M. Haralick, and L. G. Shapiro, "Toward the Automating of Mathematical Morphology Procedures Using Predicate Logic", in *Proceedings of the ICCV*, pages 156–165, 1990.

[65] T. Kanade, R.T. Collins, A.J. Lipton, P.J. Burt, L. Wixson, "Advances in Cooperative Multi-Sensor Video Surveillance", *Proceedings of the DARPA Image Understanding Workshop*, Vol. 1, pages 3–24, 1998.

[66] Gudrun J. Klinger, Steven A. Shafer, Takeo Kanade, "A Physical Approach to Color Image Understanding", *International Journal of Computer Vision*, 4, pages 7–38, Kluwer Academic Publishers, 1990.

[67] D. Koller, K. Danilidis, and H. Nagel, "Model-based object tracking in monocular image sequences of road traffic scenes", *International Journal of Computer Vision*, vol. 10, no. 3, pages 257–281, 1993.

[68] S. Konishi, A.L. Yuille, J.M. Coughlan, S.C. Zhu, "Fundamental Bounds on Edge Detection: An Information Theoretic Evaluation of Different Edge Cues", in *Proceedings of the IEEE CVPR*, Fort Collins, Vol. I, pages 573–579, 1999.

[69] A. Krishnan and N. Ahuja, "Panoramic Image Acquisition", in *Proceedings of IEEE Conf. On Computer Vision and Pattern Recognition (CVPR-96)*, pages 379–384, June 1996.

[70] D. P. Kuban, H. L. Martin, S. D. Zimmermann, and N. Busico, "Omniview Motionless Camera Surveillance System", *United States Patent No. 5,359,363*, October 1994.

[71] M. C. Lee and D. A. Adjeroh, "Indexing and Retrieval in Visual Databases via Colour Ratio Histograms", *First International Conference on Visual Information Systems*, pages 309–316, February 1996.

[72] S. Lin and S. W. Lee, "Using Chromaticy Distributions and Eigenspace Analysis for Pose-, Illumination-, and Specularity-Invariant Recognition of 3D Objects", in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* , pages 426–431, June 1997.

[73] A. Lipton, H. Fuijiyoshi, and R. Patil, "Moving target detection and classification from real-time video", in *Proceedings Of the IEEE Workshop on Applications of Computer Vision*, 1998.

[74] W. Mann and T. Binford, "Probabilities for Bayesian Networks in Vision", In *Proceedings of the ARPA IU Workshop*, Vol. 1, pages 633–643, 1994.

[75] K.V.Mardia, Multivariate Analysis, Probability and Mathmatical Statistics, edited by Z.W.Birnbaum and E.Lukacs, Academic Press, 1995.

[76] J. Matas and R. Marik and J. Kittler, "Illumination Invariant Colour Recognition", University of Surrey, Dept. of Electronic and Electrical Engineering, Guildford, UK.

[77] L. McMillan and G. Bishop. "Plenoptic Modeling: An Image-Based Rendering System", in *Computer Graphics: Proceedings of SIGGRAPH 95*, pages 39–46, August 1995.

[78] J. W. V. Miller and M. Shidhar, "Hardware Considerations for Illumination-Invariant Image Processing", in *Proceedings of The Society of Photo-Optical Instrumentation Engineers* , volume 2347, pages 290–300, 1994.

[79] K. Miyamoto. "Fish Eye Lens", *Journal of Optical Society of America*, 54(8): pages 1060–1061, August 1964.

[80] D. Mumford, "Pattern theory: a unifying perspective", in *Perception as Bayesian Inference*, edited by D.Knill and W.Richards, Cambridge Univ. Press, 1996.

[81] V. Nalwa, "A True Omnidirectional Viewer", *Technical Report*, Bell Laboratories, Holmdel, NJ 07733, U.S.A., February 1996.

[82] S. K. Nayar and S. Baker, "Catadioptric Image Formation", in *Proceedings of DARPA Image Understanding Workshop*, May 1997.

[83] S. N. Nayar, "Catadioptric omnidirectional camera", in *Proceedings of the 1997 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 482–488, July 1997.

[84] S. Nayar, "Omnidirectional Video Camera", in *Proceedings of the DARPA Image Understanding Workshop*, Vol. 1, pages 235–242, 1997.

[85] S. Nayar, T. Boult, "Omnidirectional Vision systems: 1998 PI Report", *Proceedings of the DARPA Image Understanding Workshop*, Vol. 1, pages 93–100, 1998.

[86] J. Nielsen "Characterization of Vision Algorithms: An experimental Approach", *ECVnet Workshop on Benchmarking*, 1995.

[87] K. Ohba and Y. Sato and K. Ikeuchi, “Appearance Based Visual Learning and Object Recognition with Illumination Invariance”.

[88] S. J. Oh and E. L. Hall, “Guidance of a Mobile Robot using an Omnidirectional Vision Navigation System”, in *Proceedings of the Society of Photo-Optical Instrumentation Engineers, SPIE*, 852: pages 288–300, November 1987.

[89] A. Papoulis, “Probability, Random Variables, and Stochastic Processes”, McGraw–Hill, 1986.

[90] L. Parra, V. Ramesh, S. H. Lai, “Recovering Alignment Errors via EM Algorithm”, *Technical Report*, Siemens Corporate Research, Princeton, February 1998.

[91] D. Petkovic, “The Need for Accuracy Verification of Machine Vision Algorithms and Systems”, in *Proceedings of the CVPR*, pages 430–440, 1989.

[92] R. Polana and R. Nelson, “Low level recognition of human motion”, in *Workshop on Non-rigid Motion*, pages 77–82, November 1994.

[93] K. E. Price, “Anything you can do, I can do better (No you can’t)”, *CVGIP*, Vol. 36, No. 2/3, pages 387–391, 1986.

[94] V. Ramesh and R. M. Haralick, “Random Perturbation Models and Performance Characterization in Computer Vision”, in *Proceedings of the CVPR*, Champaign, IL, pages 521–527, 1992.

[95] V. Ramesh and R. M. Haralick, “A Methodology for Automatic Selection of IU Algorithm Tuning Parameters”, in *Proceedings of the ARPA IUW*, Vol. 1, pages 675-687, 1994.

[96] V. Ramesh, R.M. Haralick, X. Zhang, D.C. Nadadur, K. Thornton, “Automatic Selection of Tuning Parameters for Feature Extraction Sequences”, in *Proceedings of the CVPR*, Seattle, WA, pages 672–677, 1994.

[97] V. Ramesh, R.M. Haralick, A.S. Bedekar, X. Liu, D.C. Nadadur, K.B. Thornton, X. Zhang, “Computer Vision Performance Characterization,” *RADIUS: Image Understanding for Imagery Intelligence*, edited by. O. Firschein and T. Strat, Morgan Kaufmann Publishers, San Francisco, 1997.

[98] V. Ramesh and R. M. Haralick, “Random Perturbation Models for Boundary Extraction Sequence”, *Special Issue on Performance Characterization, Machine Vision & Applications Journal*, Springer Verlag, 1998.

[99] Y. Ricquebourg and P. Bouthemy, “Real-time tracking of moving persons by exploiting spatio-temporal image slices”, in *IEEE Transactions On Pattern Analysis and Machine Intelligence*, pages 797–808, August 2000.

[100] P. Rosin and T. Ellis, "Detecting and classifying intruders in image sequences", in *Proceedings Of British Machine Vision Conference*, pages 293–300, September 1991.

[101] S. Rowe and A. Blake, "Statistical background modeling for tracking with a virtual camera", in *Proceedings Of British Machine Vision Conference*, 1995. (Web version of a similar TR also available.)

[102] F. Sadjadi ed., "Performance Evaluation of Signal and Image Processing systems", *SPIE*, 1993.

[103] C. Shekhar, S. Moisan, R. Vincent, P. Burlina, R. Chellappa, "Knowledge-based Control of Vision Systems", *Image and Vision Computing*, Vol. 17, No. 9, pages 667–683, July 1999.

[104] M.C. Shin, D.B. Goldgof, K.W. Bowyer, "Comparison of Edge Detectors Using an Object Recognition task", in *Proceedings of the IEEE Computer Vision and Pattern Recognition Conference*, Fort Collins, Colorado, Vol. 1, pages 360–365, 1999.

[105] D. Slater and G. Healey, "Combining Color and Geometric Information for the Illumination Invariant Recognition of 3-D Objects", in *Proceedings of International Journal Conference on Computer Vision* , pages 563–568, June 1995.

[106] D. Slater and G. Healey, "The Illumination-Invariant Recognition o f3D Objects Using Local Color Invariants", in *IEEE Transactions on Pattern Analysis and Machine Intelligence* , volume 18, number 2, pages 206–210, February 1996.

[107] D. Slater and G. Healey, "Object recognition using invariant profiles", in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition* , pages 827–832,June 1997.

[108] D. Slater and G. Healey, "The Illumination-Invariant Matching of Deterministic Local Structure in Color Images", in *IEEE Transactions on Pattern Analysis and Machine Intelligence* , volume 19, number 10, pages 1146–1151,October 1997.

[109] D. Slater and G. Healey, "Modeling the sensitivity of moment invariants in a recognition system", *J. Opt. Soc. Am. A*, volume 15, number 5, pages 1068–1076, 1998.

[110] S. Smith and J. Brady, "Asset-2: Real-time motion segmentation and shape tracking", in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no 8, pages 814–820, 1995. (Similar material in Eng. Apps. of AI April 1994, and (without Brady) in ICCV95.)

[111] C. Stauffer and W. Grimson, "Adaptive background mixture models for real-time tracking", in *Proceedings of IEEE Conference on computer vision and Pattern Recognition*, pages 246–252, IEEE, 1999.

[112] D. Stoyan, W. S. Kendall, J. Mecke, *Stochastic Geometry and its Applications*, John Wiley and Sons, 1987.

[113] T. M. Strat, "Natural Object Recognition", Springer, 1990.

[114] Z. Sun, V. Ramesh, and M. Tekalp, "Error Characterization of Factorization Technique", in *Proceedings of the ICCV Workshop on Geometic Algorithms: Validation and Practice*, Korfu, Springer Verlag, September 1999.

[115] T. S. C. Tan and J. Kittler, "Color texture analysis using color histograms", in *Proceedings-Visual Image Signal Processing*, volume 141, number 6, pages 403–412, December 1994.

[116] A. Tankus and Y. Yeshurun, "Detection of regions of interest and camouflage breaking by direct convexity estimation", in *First IEEE International Workshop on Visual Surveillance*, pages 42–48, IEEE, 1998.

[117] B. Thai and G. Healey, "Representing Multiscale N-folded Symmetry in Color Texture", in *Proceedings of IEEE International Conference on Image Processing* , volume 1, pages 807–810, October 1997.

[118] R. Vogt, "Automatic Generation of Simple Morphological Algorithms", in *Proceedings of the CVPR*, pages 760–765, 1988.

[119] L. Wang and G. Healey, "Using Steerable Filters for Illumination-Invariant Recognition in Multispectral Images", in *Proceedings of IEEE International Conference on Image Processing* , volume 3, pages 138–141, October 1997.

[120] L. Wang and G. Healey, "Using Zernike Moments for the Illumination and Geometry Invariant Classification of Multispectral Texture", in *IEEE Transaction on Image Processing* , volume 7, number 2, pages 196–203, February 1998.

[121] S. Wang, T. Binford, "Local Step Edge Estimation – A New Algorithm, Statistical Model and Performance Evaluation", in *Proceedings of the ARPA IU Workshop*, Wash DC, pages 1063–1070, April 1993.

[122] S. Wang, T. Binford, "Generic, Model-Based Estimation and Detection of Discontinuities in Image Surfaces", in *Proceedings of the ARPA IU Workshop*, Vol. 2, pages 1443–1450, 1994.

[123] T. S. J. Weaver and A. Pentland, "Real-time American sign language recognition using desk and wearable computer based video", in *IEEE Transactions On Pattern Analysis and Machine Intelligence*, 1999. (See also MIT Medial Lab TR 466.)

[124] R. Wiemker, "The Color Constancy Problem: An Illumination-Invariant Mapping Approach", *CAIP* pages 950–955, 1995.

[125] L. Wixson, "Detecting salient motion by accumulating directionally-consistent flow", in*IEEE Transactions On Pattern Analysis and Machine Intelligence*, pages 774–781, August 2000.

[126] R. W. Wood, "Fish-eye view, and vision under water", *Philosophical Magazine*, 12(Series 6):159:162, 1906.

[127] J. Woodfill and R. Zabih, "An algorithm for real-time tracking of non-rigid objects", in *Proceedings Of the National Conf. On AI*, pages 718–723, July 1991.

[128] C. Wren, A. Azarbayejani, T. Darrell, and A. Pentland, "Pfinder: Real-time tracking of the human body", in *IEEE Transactions On Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pages 780–785, 1997.

[129] G. Wyszecki, W. S. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae*, Second Ed. New York: John Wiley & Son, 1982.

[130] Y. Yagi and S. Kawato, "Panoramic Scene Analysis with Conic Projection", in *Proceedings of International Conference on Robots and Systems (IROS)*, 1990.

[131] K. Yamazawa, Y. Yagi, and M. Yachida, "Obstacle Avoidance with Omnidirectional Image Sensor HyperOmni Vision", in *Proceedings of IEEE International Conference on Robotics and Automation*, pages 1062–1067, May 1991.

[132] J. Y. Zheng an S. Tsuji, "Panoramic Representation of Scenes for Route Understanding", in *Proceedings of the Tenth International Conference on Pattern Recognition*, 1: pages 161–167, June 1990.

[133] "Remote reality, inc", makers of ParaCamera systems, www.remotereality.com, 1998.

[134] "Special Section on Empirical Evaluation of Computer Vision Algorithms", in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 21, No. 4, pages 289–290, April 1999.

Appendix A

Mathmatic Appendix

A.1 Appendix: Variance Propagation

In this work we derived variances $\sigma^2_{\hat{f}}$ of a random variable $\hat{f}$, and covariance $\sigma_{\hat{f},\hat{g}}$ of two random variables $\hat{f}$, and $\hat{g}$. $\hat{f}$ and $\hat{g}$ are assumed to be normal distributed with $\hat{f} \sim N(f, \sigma^2_{\hat{f}})$ respectively $\hat{g} \sim N(g, \sigma^2_{\hat{g}})$ and happen to be a function of n random variables $\hat{v}_i \sim N(v_i, \sigma^2_{\hat{v}_i}), i \in \{0 \dots n-1\}$. Each variance $\sigma^2_{\hat{f}}$ presented can be derived directly from following relations by propagating various $\sigma^2_{\hat{v}_i}$. Let's define $\hat{a} = \hat{v}_0$, $\hat{b} = \hat{v}_1$, $\hat{c} = \hat{v}_2$, $\hat{A} = \hat{v}_3$, $\hat{B} = \hat{v}_4$, and $\hat{\vartheta} = \hat{v}_5$, and assume that they are all statistically independent if not specifically stated otherwise.

$$\begin{aligned}
\sigma^2_{\hat{f}} &= E\{\left(f - \hat{f}\right)^2\} && \text{(A.1)}\\
&= E\{(f - (f + \eta_f))^2\}\\
&= E\{\eta_f^2\}\\
\sigma_{\hat{f},\hat{g}} &= E\{\left(f - \hat{f}\right)(g - \hat{g})\}\\
&= E\{(f - (f + \eta_f))(g - (g + \eta_g))\}\\
&= E\{\eta_f \eta_g\} && \text{(A.2)}
\end{aligned}$$

In the following, we assume that $E\{\eta_f \eta_g\} = 0$ if $\hat{f}$ and $\hat{g}$ are statistically independent, and omit higher order error-terms. We derive covariance-terms for uncertainties in the results of following transforms:

Summation: For random variables a and b being correlated.

$$\sigma^2_{\hat{f}=\hat{a}\pm\hat{b}} = \sigma^2_{\hat{a}} + \sigma^2_{\hat{b}} + E\{\eta_a \eta_b\} \tag{A.3}$$

For random variables a and b being *un*correlated, $E\{\eta_a \eta_b\} = 0$

Square:

$$\sigma^2_{\hat{f}=\hat{a}^2} \approx 4a^2\sigma^2_{\hat{a}} + \sigma^4_{\hat{a}} \tag{A.4}$$

Multiplication: For random variables a and b being correlated.

$$\sigma^2_{\hat{f}=\hat{a}*\hat{b}} \approx a^2\sigma^2_{\hat{b}} + b^2\sigma^2_{\hat{a}} + \sigma^2_{\hat{b}}\sigma^2_{\hat{a}} \tag{A.5}$$

Division: For random variables a and b being correlated.

$$\begin{aligned}
\sigma^2_{\hat{f}=\frac{\hat{a}}{\hat{b}}} &= E\{\left(\frac{a}{b} - \frac{a+\eta_a}{b+\eta_b}\right)^2\} && \text{(A.6)}\\
&= E\{\hat{f}^2\} \qquad \text{with} \\
& \hat{f}^2 = \left(\frac{a}{b} - \frac{\frac{a}{b}+\frac{\eta_a}{b}}{1+\frac{\eta_b}{b}}\right)^2 \\
& \quad \approx \left(\frac{a}{b} - \left(\left(\frac{a}{b}+\frac{\eta_a}{b}\right)\left(1-\frac{\eta_b}{b}\right)\right)\right)^2 \\
& \quad \approx A+B \qquad \text{with} \\
& \qquad A = \frac{a^2\eta_b^2 + \eta_a^2 b^2 + \eta_a^2\eta_b^2}{b^4} \\
& \qquad B = -2\frac{a\eta_a\eta_b\left(1-\frac{\eta_b}{b}\right) + \eta_a^2\eta_b}{b^3} \\
& \text{such that} \\
\sigma^2_{\hat{f}=\frac{\hat{a}}{\hat{b}}} &\approx \frac{a^2\sigma^2_{\hat{b}} + \sigma^2_{\hat{a}}b^2 + \sigma^2_{\hat{a}}\sigma^2_{\hat{b}}}{b^4} + E\{B\} && \text{(A.7)}
\end{aligned}$$

For random variables a and b being *un*correlated, $E\{B\} = 0$

Trigonometric Functions: Using first order Taylor approximation.

$$\sigma^2_{\hat{f}=cos\hat{\vartheta}} \approx \sigma^2_{\hat{\vartheta}}\sin^2\vartheta \tag{A.8}$$

$$\sigma^2_{\hat{f}=sin\hat{\vartheta}} \approx \sigma^2_{\hat{\vartheta}}\cos^2\vartheta \tag{A.9}$$

Polar to Cartesian Coordinates:

$$\sigma^2_{\hat{x}} = \sigma^2_{\hat{R}\cos\hat{\theta}} \approx \sigma^2_{\theta}\left(R^2+\sigma^2_R\right)\sin^2\theta + \sigma^2_R\cos^2\theta \tag{A.10}$$

$$\sigma^2_{\hat{y}} = \sigma^2_{\hat{R}\sin\hat{\theta}} \approx \sigma^2_{\theta}\left(R^2+\sigma^2_R\right)\cos^2\theta + \sigma^2_R\sin^2\theta \tag{A.11}$$

$$\begin{aligned}
\sigma_{\hat{x},\hat{y}} &= \sigma_{\hat{R}\cos\hat{\theta},\hat{R}\sin\hat{\theta}} \\
&\approx E\{\left(R^2\sin\theta\cos\theta - (R+\eta_R)^2\sin(\theta+\eta_\theta)\cos(\theta+\eta_\theta)\right)\} \\
&\approx E\{\left(R^2\sin\theta\cos\theta - (R+\eta_R)^2(\sin\theta+\eta_\theta\cos\theta)(\cos\theta-\eta_\theta\sin\theta)\right)\} \\
&\approx \sin\theta\cos\theta\left(R^2\sigma^2_\theta + \sigma^2_R\sigma^2_\theta - \sigma^2_R\right) && \text{(A.12)}
\end{aligned}$$

Square Root:

$$\sigma^2_{\hat{f}=\sqrt{\hat{a}}} \approx \frac{\sigma^2_{\hat{a}}}{4a} \tag{A.13}$$

Special Transforms:

a)

$$\begin{aligned}\sigma^2_{\hat{f}=\frac{\hat{a}+\hat{b}}{\hat{a}^2-\hat{b}^2}} &= E\{\left(\frac{a+b}{a^2-b^2}-\frac{a+\eta_a+b+\eta_b}{(a+\eta_a)^2-(b+\eta_b)^2}\right)^2\} \quad (A.14)\\ &= E\{\left(\frac{A}{B}-\frac{A+\eta_A}{B+\eta_B}\right)^2\}\\ &= \sigma^2_{\hat{f}=\frac{\hat{A}}{\hat{B}}}\end{aligned}$$

with

$A = a+b, \qquad \eta_A = \eta_a + \eta_b$

$B = a^2-b^2, \quad \eta_B = 2a\eta_a + \eta_a^2 - 2b\eta_b - \eta_b^2$

With equations (A.3),(A.4), and (A.7) we finally get

$$\begin{aligned}\sigma^2_{\hat{f}=\frac{\hat{a}+\hat{b}}{\hat{a}^2-\hat{b}^2}} &\approx \frac{a^2b^2\left(a^2{\sigma_{\hat{b}}}^2+{\sigma_{\hat{a}}}^2b^2+{\sigma_{\hat{a}}}^2{\sigma_{\hat{b}}}^2\right)}{(a^2-b^2)^4}+\frac{4\,a^2{\sigma_{\hat{a}}}^2+{\sigma_{\hat{a}}}^4+4\,b^2{\sigma_{\hat{b}}}^2+{\sigma_{\hat{b}}}^4}{(a^2-b^2)^2}\\ &+ \frac{\left(a^2{\sigma_{\hat{b}}}^2+{\sigma_{\hat{a}}}^2b^2+{\sigma_{\hat{a}}}^2{\sigma_{\hat{b}}}^2\right)}{(a^2-b^2)^4}\left(4\,a^2{\sigma_{\hat{a}}}^2+{\sigma_{\hat{a}}}^4+4\,b^2{\sigma_{\hat{b}}}^2+{\sigma_{\hat{b}}}^4\right)\end{aligned} \tag{A.15}$$

b)

$$\begin{aligned}\sigma^2_{\hat{f}=\hat{a}^2+\hat{b}^2+n\,\hat{a}\hat{b}\hat{c}} &\approx 2\,{\sigma_{\hat{a}}}^2{\sigma_{\hat{b}}}^2+4\,b^2{\sigma_{\hat{b}}}^2+4\,nabc{\sigma_{\hat{b}}}^2+{\sigma_{\hat{a}}}^4+n^2a^2b^2{\sigma_{\hat{c}}}^2+4\,nabc{\sigma_{\hat{a}}}^2\\ &+ {\sigma_{\hat{b}}}^4+4\,a^2{\sigma_{\hat{a}}}^2+n^2{\sigma_{\hat{a}}}^2{\sigma_{\hat{b}}}^2{\sigma_{\hat{c}}}^2+n^2{\sigma_{\hat{a}}}^2{\sigma_{\hat{b}}}^2c^2+n^2a^2{\sigma_{\hat{b}}}^2c^2\\ &+ n^2a^2{\sigma_{\hat{b}}}^2{\sigma_{\hat{c}}}^2+n^2{\sigma_{\hat{a}}}^2b^2c^2+n^2{\sigma_{\hat{a}}}^2b^2{\sigma_{\hat{c}}}^2\end{aligned} \tag{A.16}$$

can be derived from equations (A.3),(A.4), and (A.5)

c)

$$\sigma^2_{\hat{f}=\frac{a}{a+c}} = E\{\left(\frac{a}{a+c}-\frac{a+\eta_a}{a+\eta_a+c+\eta_c}\right)^2\} \tag{A.17}$$

$$\begin{aligned}&= E\{\left(\frac{a}{b}-\frac{a+\eta_a}{b+\eta_b}\right)^2\}\\ &= \sigma^2_{\hat{f}=\frac{\hat{a}}{\hat{b}}}\end{aligned} \tag{A.18}$$

Continue with equation (A.7) where $b = a+c$ and $\eta_b = \eta_a + \eta_c$

Covariance for Normalized Color: With $S = R+G+B$ and $\eta_S = \eta_R+\eta_G+\eta_B$ and R, G, B being uncorrelated, we derive

$$\sigma^2_{\hat{r},\hat{g}} = E\{(\hat{r}-r)(\hat{g}-g)\} \tag{A.19}$$

$$\begin{aligned}&= E\{\left(\frac{R+\eta_R}{S+\eta_S}-\frac{R}{S}\right)\left(\frac{G+\eta_G}{S+\eta_S}-\frac{G}{S}\right)\}\\ &\approx E\{\left(\left(\frac{R}{S}+\frac{\eta_R}{S}\right)\left(1-\frac{\eta_S}{S}\right)-\frac{R}{S}\right)\left(\left(\frac{G}{S}+\frac{\eta_G}{S}\right)\left(1-\frac{\eta_S}{S}\right)-\frac{G}{S}\right)\}\\ &\approx E\{\left(-\frac{R\eta_S}{S^2}+\frac{\eta_R S}{S^2}-\frac{\eta_R\eta_S}{S^2}\right)\left(-\frac{G\eta_S}{S^2}+\frac{\eta_G S}{S^2}-\frac{\eta_G\eta_S}{S^2}\right)\}\\ &\approx \frac{1}{S^4}\left(RG\sigma_S^2-RS\sigma_G^2-GS\sigma_R^2\right)\end{aligned} \tag{A.20}$$

A.2 Appendix: Stauffer / Grimson Algorithm

The algorithm proposed [111] models each value $X_t \in R(t), G(t), B(t), I(t)$ of a particular pixel at time t as a mixture of Gaussians:

$$P(X_t) = \sum_{i=1}^{K} w_{i,t} * \zeta(X_t, \mu_{i,t}, \sigma_{i,t}) \tag{A.21}$$

where K is the number of distributions, $w_{i,t}$ is an estimate of the weight, and $\mu_{i,t}$ and $\sigma_{i,t}$ are the mean respectively variance values of the i^{th} Gaussian in the mixture at time t, and where ζ is a Gaussian probability function. In our system K is set to 4; K is essentially determined by computational power and availability of memory.

Based on the persistence and the variance of each of the Gaussians of the mixture, it is determined which Gaussian may correspond to background values. Each new pixel gray value X_t is checked against the existing K Gaussian distributions, until the best match is found. If none of the K distributions match the current gray value, the least probable distribution – the one with the smallest weight $w_{i,t}$ – is replaced with a distribution, that is characterized by the current value as its mean, and an initially high variance, and low weight. The prior weights $w_{k,t}$ of the K distributions at time t are adjusted as follows

$$w_{k,t} = (1 - \alpha_t)w_{k,t-1} + \alpha_t M_{k,t} \tag{A.22}$$

where α_t is the learning rate and $M_{k,t}$ is 1 for the model which matched and 0 for the remaining models. For details please refer to [111].

The μ and σ parameters for the unmatched distributions remain the same, while for the best matched distribution they are updated as follows:

$$\begin{aligned} \mu_t &= (1-\rho)\mu_{t-1} + \rho X_t && \text{(A.23)} \\ \sigma_t^2 &= (1-\rho)\sigma_{t-1}^2 + \rho(X_t - \mu_{t-1})^2 && \text{(A.24)} \\ \text{with} \quad \rho &= \alpha\zeta(X_t|\mu_{t-1}, \sigma_{t-1}) && \text{(A.25)} \end{aligned}$$

To determine, if a pixel represents object or background, the Gaussians are ordered by $w_{k,t}/\sigma_{k,t}$. A pixel is classified as background pixel if it is represented best by one of the first B distributions, where

$$B = argmin_b \left(\sum_{k=1}^{b} w_k > T \right) \tag{A.26}$$

and T is a measure for the minimum portion of the data that should be accounted for by the background. Otherwise, it is classified as object pixel.

This algorithm contains two significant priors, which are application dependent, and are imposed by the scene model: α, the learning constant, and T, the background proportion.

A.3 Appendix: Shadow Augmentation

To re-use the third-party module as proposed in chapter 5, we need to augment it such that it meets our application requirements. In the following, the augmentations are described.

Hypothesis Test for Classification of Shadow Pixels: Let $\hat{\mathbf{v}}_\mathbf{c} = (\hat{R}_c, \hat{G}_c, \hat{B}_c)^T$ be the current color values, and $\hat{\mathbf{v}}_\mathbf{b} = (\hat{R}_b, \hat{G}_b, \hat{B}_b)$ be most dominant background values of the mixture model in each color band with $\hat{\mathbf{v}}_\mathbf{c} \sim N(\vec{\mu}, \mathbf{\Sigma})$, and $\hat{\mathbf{v}}_\mathbf{b} \sim N(\vec{\mu}_\mathbf{b}, \mathbf{\Sigma})$. In the following we use the feature that the ratios between the current and the most dominant background values are approximately the same across all color bands (hypothesis H^0), while the ratio varies across the bands for non-shadow pixels (hypothesis H^1). We then test the hypothesis H^0 against H^1. Hypothesis H^0, which supports the shadow assumption reads as follows:

$$H^0: \quad \vec{\mu}_\mathbf{b} = k\vec{\mu}, \quad \vec{\mu}, \mathbf{\Sigma} \quad \text{known.} \tag{A.27}$$

Let $\hat{y}$ be the vector combining the current and most dominant background pixel estimates:

$$\hat{\mathbf{y}} = \begin{pmatrix} \hat{\mathbf{v}}_\mathbf{c} \\ \hat{\mathbf{v}}_\mathbf{b} \end{pmatrix} \sim N(\mathbf{y}, \mathbf{\Sigma}_{\hat{\mathbf{y}}}) \quad \text{with} \tag{A.28}$$

$$\mathbf{y} = \begin{pmatrix} \vec{\mu} \\ k\vec{\mu} \end{pmatrix}, \quad \mathbf{\Sigma}_{\hat{\mathbf{y}}} = \begin{pmatrix} \mathbf{\Sigma} & \mathbf{0} \\ \mathbf{0} & \mathbf{\Sigma} \end{pmatrix} \tag{A.29}$$

The probability $p^0 = p(\hat{\mathbf{y}}|\vec{\mu}, \mathbf{\Sigma}, k, H^0)$ can be written as

$$p^0 = \frac{1}{(2\pi)^{\frac{6}{2}}|\mathbf{\Sigma}|^{\frac{1}{2}}} exp\left(-\frac{1}{2}(\hat{\mathbf{y}} - \mathbf{y})^T \mathbf{\Sigma}_{\hat{\mathbf{y}}}{}^{-1}(\hat{\mathbf{y}} - \mathbf{y})\right) \tag{A.30}$$

Deriving the m.l.e. for k is straight forward. Solving $\frac{dp^0(k)}{dk}$ for $\hat{k}$ leads to

$$\hat{k} = (\mathbf{v_b}^T \mathbf{\Sigma}^{-1} \mu)/(\mu^T \mathbf{\Sigma}^{-1} \mu) \tag{A.31}$$

For hypothesis H^1, which supports the non-shadow assumption, only the scalar k is replaced by matrix $\mathbf{K}$:

$$H^1: \vec{\mu}_\mathbf{b} = \mathbf{K}\vec{\mu}, \quad \mathbf{K} = \begin{pmatrix} k_1 & 0 & 0 \\ 0 & k_2 & 0 \\ 0 & 0 & k_3 \end{pmatrix}; \quad \vec{\mu}, \mathbf{\Sigma} \text{ known.} \tag{A.32}$$

Deriving the m.l.e. for k_i follows the same principles as described above. Solving $\frac{dp^1(k_i)}{dk_i}$ for $\hat{k}_i$ leads to

$$\hat{k}_i = \frac{\mathbf{v_b}^T \mathbf{\Sigma}_i^{-1}}{\mu^T \mathbf{\Sigma}_i^{-1}} \tag{A.33}$$

where $\mathbf{\Sigma}_i^{-1}$ denotes the i^{th} column vector of $\mathbf{\Sigma}^{-1}$.

It can be easily shown that the likelihood ratio test statistic $\hat{d}$ is given by:

$$\begin{aligned} \hat{d} &= (\hat{\mathbf{v}}_\mathbf{b} - \hat{k}\vec{\mu})^T \mathbf{\Sigma}^{-1} (\hat{\mathbf{v}}_\mathbf{b} - \hat{k}\vec{\mu}) \\ &- (\hat{\mathbf{v}}_\mathbf{b} - \hat{\mathbf{K}}\vec{\mu})^T \mathbf{\Sigma}^{-1} (\hat{\mathbf{v}}_\mathbf{b} - \hat{\mathbf{K}}\vec{\mu}) \end{aligned} \tag{A.34}$$

For R, G, B being uncorrelated, $\mathbf{\Sigma} = \text{diag}(\sigma^2_{\hat{R}}\sigma^2_{\hat{G}}\sigma^2_{\hat{B}})$, such that equation (A.34) can be rewritten as

$$\hat{d} = (\hat{\mathbf{v}}_\mathbf{b} - \mathbf{M}\hat{\mathbf{v}}_\mathbf{b})^T \mathbf{\Sigma}^{-1} (\hat{\mathbf{v}}_\mathbf{b} - \mathbf{M}\hat{\mathbf{v}}_\mathbf{b})$$

$$
\begin{aligned}
&= (\tilde{\mathbf{M}}\hat{\mathbf{v}}_{\mathbf{b}})^T\mathbf{\Sigma}^{-1}(\tilde{\mathbf{M}}\hat{\mathbf{v}}_{\mathbf{b}}), \quad \text{with} \\
\tilde{\mathbf{M}} &= \begin{pmatrix} \left(-\frac{G_c^2}{\sigma_{\hat{G}}^2} - \frac{B_c^2}{\sigma_{\hat{B}}^2}\right) & \frac{R_cG_c}{\sigma_{\hat{G}}^2} & \frac{R_cB_c}{\sigma_{\hat{B}}^2} \\ \frac{R_cG_c}{\sigma_{\hat{R}}^2} & \left(-\frac{R_c^2}{\sigma_{\hat{R}}^2} - \frac{B_c^2}{\sigma_{\hat{B}}^2}\right) & \frac{G_cB_c}{\sigma_{\hat{B}}^2} \\ \frac{R_cB_c}{\sigma_{\hat{R}}^2} & \frac{G_cB_c}{\sigma_{\hat{G}}^2} & \left(-\frac{R_c^2}{\sigma_{\hat{R}}^2} - \frac{G_c^2}{\sigma_{\hat{G}}^2}\right) \end{pmatrix} \qquad (A.35)
\end{aligned}
$$

Since rank($\tilde{\mathbf{M}}$) = 2, it is obvious that for hypothesis H^0 (meaning k_1=k_2=k_3) $\hat{d}$ is χ^2 distributed with 2 degrees of freedom. It is easy to see that pixel values which support H^0 minimize $\hat{d}$. If the test statistic satisfies the 90-percentile of the distribution, we label a pixel as "shadow" pixel.

If the test statistic satisfies the 90-percentile of the distribution, we classify a pixel as shadow pixel.

For shadow pixels we introduce an additional mode parameterized by $w'_{K+1,t}, \mu_{K+1,t}, \sigma_{K+1,t}$ to the mixture model, so that equation (A.21) changes to

$$
\begin{aligned}
P(X_t) &= \sum_{i=1}^{B} w_{i,t} * \zeta(X_t, \mu_{i,t}, \sigma_{i,t}) + \sum_{l=B+1}^{K} w_{l,t} * \zeta(X_t, \mu_{l,t}, \sigma_{l,t}) + w_{K+1,t} * \zeta(X_t, \mu_{K+1,t}, \sigma_{K+1,t}) \\
&:= P(X_t)_1^B + P(X_t)_{B+1}^K + P(X_t)_{K+1} \qquad (A.36)
\end{aligned}
$$

Nevertheless, the background adaptation algorithm still runs as originally proposed on modes 1...K only. In case, sample X_t is classified as background, nothing changed. In case, the sample is classified as object and $P(H^o) < 90\%$ nothing changes, either. But, in case, the sample is classified as object *and* $P(H^o) > 90\%$, then following additional update procedure for the $K + 1^{th}$ mode is executed:

$$
\begin{aligned}
w_{k+1,t} &= (1 - \alpha_t P(H^o))w_{k+1,t-1} + \alpha_t P(H^o) && (A.37) \\
\mu_{k+i,t} &= (1 - \tilde{\tilde{\rho}})\mu_{k+1,t-1} + \tilde{\tilde{\rho}} X_t && (A.38) \\
\sigma_{k+i,t}^2 &= (1 - \tilde{\tilde{\rho}})\sigma_{k+1,t-1}^2 + \tilde{\tilde{\rho}}(X_t - \mu_{k+i,t})^2 && (A.39) \\
\text{with} \quad & \tilde{\tilde{\rho}} = \tilde{\rho} P(H^o) && (A.40)
\end{aligned}
$$

Additionally, the parameters which corresponds to the object mode that best represents the sample, is modified as follows. Let's assume the sample got assigned to object mode $o \in \{1 \dots K\}$:

$$
\begin{aligned}
w_{o,t} &= (1 - \alpha_t \bar{P}(H^o))w_{o,t-1} + \alpha_t \bar{P}(H^o) && (A.41) \\
\mu_{o,t} &= (1 - \tilde{\tilde{\rho}})\mu_{o,t-1} + \tilde{\tilde{\rho}} X_t && (A.42) \\
\sigma_{o,t}^2 &= (1 - \tilde{\tilde{\rho}})\sigma_{o,t-1}^2 + \tilde{\tilde{\rho}}(X_t - \mu_{o,t})^2 && (A.43) \\
\text{with} \quad & \tilde{\tilde{\rho}} = \tilde{\rho}\bar{P}(H^o) \quad \text{and} \quad \bar{P}(H^o) = 1 - P(H^1) && (A.44)
\end{aligned}
$$

Finally, all K+1 weights are normalized to so that they sum to 1.

Please note, that the shadow mode never gets destroyed or eliminated by object data or newly introduced modes[1].

The "best" matched mode j is defined as

$$j \quad = \quad \underset{k}{\text{argmin}} \left(d_{k,t} | d_{k,t} < (2.5\sigma_{k,t})\right) \quad \text{with} \quad d_{k,t} = |X_{c,t} - \mu_{k,t}| \forall k \in \{0 \dots B, K+1\} \qquad (A.45)$$

This means, that the shadow mode is included in the background representation. This changes equation (5.2) to

$$X_{b,t} = \mu_{j,t} | j = \underset{k}{\text{argmin}} \left(\mu_k - X_{c,t}\right)^2 \forall k \in \{0 \dots B, K+1\} \qquad (A.46)$$

A.4 Appendix: Hardware

Omnidirectional Camera

- CycloVision optics (parabolic mirror and orthographic lens)
- CCD Color Camera Panasonic GP-KR222

Foveal Camera

- Sony EVI D30

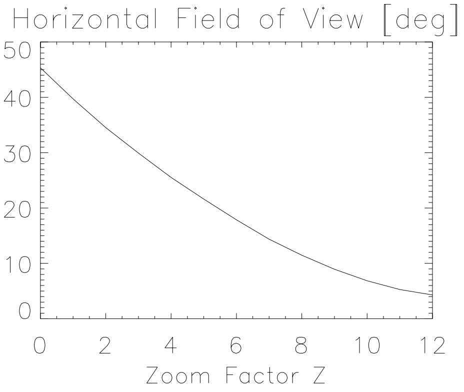

Figure A.1: Horizontal field of view in degrees. Corresponds to function $\left(T_Z^h\right)^{-1}(Z) = 2\gamma_h$.

Workstation

- Dell Precision 620, Dual Pentium III 600MHz, only one processor used for image processing and frame grabbing.
- Operating system Windows NT 4.0

[1] Its index is always $K + 1 > B$

Frame-Grabber

- Truevision Targa 2000
- Maximal frame-rate 10 frames per second, when operating on a 320 × 240 RGB image, 8 bit color depth.
- Using Microsoft's Video for Windows library

Appendix B

Notation and Symbols

Table B.1: Geometric model parameters (see also Figure 4.2, 4.3). Capital variables are variables in 3D, and small variables are given in image coordinates. ˆ (hat) indicates data values being observation of a random variable.

$\hat{H}_o$	height of OmniCam above floor (meters)
$\hat{H}_f$	height of foveal camera above floor (meters)
$\hat{H}_p$	person's height (meters)
$\hat{R}_h$	person's head radius (meters)
$\hat{R}_p$	person's foot position in world coordinates (meters)
$\hat{S}_p$	person's size (meters)
$\hat{D}_c$	on floor projected distance between cameras (meters)
$\hat{D}_p$	on floor projected distance between foveal camera and person (meters)
$\hat{D}'_p$	direct distance between foveal camera and person's center of face (meters)
$(\hat{x}_c, \hat{y}_c)$	position of OmniCam center, (in omni-image, pixel coordinates, Cartesian)
$(\hat{x}, \hat{y})$	position in omni space, (in omni-image, pixel coordinates, Cartesian)
$\hat{r}_m$	radius of parabolic mirror (in omni-image) (pixels)
$\hat{r}_h$	distance person's head – (in omni-image) (pixels)
$\hat{r}_f$	distance person's foot – (in omni-image) (pixels)
$\hat{s}$	projected size of person – (in omni-image) (pixels)
$\hat{k}$	number of pixels a person projects onto omni image plane
$\hat{\vartheta}$	angle between the person and the foveal camera relative to the OmniCam image center (please see Figure 4.3)
$\hat{\theta}_l$	angle between the left side of person and the foveal camera relative to the OmniCam image center.
$\hat{\theta}_r$	angle between the right side of person and the foveal camera relative to the OmniCam image center.
$\hat{\theta}$	angle between the radial line corresponding to the person and the zero reference line (please see Figure 4.3)
$\sigma^2_{(.)}$	Denotes variance of the variable used in the subscript
$\hat{\alpha}$	Tilt angle
$\hat{\beta}$	Pan angle
Z	Zoom factor
q'	Number of pixels summed within sector of interest in radial direction
	to generate feature $\hat{M}_{theta}$.
s'	Number of pixels summed in direction orthogonal to radial direction
	to generate feature $\hat{M}^{\top}_{r,\theta}$.
r'_i	Number of pixels summed in ith sub-sector along radial line in direction
$\hat{M}_{\theta}$	Sum of d^2 along radial line
${}^{i}\hat{M}_{\theta}$	Sum of d^2 along radial line within sub-sector
$\hat{M}^{\top}_{r,\theta_f}$	Sum in orthogonal direction bounded by θ_l and θ_r
$b^i(\theta)$	Binary sub-sector profile