People Tracking by Mobile Robots using Thermal and Colour Vision

Grzegorz Cielniak

Department of Technology
Örebro University

March 19, 2007

Abstract

This thesis addresses the problem of people detection and tracking by mobile robots in indoor environments. A system that can detect and recognise people is an essential part of any mobile robot that is designed to operate in populated environments. Information about the presence and location of persons in the robot's surroundings is necessary to enable interaction with the human operator, and also for ensuring the safety of people near the robot.

The presented people tracking system uses a combination of thermal and colour information to robustly track persons. The use of a thermal camera simplifies the detection problem, which is especially difficult on a mobile platform. The system is based on a fast and efficient sample-based tracking method that enables tracking of people in real-time. The elliptic measurement model is fast to calculate and allows detection and tracking of persons under different views. An explicit model of the human silhouette effectively distinguishes persons from other objects in the scene. Moreover the process of detection and localisation is performed simultaneously so that measurements are incorporated directly into the tracking framework without thresholding of observations. With this approach persons can be detected independently from current light conditions and in situations where other popular detection methods based on skin colour would fail.

A very challenging situation for a tracking system occurs when multiple persons are present on the scene. The tracking system has to estimate the number and position of all persons in the vicinity of the robot. Tracking of multiple persons in the presented system is realised by an efficient algorithm that mitigates the problems of combinatorial explosion common to other known algorithms. A sequential detector initialises an independent tracking filter for each new person appearing in the image. A single filter is automatically deleted when it stops tracking a person.

While thermal vision is good for detecting people, it can be very difficult to maintain the correct association between different observations and persons, especially where they occlude one another, due to the unpredictable appearance and social behaviour of humans. To address these problems the presented tracking system uses additional information from the colour camera. An adaptive colour model is incorporated into the measurement model of the tracker to improve data association. For this purpose an efficient integral image based method is used to maintain the real-time performance of the tracker.

To deal with occlusions the system uses an explicit method that first detects situations where people occlude each other. This is realised by a new approach based on a machine learning classifier for pairwise comparison of persons that uses both thermal and colour features provided by the tracker. This information is then incorporated into the tracker for occlusion handling and to resolve situations where persons reappear in a scene.

Finally the thesis presents a comprehensive, quantitative evaluation of the whole system and its different components using a set of well defined performance measures. The behaviour of the system was investigated on different data sets including different real office environments and different appearances and behaviours of persons. Moreover the influence of all important system parameters on the performance of the system was checked and their values optimised based on these results.

Acknowledgments

This thesis was not and could not be written in the quite solitude of a monk's cell. Therefore I would like now to express my gratitude to persons that contributed to this work.

First of all I am deeply indebted to my Supervisor, Dr. Tom Duckett. He guided me greatly through the sometimes curly roads of doing research. He taught me what research is really about, supported me with great ideas and enthusiasm, and spent many hours proof-reading and correcting this thesis and publications. He also showed great patience for misuse of articles and introduced me to an eccentric figure in the world of science: Dr. Who. He was a great companion during long runs on Markaspåret. Tom, I am proud to have you as a teacher and friend.

I was given a chance to work in an excellent research environment with great people and facilities. For that, I would like to thank the management of the Centre for Applied Autonomous Sensor Systems: Prof. Dimiter Driankov and Prof. Peter Wide. I would like to express my thanks to Prof. Ivan Kalaykov for arranging and initiating my stay in Sweden and for his help and support that was so needed at the beginning of my studies. I would also like to thank Prof. Krzysztof Tchoń, who was my supervisor during my undergraduate studies, for introducing me to Robotics and advising me to apply to Örebro University.

I would like to express my gratitude to Dr. Achim Lilienthal, my co-supervisor, for all the time he spent on reading the thesis and publications, providing excellent advice and apt comments. He was an irreplaceable collaborator while working on the occlusion detector and also provided a positioning system for the experiments with an omnidirectional camera. Achim, thank you for your support, friendship and being a great companion on Markaspåret.

I am grateful to Dr. André Treptow, a great collaborator on the tracking system. Thanks to his expertise on visual tracking I was introduced to the field without unnecessary pain. He is also a Godfather of the *PeopleBoy* robot. It was great and fun to work as well as run together on Markaspåret.

During my studies I had the honour to work in the lab of the Prof. Wolfram Burgard at the University of Freiburg, Germany. Thanks to his hospitality I got a chance to be introduced to the probabilistic methodology and work in a fine and supportive research environment. I would like to thank Dr. Maren Bennewitz for a friendly and fruitful cooperation and other members of the lab for help with the experiments and kind atmosphere. I acknowledge also a Marie Curie scholarship, as part of the European Commission's 5th Framework Programme, for funding these four months in Freiburg.

While working on experiments in the robot lab I got a helpful hand from several persons. I would like to thank students Mihajlo Miladinovic and Daniel Hammarin from Örebro University who contributed to the early experiments with the robot and omni-directional camera. I would like to thank the good fellows from the Learning System Lab, especially Henrik Andreasson and Jun Li, for their help with hardware and software implementations. I thank the lab engineer and great friend Per Sporrong, for his 24 hour technical support, keeping the robot up and running, and from whom I learned that things should also look the right way and that one good Swedish radio-station is P3.

I would like to express my gratitude to my sisters and brothers in arms: Ph.D. students at AASS (those who run and who do not on Markaspåret). For the research feedback, the patience and good will during these infamous data collection sessions and the exceptional atmosphere in the corridor, during lunch and coffee breaks as well as Monday meetings. Especially I would like to thank Amy Loutfi, Malin Lindquist, Abdelbaki Bouguerra, Boyko Iliev and Robert Lundh for their friendship.

Ultimately, I would like to thank my family and friends back home for supporting and cheering me up during these years.

Contents

List of Figures

List of Tables

List of Algorithms

Chapter 1

Introduction

1.1 Motivation

Many applications require or could benefit from a system that could “look” at people and answer relevant questions about them. The ultimate people recognition system would be able to answer questions such as: “are there any persons in the surroundings?”, “how many persons are there?”, “who are they?” and “what are they doing?” (see Fig. 1.1). Such a system could assist or completely replace a human operator in tasks that are too complex, difficult, monotonous, boring or badly paid. In addition it would open the possibility for completely new and interesting applications. Examples of existing systems involving people recognition are: automated surveillance systems that can detect an intruder or suspicious behaviour in public places, security systems verifying the identity of a person that limit the access to restricted areas, and driver assistant systems that can detect pedestrians and warn the driver in advance about possible danger. Other systems providing detailed analysis of human body movement are used in fields such as medicine, sports or for creating virtual agents in computer graphics and games.

A system that is able to “see” humans would also be an important component of a mobile robot that operates in a populated environment. Until now robots were used mainly in industrial applications, being deployed in highly controlled environments and having little or no possibility of interaction with people. In addition the mobility and autonomy of these robots was very limited. Recently, however, more and more mobile robots are designed to operate in populated environments. These so-called service robots are designed to work in hospitals, museums, office buildings or supermarkets, where they perform tasks such as cleaning, surveillance, entertainment, education and delivery. The autonomy of these robots opens possibilities for new interesting applications. In the future robots may also fight fires, rescue persons from the rubble, per-

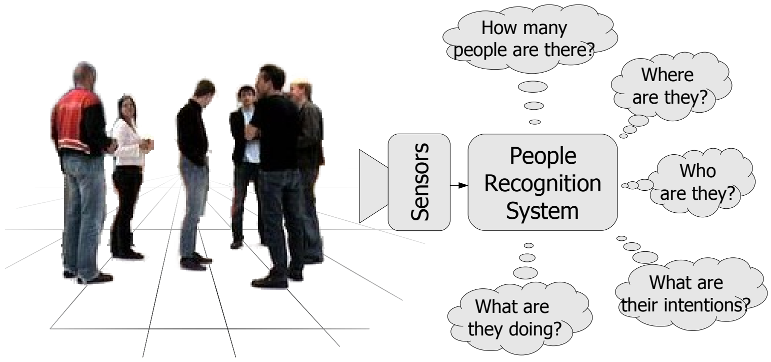
How many people are there?
Where are they?
Sensors
People Recognition System
Who are they?
What are they doing?
What are their intentions?

Figure 1.1: A generic people recognition system provides relevant information about humans.

form as security guards, and assist elderly people or customers in supermarkets. To realise all of these applications such a robot needs to have certain skills involving knowledge about people. First of all a robot must be aware of human presence to be able to navigate safely without the possibility of harming or disturbing people. A mobile robot should not only avoid persons but also adapt its navigation strategy, for example, to make way for people. A successful mobile robot, besides navigation skills, would need also the ability to communicate and cooperate with people.

An essential part of every people recognition system regardless of the application is a component that detects and localises humans. This information could be used by other components of the recognition system, for example, to localise human faces used later by a face recognition module or to localise body parts to recognise gestures or human behaviours. It could also be used by other components of the specific application, for example, by a mobile robot in navigation tasks such as avoiding persons or person following. The work presented in this thesis is concerned with people detection and tracking for mobile robotic systems.

1.2 The Problem

The main challenges for people tracking systems come from the fact that people have articulated bodies and their appearance can change drastically depending on pose, view, clothes, self-occlusions of different body parts, etc. Moreover their behaviour can be very unpredictable. To create a good model of persons it is necessary to extract common properties from all these variations either for all people (detection task) or for specific individuals (identification task). A successful people tracking system requires both these tasks to be solved, which leads to a trade-off between specificity and invariance. This is why creating an effective model of human appearance and behaviour can be a very complex task. By contrast tracking of rigid and predictable objects such as cars is much easier, which has resulted in many successful existing applications.

Other challenges appear when multiple persons are present on the scene. The tracking system has to estimate the number and position of all persons in the vicinity of its sensors. Additionally problems of occlusions by other persons or objects arise, as well as problems related to the identification of individuals including: the correct assignment of sensor measurements to persons (i.e., data association), identification of persons re-appearing on the scene ("have I seen this person before?"), and absolute identification ("exactly which person is it?"). Solving each of these problems would require an increased amount of resources: memory of previous frames, previous tracks and of all individuals in the database, and also increased computational demands. This thesis does not consider the problem of absolute identification, focusing rather on reliable data association and re-identification of temporarily occluded persons within the tracking process. Use of different sensors and modalities further complicates the whole problem, since data fusion has to be performed.

People tracking from a mobile platform differs in some aspects from non-mobile applications. There are several requirements that have to be fulfilled when designing mobile robotic systems. First of all useful approaches for mobile robots are those that can be utilised from a distance, so methods popular in non-mobile applications based on scanning of finger prints or the retina cannot be used. The ideal system should be able to recognise humans in their natural environment, without requiring any special registration or scanning procedure. The increased amount of sensor noise caused by the movement of the platform requires the methods to be robust. In addition robots operate in real-time and their computational resources are limited so the methods used should also be fast and efficient.

Our people tracking system meets these requirements:

- It is *non-invasive*, since the only sensors used are thermal and colour cameras.

- It is *robust*, due to the use of a probabilistic tracking algorithm and a thermal camera.

- It is *fast*, thanks to an efficient sample-based tracking algorithm and fast calculation methods for gradient and colour measurements. It allows for tracking of several people in real-time simultaneously.

The basic information about the location of persons provided by a tracking system can serve as a basis for designing more complex robotic systems. Possible extensions include recognising gestures, face expressions, intentions and behaviours. All these components would create a perception system oriented towards humans. Depending on the robot's task, this knowledge could be used to interact with people, avoid them and serve them, efficiently and reliably.

All of the issues and problems presented make the field of people tracking an open field for research, leaving many possibilities for improvements. There is no single method that would solve all existing problems related to people tracking and the right choice depends heavily on the application.

1.3 The Proposed Solution

The people tracking system presented in this thesis was entirely implemented on an ActivMedia PeopleBot robot (Fig. 1.2) and tested in different indoor environments. The sensory information for the tracking system is provided by two robot cameras: thermal and colour. A more detailed description of the robot and its environment is presented in Chapter 3.

The people tracking system uses a combination of thermal and colour information to robustly track persons (see Fig. 1.3). The use of a thermal camera simplifies the detection problem, especially on a mobile platform, and the colour information from a standard camera helps in situations with multiple persons. The system is based on a fast and efficient sample-based tracking method. Tracking of multiple persons is realised by an efficient algorithm that mitigates the problems of combinatorial explosion common to other known algorithms. A sequential detector initialises an independent tracking filter for each new person appearing in the image. Individual filters are automatically deleted when they stop tracking persons. Information from the colour camera is first aligned to the thermal image using an affine transform and after that it is incorporated into the tracking framework. A colour appearance model of a person is calculated using an efficient integral image method. Occlusions in the system are treated explicitly. A classifier learned using the AdaBoost algorithm [Freund and Schapire, 1995] allows the tracker to detect occlusions. Thus, the system can reason about occlusions in order to resolve situations where persons reappear in a scene.

Classical people tracking systems usually handle the detection and tracking tasks separately. This is done mostly to simplify the whole problem. However, such an architecture can cause loss of information

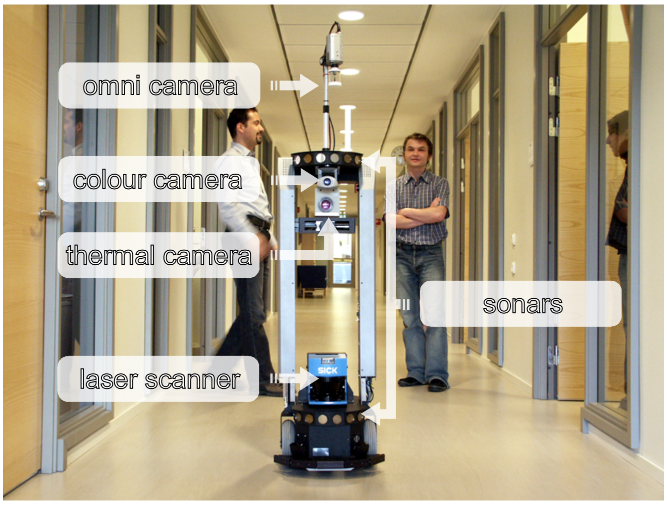

Figure 1.2: The ActivMedia PeopleBot robot *PeopleBoy* - the experimental platform used for testing the people tracking system.

between these steps, in addition to the computational cost of detection by exhaustive search of all possible poses of persons. Recent trends and techniques consider these problems simultaneously (track-before-detect, also called unified tracking [Stone et al., 1999]). Our system is designed in this latter spirit, using a track-before-detect technique.

In this work we do not use a global representation of the environment, but instead all interesting information about persons is expressed in sensor coordinates. This makes our approach similar to image-based servoing in robotic manipulators or behavior based robotics. In selected applications (e.g., a vision-based version of the “peg-in-a-hole” task [Yoshimi and Allen, 1994], a can collecting task based on Brooks’ subsumption architecture [Connell, 1989]) it has been shown that this approach can lead to more successful applications, being more robust and computationally efficient than systems using a global representation. A mobile robot with a people tracking system using a local representation of the environment should be able to successfully perform tasks such as finding and following persons in the neighbourhood, avoiding them, and interacting with them. A global representation of the environment is usually required in more abstract and complex tasks in combination with navigation behaviours that would allow a robot, for example, to find a person in a specified location. Such systems would involve complex methods providing more detailed information about humans at the cost of higher resource demands.

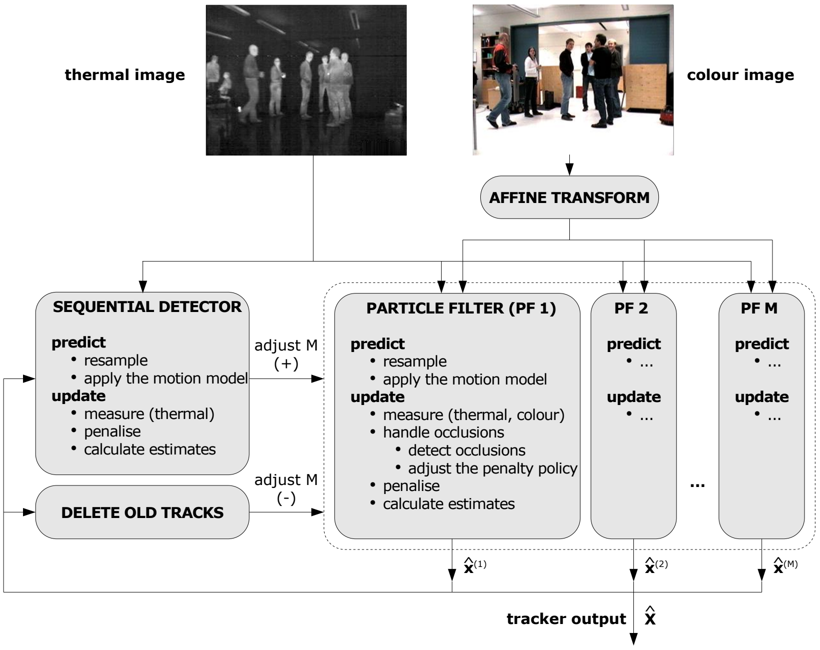

Figure 1.3: An overview of the people tracking system for mobile robots presented in this thesis.

1.4 Contributions

This thesis presents a people tracking system suitable for mobile robots. The specific contributions presented in this thesis include:

- Development of a vision-based people tracking system working on a real mobile robot.

- Introduction of a unified tracking method based on a particle filter and fast contour model of a person using thermal information to ensure a high frame rate and robustness to noise and occlusions.

- Proposal of an efficient heuristic tracking algorithm enabling tracking of a varying number of persons without a combinatorial explosion in the complexity.

- A new fusion method combining thermal and colour information for improved data association, using the integral image representation to speed up processing.

- Detection of occlusions using a combination of different visual cues selected by a machine learning classifier. This functionality is demonstrated by incorporating an explicit method of occlusion handling into the tracker based on the occlusions detected.

- A comprehensive, quantitative evaluation of the whole system using different performance measures.

1.5 Publications

Part of the content of this thesis has already been presented in a number of journal articles, conferences and workshops. Here is a complete list of publications arising during the course of this Ph.D. study. The publications are available on-line at http://aass.oru.se/pub/∼gck.

Journal Articles

- Grzegorz Cielniak, Achim Lilienthal and Tom Duckett. Multimodal People Tracking by Mobile Robots: combining colour and thermal vision with learned detection and handling of occlusions, *Submitted.*

- André Treptow, Grzegorz Cielniak and Tom Duckett. Real-Time People Tracking for Mobile Robots using Thermal Vision, *Robotics and Autonomous Systems*, Vol. 54, Nr. 9, pp. 729-739, 2006.

- Maren Bennewitz, Wolfram Burgard, Grzegorz Cielniak and Sebastian Thrun. Learning Motion Patterns of People for Compliant Robot Motion, *The International Journal of Robotics Research*, Vol. 24, No. 1, 2005.

- Grzegorz Cielniak and Tom Duckett. People Recognition by Mobile Robots, *Journal of Intelligent and Fuzzy Systems*, Vol. 15, No. 1, pp. 21-27, 2004.

Conference Proceedings

- Grzegorz Cielniak, André Treptow and Tom Duckett. Quantitative Performance Evaluation of A People Tracking System on a Mobile Robot, *Proc. of the European Conference on Mobile Robots*, Ancona, Italy, September 7-10, 2005.

- André Treptow, Grzegorz Cielniak and Tom Duckett. Comparing Measurement Models for Tracking People in Thermal Images on a Mobile Robot, *Proc. of the European Conference on Mobile Robots*, Ancona, Italy, September 7-10, 2005.

- André Treptow, Grzegorz Cielniak and Tom Duckett. Active People Recognition Using Thermal and Grey Images on a Mobile Security Robot, *Proc. of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Edmonton, Alberta, Canada, August 2-6, 2005.

- Grzegorz Cielniak, Maren Bennewitz and Wolfram Burgard. Robust Localization of Persons Based on Learned Motion Patterns, *Proc. of the European Conference on Mobile Robots*, Radziejowice, Poland, September 4-6, 2003.

- Grzegorz Cielniak, Maren Bennewitz and Wolfram Burgard. Where is ...? Learning and Utilizing Motion Patterns of Persons with Mobile Robots, *Proc. of the International Joint Conference on Artificial Intelligence*, Acapulco, Mexico, August 9-15, 2003.

Workshop and Symposium Papers

- Grzegorz Cielniak and Tom Duckett, People Recognition by Mobile Robots, *Proc. of the Joint SAIS/SSLS Workshop*, Lund, Sweden, April 15-16, 2004.

- Maren Bennewitz, Grzegorz Cielniak and Wolfram Burgard. Utilizing Learned Motion Patterns to Robustly Track Persons, *Proc. of the Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, Nice, France, October 11-12, 2003.

- Grzegorz Cielniak, Mihajlo Miladinovic, Daniel Hammarin, Linus Göransson, Achim Lilienthal and Tom Duckett. Appearance-based Tracking of Persons with an Omnidirectional Vision Sensor, *Proc. of the IEEE Workshop on Omnidirectional Vision*, Madison, Wisconsin, USA, June 21, 2003.

- Grzegorz Cielniak and Tom Duckett. Person Identification by Mobile Robots in Indoor Environments, *Proc. of the IEEE International Workshop on Robotic Sensing*, Örebro, Sweden, June 5-6, 2003.

1.6 Outline

The reminder of this thesis is organised as follows:

- **Chapter 2** presents the state of the art in people tracking including models and sensors used for detecting people, the theory behind Bayesian state estimation together with an efficient solution – the particle filter – and problems related to tracking of multiple persons. We also include a brief review on person identification and finally present existing applications of people tracking with a special focus on mobile robotics.

- **Chapter 3** introduces the experimental set-up, including a mobile robot and its sensors, on which the entire system was implemented, and the process of collecting ground truth data together with the metrics used for evaluation of different components of the system.

- **Chapter 4** presents a sample-based tracking filter enabling tracking of a single person in thermal images using an elliptic contour model. The experimental section of this chapter presents the overall performance of the system and the influence of different system parameters on performance.

- **Chapter 5** presents an extension to the basic system enabling efficient tracking of multiple persons together with an evaluation of the performance of the system.

- **Chapter 6** describes how the colour information is incorporated into the system, including the solution to the correspondence problem between thermal and colour cameras, a compact and efficient colour representation based on rapid rectangular features and data fusion of thermal and colour modalities. The experimental section provides a comparison of the performance of the system with and without colour information.

- **Chapter 7** presents an occlusion detector based on an AdaBoost classifier using a combination of thermal and colour features together with the evaluation and analysis of the performance of the detector. The learned occlusion detector is used for improved occlusion handling. An evaluation of the proposed approach is presented in the experimental part.

- **Chapter 8** concludes the thesis, presenting open questions, limitations of the system and possible improvements.

Chapter 2

Survey of Existing Methods for Detection, Tracking and Identification of People by Mobile Robots

This chapter presents the state of the art in people tracking and the theoretical basis for the people tracking system presented in this thesis. We first give an overview of the most popular models and sensors used for detecting people. Later we present the theory of people tracking, covering general Bayesian state estimation together with an efficient solution – the particle filter – and the problems of tracking multiple persons. In addition we briefly review related work on person identification. Finally we give an overview of existing applications of people tracking with special focus on mobile robotics.

2.1 Models

In people recognition and tracking models of people are used to help solve two different problems: to separate persons from other objects in the environment (detection) and to distinguish between different individuals (identification). The latter problem could be further decomposed, in increasing order of difficulty, into the problems of data association (deciding on a frame-by-frame basis "which observation corresponds to which person?"), association of new tracks with old tracks for persons that already appeared and disappeared ("have I seen this person before?"), and absolute identification ("exactly which person is it?"). This thesis is focused on the problem of detection and tracking; therefore only the problems of data association and re-identification of persons in the occlusion handling procedure are considered. However it should be straightforward to extend the system to also identify people re-appearing on the scene. Further extensions allowing for absolute identification of persons, even though possible within the existing framework, would require incorporation of reliable recognition techniques based, for example, on face recognition. The increasing complexity of these extensions would require more resources such as an increasing amount of memory (i.e., memory of recent frames, previous tracks and of all people in database) and computational power.

In detection the main difficulty is to extract common properties for all persons from the broad variety of human appearances. This appearance depends on a person's size, shape, clothes and additional features such as mustache, beard, glasses, jewelry, bags, etc. Moreover the appearance is affected by projection of the scene onto the sensor space, resulting in self-occlusions and occlusions by other objects and persons in the environment. In addition different individuals behave in different ways (standing, walking, sitting, lying down, cycling etc.) and their bodies can assume different poses. This also affects the detection task. On the other hand all these variations in appearance and behaviour make the identification task possible. The main goal in this case is to find specific and invariant properties for each individual. Therefore the choice of a proper person model for a specific application will always be related to a trade-off between specificity and invariance.

Another important issue that should be discussed is the complexity of the model. Complex models can provide very detailed information that is required in applications such as simulating virtual agents, or systems analysing the movement of a sportsman or dancer (see [Gavrila, 1999] for a survey of the existing applications). Such systems usually do not have strong constraints about processing time, often working in an off-line manner, and allow for special arrangements of the environment. In contrast on-line systems such as mobile robots usually do not require such detailed information, and therefore tend to favour simpler models that can fulfil the strict requirements for processing speed and robustness suffice. Therefore the complexity of a model will be dictated by the demands of a specific application limited by the available resources (sensors and computational power).

Let us present some of the existing models used in people recognition systems (see Fig. 2.1). We will use a general classification that separates them into object-centred and view-centred models.

Object-centred (also called view-independent) models are based on the structural characteristics of a person that are invariant to different

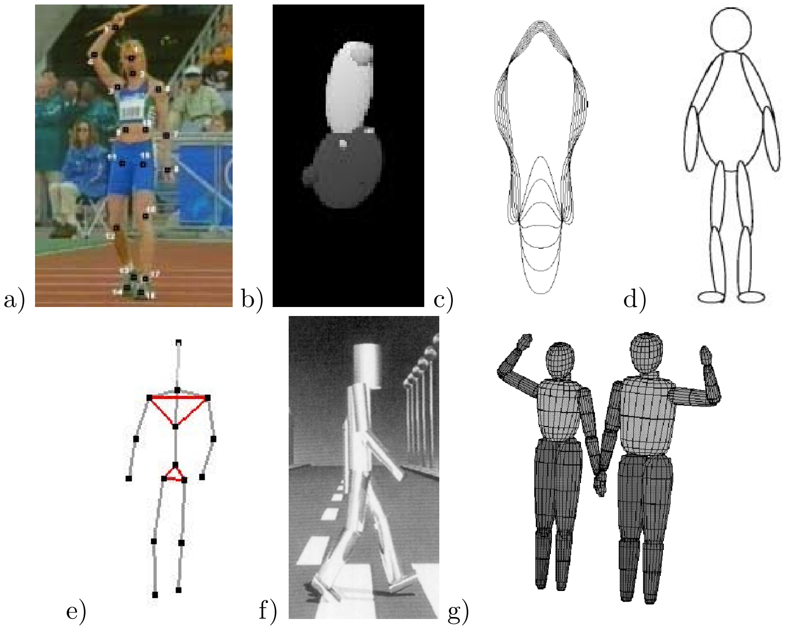

Figure 2.1: Different representations of the human body: a) points [Panagiotakis and Tziritas, 2004] b) blobs [Wren et al., 1997] c) splines [Baumberg and Hogg, 1994] d) ellipses e) skeleton [Liu et al., 1999] f) cylinders [Rohr, 1994] g) 3D model [Gavrila and Davis, 1996].

view-points. Depending on the representation these models can be categorised into stick figures [Chen and Lee, 1992] and volumetric models [Rohr, 1994]. Stick figures represent the skeletal structure of the body while volumetric models attempt to represent the whole body by decomposition into basic geometrical shapes such as spheres or cylinders. Object-centred models are used mostly in recognition tasks that require more complex analysis of the human body (e.g. gait recognition). One serious drawback of these models is the fact that they require a pose recovery procedure that maps information provided by the sensors to a $3D$ representation. This task is often computationally complex and demands special conditions such as use of multiple sensors and/or markers mounted on the person's body.

View-centred models (or appearance models) are grounded in features extracted from the information provided by sensors. These features correspond to different appearances of a person due to, e.g., different view-points, light conditions, poses of the body, etc. Existing approaches use features such as points, edges, ribbons or blobs [Chen and Lee, 1992], [Wren et al., 1997]. View-centred models avoid the difficult pose recovery step required by object-centred models. This fact makes view-centred models more robust in general to noisy sensory information. Moreover appearance models are not restricted to $2D$ information but may also contain $3D$ information (obtained from e.g., stereo-vision, structure from motion, range sensors, etc.).

From the perspective of mobile robotics, appearance models are more desirable since they are directly grounded in the robot's perception (there is no need to find correspondences between model components and image features). The internal representation in the sensor space does not limit possible applications and tasks (e.g., person following, user recognition). In general appearance models are also more robust and require less computational power, which in the case of limited hardware resources of a robot and high real-time demands cannot be ignored.

In this thesis we use a simple appearance model that approximates a person's projection onto the image space. Its simplicity allows this model to be combined with a fast tracking method. The model is based on thermal information which allows robust tracking of persons even in darkness. Our model helps to solve the two problems of detection and identification: an elliptic approximation of the person's contour is used to separate the person from the background, together with a colour model that allows the system to distinguish between different individuals and helps to solve problems caused by occlusions. More details about the elliptic model are given in Section 4.2.

2.2 Detection

Traditionally people detection is considered as a task carried out before tracking that determines the presence and number of persons from the input sensory data. This is realised by segmentation of the image data into regions corresponding to each detected person, usually by use of some model of a person (see previous section). In this section we present the most popular sensors and methods used to detect people, with a special focus on mobile robotic applications.

The most popular sensors used for detecting people are vision cameras. Most existing vision-based methods concern non-mobile applications (e.g., surveillance, pedestrian detection) where the pose of the camera is fixed. Detection in this case can be solved by background subtraction [Haritaoglu et al., 1998] or temporal differencing [Rohr, 1994]. In the first method foreground objects in the image frame are segmented after subtraction of the background model of the scene. The temporal differencing method uses differences between two consecutive frames to determine moving objects. Both approaches make a strong assumption that detected objects are persons. Other techniques use a further recognition step in which persons are discriminated from other objects [Niyogi and Adelson, 1994; Lipton et al., 1998].

Techniques based on skin colour can be used regardless of the motion of the sensor, therefore being very popular in mobile robotics applications [Wilhelm et al., 2002; Brèthes et al., 2004]. The skin colour of the human body is quite unique compared to other objects, which allows segmentation of regions in the image corresponding to the face or hands of a person. Similar approaches for detecting humans are based on face detection algorithms (see [Yang et al., 2002] for a detailed survey). Some popular methods from the vast variety of different algorithms include principal component analysis (PCA) [Turk and Pentland, 1991], template matching [Craw et al., 1992], or rapid detectors [Viola and Jones, 2001]. However, methods based on skin colour or face detection are usually limited to face and hand detection (assuming that people generally do not wander around naked!), hence persons must be facing the sensor. Recent advances in visual object recognition provide learning techniques that enable detection of people without assuming any a priori knowledge of the scene [Mohan et al., 2001]. They are, however, computationally demanding. All of the above mentioned vision-based systems share common problems such as shadows, varying lighting conditions and occlusions.

Use of non-standard vision sensors for people detection such as a stereo camera [Zhao and Thorpe, 1999] or thermal sensor [Nanda and Davis, 2002] helps to overcome some of the problems related to colour vision. Stereo vision provides extra range information that makes segmentation easier, allowing for detection of both standing and moving

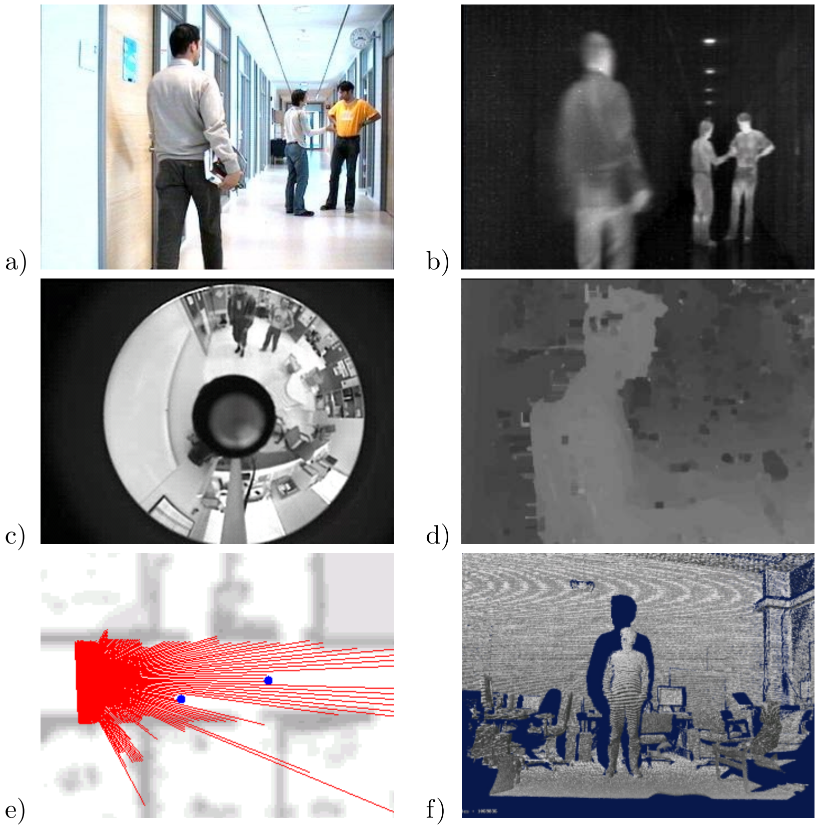

Figure 2.2: A populated environment seen from different robot sensors: a) colour camera image, b) thermal camera image, c) omni-directional camera image, d) a disparity map from a stereo camera, e) range readings from a laser scanner, f) a 3D point cloud model of a scene with added colour information.

people. Stereo vision has been applied only in a few mobile robotic applications [Huber and Kortenkamp, 1995; Kahn et al., 1996], perhaps due to the low resolution of depth information available from these sensors (typical stereo vision systems quantize the depth estimates into a maximum of 32 layers/disparities). Thermal vision takes advantage of the fact that humans have a distinctive thermal profile compared to non-living objects. Moreover thermal information is not influenced by changing lighting conditions and allows detection of people even in darkness. Infrared sensors have been applied to detect pedestrians in a driving assistance system: [Bertozzi et al., 2003] use a template based approach while [Nanda and Davis, 2002] apply different image filtering techniques. [Meis et al., 2003] filter the whole image and classify persons based on the symmetry of detected gradients. [Xu et al., 2004] employ a classification method based on a support vector machine. As yet, however, there is hardly any published work on using thermal sensor information to detect humans on mobile robots. The main reason for the limited number of applications using thermal vision so far is probably the relatively high price of this sensor, which is gradually decreasing.

Other types of sensors that can be used for people detection include range-finder sensors such as laser and sonar. These are very popular sensors in mobile robotics for navigation and localisation tasks [Fox et al., 1999]. A system described in [Schulz et al., 2001] detects local minima in range readings caused by the legs of a person and then removes all static objects by subtracting consecutive laser readings. In [Kluge et al., 2001] the authors cluster scan data into a set of points representing objects and by performing shape analysis extract those points corresponding to people. Both approaches detect moving objects rather than people. Despite the limitations of systems based on laser scanners (i.e. they can only detect “moving objects” rather than humans), they remain popular sensors in mobile robotic applications because of the low computational demands due to the low dimensionality of sensor data. Recent progress in building 3D range sensors (see an example in Fig. 2.2f) makes them promising sensors for future applications requiring people detection.

To overcome some problems related to a specific sensor it is possible to combine information from different sensors. For example, [Feyrer and Zell, 2000] use different features provided by a colour and stereo camera together with a laser scanner, and [Wilhelm et al., 2002] combine colour vision with sonars. This approach generally leads to more robust recognition systems. However, another problem arises here, namely sensor fusion – how to combine the different types of sensor information.

Our mobile robotic system uses a thermal camera to efficiently detect persons despite the motion of the platform. The distinct thermal profile of the human body is segmented by use of an elliptic model that can distinguish people from other warm objects such as radiators, lamps, monitors, etc. The results of the segmentation are also used later to select regions corresponding to persons on a colour image, providing additional information to distinguish between different persons (data association) during the tracking process.

2.3 Tracking a Single Person

Information provided by sensors can be imprecise or even misleading due to the sensor noise, clutter and dynamic occlusions caused by other objects or persons. Therefore to reliably estimate the location and movement of persons it is necessary to apply a tracking procedure. Tracking also enables combination of information from different sensors, giving more accurate and complete results.

The most popular approach to the tracking problem is based on the state space representation. Following this method, we describe a person's kinematics by a state vector and create a dynamical model of the person's movement. Tracking in this case is equivalent to the state estimation problem for a dynamical system given sensor observations. This work makes use of Bayesian inference, a widely accepted framework within the tracking community that models uncertainty in the system by means of probabilities.

We first describe the Bayesian estimation problem and its general solution for a single person (also referred to as a *target* in the general case). Later we present existing algorithms to solve this problem with special focus devoted to Monte-Carlo methods, which form the basis of the tracking methods used in this thesis. Multi-person tracking is then described in Section 2.4.

2.3.1 Bayesian State Estimation

The Bayesian approach to the estimation problem requires a probabilistic representation of the model dynamics. We will consider the case when the state changes continuously in time but can only be observed in discrete time steps through measurements. Having a sequence of measurements the estimation procedure could be done in two manners: either in batch mode or recursively. In batch mode estimated quantities are obtained from the whole set of observations. Each time a new observation arrives it is necessary to recalculate everything from scratch. The recursive case is much more appealing since estimates are just updated when necessary. This makes the recursive case well suited to on-line applications, requiring less resources and being faster than batch processing. However in the recursive case errors can accumulate with time and care has to be taken over the stability of sequential methods [Doucet et al., 2001].

The Model

Let us describe the state vector of a dynamical system at time step $t \in \mathbb{N}$ by $\boldsymbol{x}_t \in \mathbb{R}^{n_x}$ and the corresponding measurement vector as $\boldsymbol{z}_t \in \mathbb{R}^{n_z}$. To build a model of the dynamical system we would need the two following components:

- a system model, describing the temporal evolution of the state:

$$\boldsymbol{x}_t = \boldsymbol{f}_{t-1}(\boldsymbol{x}_{t-1}, \boldsymbol{v}_{t-1}), \tag{2.1}$$

  where $\boldsymbol{f}_{t-1}$ is a known, possibly nonlinear function of the state and $\boldsymbol{v}_{t-1} \in \mathbb{R}^{n_v}$ represents the process noise;

- an observation model:

$$\boldsymbol{z}_t = \boldsymbol{h}_t(\boldsymbol{x}_t, \boldsymbol{w}_t), \tag{2.2}$$

  where $\boldsymbol{h}_t$ is a known, possibly nonlinear function and $\boldsymbol{w}_t \in \mathbb{R}^{n_w}$ represents the measurement noise.

Noise sequences $\boldsymbol{v}_{t-1}$ and $\boldsymbol{w}_t$ are assumed to be white, independent, with known probability density functions (pdf or density).

Equation 2.1 represents a first order Markov model. We also assume that each observation $\boldsymbol{z}_t$ depends only on the system state at time t and not on past observations. Both these assumptions allow us to formulate a recursive version of the Bayesian estimator.

The Optimal Bayesian Solution

Our goal is to construct the posterior pdf of the state $\boldsymbol{x}_t$ given all the available information provided by the set of measurements $\boldsymbol{z}_{1:t} = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_t\}$. Using Bayes' formula the posterior density can be written as

$$p(\boldsymbol{x}_t|\boldsymbol{z}_{1:t}) = \frac{p(\boldsymbol{z}_t|\boldsymbol{x}_t, \boldsymbol{z}_{1:t-1})p(\boldsymbol{x}_t|\boldsymbol{z}_{1:t-1})}{p(\boldsymbol{z}_t|\boldsymbol{z}_{1:t-1})}. \tag{2.3}$$

We assume that the initial pdf $p(\boldsymbol{x}_0)$ is known.

Due to the independence assumption made on the observations $\boldsymbol{z}_{1:t}$, expression 2.3 can be simplified to

$$p(\boldsymbol{x}_t|\boldsymbol{z}_{1:t}) = \frac{p(\boldsymbol{z}_t|\boldsymbol{x}_t)p(\boldsymbol{x}_t|\boldsymbol{z}_{1:t-1})}{p(\boldsymbol{z}_t|\boldsymbol{z}_{1:t-1})}. \tag{2.4}$$

By introducing a new intermediate variable $\boldsymbol{x}_t$ we can transform the denominator $p(\boldsymbol{z}_t|\boldsymbol{z}_{1:t-1}) = \int p(\boldsymbol{z}_t|\boldsymbol{x}_t)p(\boldsymbol{x}_t|\boldsymbol{z}_{1:t-1})d\boldsymbol{x}_t$ (also called the *evidence*) and obtain the *update* equation:

$$p(\boldsymbol{x}_t|\boldsymbol{z}_{1:t}) = \frac{p(\boldsymbol{z}_t|\boldsymbol{x}_t)p(\boldsymbol{x}_t|\boldsymbol{z}_{1:t-1})}{\int p(\boldsymbol{z}_t|\boldsymbol{x}_t)p(\boldsymbol{x}_t|\boldsymbol{z}_{1:t-1})d\boldsymbol{x}_t}. \tag{2.5}$$

The likelihood function $p(\boldsymbol{z}_t|\boldsymbol{x}_t)$ is defined by the observation model in Equation 2.2. The term $p(\boldsymbol{x}_t|\boldsymbol{z}_{1:t-1})$ is the *prediction* density (or dynamical prior) that can be obtained by introducing an intermediate variable $\boldsymbol{x}_{t-1}$:

$$p(\boldsymbol{x}_t|\boldsymbol{z}_{1:t-1}) = \int p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})p(\boldsymbol{x}_{t-1}|\boldsymbol{z}_{1:t-1})d\boldsymbol{x}_{t-1}. \tag{2.6}$$

The transitional prior $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$ can be derived from the system model in Equation 2.1. The term $p(\boldsymbol{x}_{t-1}|\boldsymbol{z}_{1:t-1})$, which is referred to as the *prior* is exactly the posterior density from the previous time step, and because of the Markov assumption contains all previous information about the system up to time $t-1$.

The prediction and update equations (Equation 2.6 and 2.5 respectively) form the Bayesian filter, a recursive optimal estimator. Unfortunately this is only a conceptual definition since there is no general analytical solution for this filter. However in special cases (under certain assumptions) an optimal solution can be derived. Other methods provide approximate solutions. The next section presents optimal and approximate solutions to the Bayesian filtering problem. The classification used follows the presentation found in [Ristic et al., 2004], which also includes further references and more detailed descriptions.

Algorithms

Optimal solutions for the recursive Bayesian state estimator can be obtained under certain assumptions. In real systems, cases where these relatively strong assumptions hold are rare. Optimal algorithms include:

- *The Kalman filter*
  Assumptions: state and measurement functions are linear, process and measurement noise are Gaussians of known parameters. In this case the posterior density at every time step is a Gaussian characterised by two parameters, its mean and covariance. Despite the mentioned limitations the Kalman filter is still a very popular method in many existing tracking applications. A more detailed description is presented, for example, in [Bar-Shalom et al., 2001].

- *Grid-based methods*
  Assumptions: the state space is discrete and consists of a finite number of states. These methods become computationally inefficient with increasing size of the state space.

- *Beneš and Daum filters*
  Assumptions: the measurement model is linear. This is a limited class of non-linear filters for which there exists an exact solution. More details about this class of filters and grid-based methods can be found in [Ristic et al., 2004].

The above solutions are often inadequate for application in real tracking systems that must handle non-Gaussian, non-linear and non-stationary phenomena. Other solutions use suboptimal methods instead. We shortly describe the most popular methods to give a general overview. These methods can be divided into the following groups:

- *Analytic approximations*
  These methods are based on the extended Kalman filter (EKF). The main idea is to locally linearise the non-linear system and measurement functions in the model. The linearisation is done analytically and allows to represent the posterior $p(\boldsymbol{x}_t|\boldsymbol{z}_{1:t})$ by a Gaussian density. The basic EKF uses a linearisation procedure based on the first term of the Taylor expansion series and an obvious extension is to use further terms which results in the higher-order EKF. Another version of the EFK is its iterative variant that performs linearisation of the measurement equation based on the updated state of the filter (see [Bar-Shalom et al., 2001] for more details about the EKF and its different versions). All these filters are inappropriate for multi-modal densities because of the Gaussian assumption.

- *Numerical approximations*
  These methods apply numerical integration to solve the integrals found in Equation 2.6 and 2.5. They are also referred to as approximate grid-based methods. The computational cost of the approach increases dramatically with increasing size of the state space. Higher dimensionality also affects the convergence ratio. The state space must be predefined and therefore cannot be partitioned unevenly without the prior knowledge.

- *Gaussian sum filters*
  These methods are also known more generally as multiple model filters. The key idea is to approximate the posterior by a Gaussian mixture (a weighted sum of Gaussian density functions). There is a static version to approximate on-line parameters of the filter with a fixed number of components [Alspach and Sorenson, 1972] and a dynamic one using mixture models [Bar-Shalom et al., 2001].

- *Sampling approaches*
  These methods include the Unscented Kalman Filter (UKF) and Monte Carlo (MC) methods. The UKF uses the non-linear system model directly, unlike the EKF that performs analytical linearisation of the system model [Julier and Uhlmann, 1997]. The UKF represents the Gaussian distribution with a minimal set of sample points, which is far fewer than the number of samples needed by Monte Carlo methods. At each time-step, the UKF samples the

state around the current estimate in deterministic fashion. Each sample is updated using the non-linear system model and a new estimate is calculated after incorporating the new observations. The UKF produces a better approximation than the EKF for non-linear systems but its computational complexity is higher than the EKF. The UKF still makes the assumption of Gaussian probability distributions, hence it cannot handle multi-modal distributions. Monte Carlo methods, which are able to deal with non-linearities and multi-modal distributions, are described in more detail in the following section.

2.3.2 Monte Carlo Methods

Monte Carlo methods provide an approximate sample-based solution to the Bayesian estimation problem. The key idea is to represent the required posterior density function by a set of random samples with associated weights and to compute estimates based on these samples. As the number of samples becomes very large, this representation becomes equivalent to the true posterior density. Simultaneously such a filter approaches the optimal Bayesian estimator. These methods appear in different names depending on the domain where they are applied: particle filtering [Carpenter et al., 1997], bootstrap filtering [Gordon et al., 1993], interacting particle approximations [Del Moral, 1996], the condensation algorithm in computer vision [Isard and Blake, 1998] or “survival of the fittest” in genetic algorithms [Kanazawa et al., 1995]. The basic idea of the Monte Carlo methods is presented in Fig. 2.3.

Monte Carlo Integration

One way to deal with multidimensional integrals is to apply Monte Carlo integration. Suppose that we want to evaluate the following integral:

$$I = \int \boldsymbol{g}(\boldsymbol{x}) d\boldsymbol{x}, \qquad (2.7)$$

where $\boldsymbol{x} \in \mathbb{R}^{n_x}$.

If we can draw $N \gg 1$ samples $\{\boldsymbol{x}^i; i = 1, \ldots, N\}$ from the probability density function $\pi(\boldsymbol{x})$ such that $\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x})\pi(\boldsymbol{x})$ then we can obtain a MC estimate of the integral 2.7:

$$I_N = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{f}(\boldsymbol{x}^i). \qquad (2.8)$$

If the samples $\boldsymbol{x}^i$ are independent then the estimate I_N is unbiased and will almost surely converge to I. The variance of function $\boldsymbol{f}(\boldsymbol{x})$ is

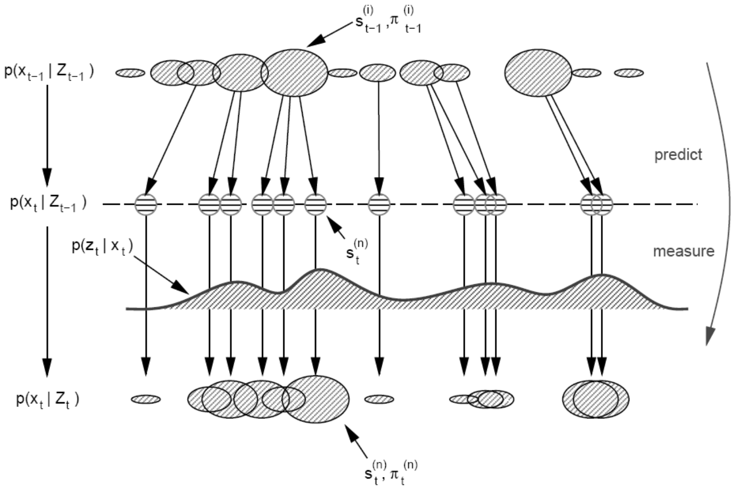

Figure 2.3: Example of a particle filter showing the main steps of prediction and update. A one-dimensional state space is represented, and the weight of the samples is indicated by their relative size. After calculation of the importance weights and resampling, the distribution of particles becomes more sharply peaked around several modes. Taken from [Blake et al., 1998].

of the form

$$\sigma^2 = \int (\boldsymbol{f}(\boldsymbol{x}) - I)^2 \pi(\boldsymbol{x}) d\boldsymbol{x} \tag{2.9}$$

and if it is finite then under conditions of the central limit theorem the MC estimation error $e = I_N - I$ converges such that

$$\lim_{N\to\infty} \sqrt{N}(I_N - I) \sim \mathcal{N}(0, \sigma^2). \tag{2.10}$$

The rate of convergence of this estimate is $O(N^{\frac{1}{2}})$ independent of the dimension of the integrand. This is a very important property that makes MC methods especially efficient in high dimensional problems. In contrast the convergence rate of any numerical integration method depends on the size of the integrand.

Usually it is not possible to sample effectively from the posterior distribution density $\pi(\boldsymbol{x})$ which is multivariate, nonstandard and known only partially up to proportionality constant [Ristic et al., 2004]. One way to overcome this limitation is to apply importance sampling.

Importance Sampling

If we cannot sample from $\pi(\boldsymbol{x})$ directly but we can sample from another distribution which is similar, then MC estimation is still possible. The only requirement on the so-called *importance* (or proposal) density $q(\boldsymbol{x})$ is that it has the same support as $\pi(\boldsymbol{x})$, where the support of a real-valued function f on a set X is defined as the subset of X on which f is nonzero, i.e.,

$$\pi(\boldsymbol{x}) > 0 \rightarrow q(\boldsymbol{x}) > 0, \tag{2.11}$$

for all $\boldsymbol{x} \in R^{n_x}$. Then

$$\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x}) \frac{\pi(\boldsymbol{x})}{q(\boldsymbol{x})} q(\boldsymbol{x}), \tag{2.12}$$

where $\frac{\pi(\boldsymbol{x})}{q(\boldsymbol{x})}$ is upper bounded and the MC estimate becomes a weighted sum

$$I_N = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{f}(\boldsymbol{x}^i) \tilde{w}(\boldsymbol{x}^i), \tag{2.13}$$

but this time samples $\boldsymbol{x}^i$ are drawn from the importance distribution $q(\boldsymbol{x})$. The importance weights are

$$\tilde{w}(\boldsymbol{x}^i) = \frac{\pi(\boldsymbol{x}^i)}{q(\boldsymbol{x}^i)}. \tag{2.14}$$

If we do not know the normalising factor (denominator) in the expression 2.14 then we have to perform normalisation of the weights as

$$w(\boldsymbol{x}^i) = \frac{\tilde{w}(\boldsymbol{x}^i)}{\sum_{j=1}^{N} \tilde{w}(\boldsymbol{x}^j)}. \tag{2.15}$$

The MC estimate can then be calculated as

$$I_N = \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{f}(\boldsymbol{x}^i) w(\boldsymbol{x}^i). \tag{2.16}$$

Sequential Importance Sampling (SIS)

The derivations provided in the previous sections will now be applied in a recursive manner to the Bayesian estimation problem. This will form the basis to most of the recursive MC methods. Different versions of the particle filter correspond to different choices for the proposal distribution. The posterior distribution (Eq. 2.3) at a given time step t is approximated by a set of weighted samples:

$$p(\boldsymbol{x}_t|\boldsymbol{z}_{1:t}) \approx \sum_{i=1}^{N} w_t^i \delta(\boldsymbol{x}_t - \boldsymbol{x}_t^i), \tag{2.17}$$

where δ is the Kronecker delta function. It can be shown [Ristic et al., 2004] that by introduction of the importance function $q(\boldsymbol{x})$ the weights w_t^i are updated as

$$w_t^i \propto w_{t-1}^i \frac{p(\boldsymbol{z}_t|\boldsymbol{x}_t^i) p(\boldsymbol{x}_t^i|\boldsymbol{x}_{t-1}^i)}{q(\boldsymbol{x}_t^i|\boldsymbol{x}_{t-1}^i, \boldsymbol{z}_t)}. \tag{2.18}$$

Unfortunately the form of the importance function $q(\boldsymbol{x})$ implies that the variance of the importance weights can only increase with time [Ristic et al., 2004]. This affects the accuracy of the MC estimate and leads to a phenomenon known as the degeneracy problem. After a few iterations of the SIS algorithm there will be only few particles with significant weight values. The negative effects of the degeneracy of particle weights can be reduced by introducing a resampling procedure.

Sequential Importance Resampling (SIR)

Negative effects of the degeneracy phenomenon appearing in the SIS filter can be eliminated by introduction of an additional resampling step in the filtering procedure. Resampling generates a new set of independent samples $\{\boldsymbol{x}_t^{i^*}; i = 1, \ldots, N\}$ from the original set of samples $\{\boldsymbol{x}_t^i; i = 1, \ldots, N\}$. The original samples are reselected with probability equal to their weights $Pr\{\boldsymbol{x}_t^{i^*} = \boldsymbol{x}_t^j\} = w_t^i$. As a result samples with high weights are duplicated and samples with low weight values are removed. There are efficient resampling methods of complexity $O(N)$ e.g., stratified, residual, systematic resampling (see [Douc et al., 2005] for comparison of different resampling methods).

The Sequential Importance Resampling (SIR) filter, introduced by [Gordon et al., 1993], originates from the SIS filter where the proposal distribution is chosen as a transitional prior (i.e. the density from the previous iteration after updating with the motion model). Additionally resampling is included after every filtering step. Substituting into 2.15 results in

$$w_t^i \propto w_{t-1}^i p(\boldsymbol{z}_t|\boldsymbol{x}_t^i). \tag{2.19}$$

Since resampling is done at every step the weights of all particles are set to uniform values. This implies that the weight update simplifies further to

$$w_t^i \propto p(\boldsymbol{z}_t|\boldsymbol{x}_t^i). \tag{2.20}$$

Other Filters

The SIR filter is regarded as a standard realisation of MC algorithms, or the "standard particle filter" in robotics. It is easy to implement since the importance density, which is chosen to be the transitional prior $p(\boldsymbol{x}_t|\boldsymbol{x}_{t-1})$, can be easily sampled. Moreover the sample weights can be directly evaluated from the likelihood $p(\boldsymbol{z}_t|\boldsymbol{x}_t)$ and there is no need to pass their values from the previous steps. However there are several drawbacks of this method and various other methods and improvements have been proposed.

The importance density $q(\boldsymbol{x})$ of the SIR filter does not contain any information about the latest observation $\boldsymbol{z}_t$, which results in degraded efficiency and sensitivity to outliers. The auxiliary SIR filter [Pitt and Shephard, 2001] tries to overcome these limitations. The resampling procedure is performed on samples from the previous time step $t-1$, which allows the current measurements to be incorporated into the sample weights. This makes the ASIR filter less sensitive to outliers in cases where the process noise is small. However, the usability of the ASIR filter is limited since its performance degrades with increasing process noise.

Resampling eliminates problems with sample degeneracy but creates another serious drawback. The so-called sample impoverishment phenomenon is caused by the fact that in the resampling step samples are selected from the discrete (not continuous) distribution. This very quickly causes a loss of diversity among the samples (especially in cases where the process noise is low) and after a few iterations almost all samples collapse into the same region. Negative effects are especially severe in the SIR filter, in which resampling is done at every step of the algorithm. One way to overcome this problem is to add some extra noise to the samples ("jittering"). [Gordon et al., 1993] proposed a roughening method which adds an amount of independent noise to all particles. An alternative solution proposed by the same authors, called prior boosting, performs sampling from an increased set of $M > N$ samples from the proposal distribution but uses only N samples in the resampling procedure. The regularised Particle Filter (RPF) [Oudjane et al., 2001] performs an additional regularisation step in the resampling procedure that “jitters” the samples. The RPF filter avoids sampling from the discrete distribution and samples from the continuous approximation of the posterior $p(x_t|z_{1:t})$ instead. The Resample-Move algorithm [Berzuini and Gilks, 2001] is based on a similar principle as the RPF, but in addition it checks whether the regularisation step for each sample should be accepted or rejected. This Markov chain step guarantees that samples asymptotically approximate those from the posterior. Both the RPF and Markov chain based methods perform better than the SIR filter in cases when the sample impoverishment is severe, for example, due to low process noise.

There is a vast variety of improvements to the standard particle filter (see [Ristic et al., 2004] and [Doucet et al., 2001] for full details) but they are outside the scope of this thesis, since the main focus is on real-time tracking of persons using available computational resources on a typical mobile service robot (see Chapter 3 for full details of our experimental platform). However, it is assumed that any future improvements to the SIR filter or enhancements made possible by increased computing power (e.g., parallelisation) could also be applied to our tracking system.

2.4 Multi-Person Tracking

Tracking of multiple persons introduces new problems that do not appear in the single-person case. First, the number of persons is not known since persons can appear/disappear from the scene but also can be occluded by other persons or objects. Second, it is not clear which sensor observation corresponds to which person, known as the *data association* problem. The aim of a tracking algorithm in the multi-person (or more general *multi-target*) case is to estimate both the number of persons and the state of all persons given a set of noisy measurements.

2.4.1 The Bayesian Formulation

To formulate the multi-target tracking (MTT) problem in the Bayesian framework let us introduce a multi-target state variable $\boldsymbol{X}_t = \{\boldsymbol{x}_t^1, \ldots, \boldsymbol{x}_t^M\}$ which consists of M (which is an unknown value) state vectors for all targets. Respectively $p(\boldsymbol{X}_t|\boldsymbol{z}_{1:t})$ will be the joint multi-target probability density (JMPD).

The Bayesian filter in this case consists of the following prediction equation

$$p(\boldsymbol{X}_t|\boldsymbol{z}_{1:t-1}) = \int p(\boldsymbol{X}_t|\boldsymbol{X}_{t-1}) p(\boldsymbol{X}_{t-1}|\boldsymbol{z}_{1:t-1}) d\boldsymbol{X}_{t-1}, \tag{2.21}$$

and update equation

$$p(\boldsymbol{X}_t|\boldsymbol{z}_{1:t}) = \frac{p(\boldsymbol{z}_t|\boldsymbol{X}_t)p(\boldsymbol{X}_t|\boldsymbol{z}_{1:t-1})}{\int p(\boldsymbol{z}_t|\boldsymbol{X}_t)p(\boldsymbol{X}_t|\boldsymbol{z}_{1:t-1})d\boldsymbol{X}_t}. \tag{2.22}$$

In such formulation the transitional prior $p(\boldsymbol{X}_t|\boldsymbol{z}_{1:t-1})$ is responsible both for evolution of the target states and their number M.

Practical solutions to the Bayesian formulation of the multi-target tracking problem are discussed as follows.

2.4.2 Classical Methods

Ideally a joint state representation should be used, which would allow to reliably estimate all target states including correlations between them. However solutions based on this representation quickly become inefficient due to the exponential growth of the state space. Thus the integrals in Equations 2.21 and 2.22 usually become intractable. A common practice to avoid this problem is to represent the state space as a set of independent single-target states, known as a factorial representation, in which the transitional prior can be expressed as

$$p(\boldsymbol{X}_t|\boldsymbol{X}_{t-1}) \propto \prod_{j=1}^{M} p(\boldsymbol{x}_t^{(j)}|\boldsymbol{x}_{t-1}^{(j)}). \tag{2.23}$$

The traditional approach to the multi-target tracking (MTT) problem makes use of this representation, where each target is assigned to a separate single-target tracking filter. Classical MTT methods based on the factorial representation require a pre-processing stage to search the raw sensor data for features corresponding to persons. With this approach the measurements are thresholded to form a set of observations. The observed features are then explicitly associated with existing tracks, used to create new tracks, or rejected as false alarms. In this detection-association-update scheme the main computational burden lies in the data association step (deciding which observation corresponds to which person).

The simplest data association methods are based on the nearest-neighbour approach [Bar-Shalom and Fortmann, 1988]. They use the most probable hypothesis (i.e. the closest or the strongest observation) about observation-target correspondence at a given time step t, discarding all the other possible assignments. These solutions usually do not perform satisfactorily, especially in cases where the targets are not well separated or when the false alarm rate increases.

Multi-Hypothesis Tracker (MHT)

The Multi-Hypothesis Tracker, introduced by [Reid, 1979], maintains all possible association hypotheses between observations and targets (see

Algorithm 1). A single hypothesis consists of one possible set of associations between all measurements and existing target hypotheses including the possibility of measurement being a false alarm. The MHT algorithm evaluates the posterior probabilities of the hypotheses and propagates them over time. Since all possible hypothesis are included the MHT algorithm is able to track a varying number of targets. Because of its complexity, which grows exponentially with time, this algorithm usually requires pruning and merging techniques to limit the number of possible hypothesis and make the method computationally tractable [Cox and Hingorani, 1996].

Algorithm 1 A single iteration of the MHT

```
given:
  - set of all association hypotheses $H_{t-1}$ together with corresponding multi-target states $\boldsymbol{X}_{t-1}$
  - set of $n$ new measurements $\boldsymbol{z}_t = \{z_t^1, \dots, z_t^n\}$
predict:
  - apply a motion model to all hypothetical multi-target states $\boldsymbol{X}_{t-1}$
associate:
  - generate a new set of hypotheses $H_t$ based on the matching of all $n$ new measurements with the predicted target hypotheses $H_{t-1}$
update:
  - evaluate all new hypotheses $H_t$ i.e., calculate their posterior probabilities
```

Joint Probabilistic Data Association Filter (JPDAF)

The Joint Probabilistic Data Association Filter (JPDAF) [Bar-Shalom and Fortmann, 1988] propagates only one set of association hypothesis, which results in reduced complexity compared to the standard implementation of the MHT. Estimates of the individual target states are calculated based on all measurements weighted according to the individual association probabilities, including the possibility of a measurement being a false alarm. However, the number of feasible individual associations grows exponentially with the number of targets and measurements. In contrast with the MHT, JPDAF can track only a fixed number of targets. It therefore requires some pre-processing to estimate the number of targets (see the following discussion). Efficient sample-based versions of JPDAF were proposed by [Schulz et al., 2001] and [Karlsson and Gustafsson, 2001]. [Maskell et al., 2004] proposed a solution that limits the combinatorial explosion, allowing tracking of a higher number of targets using JPDAF-based methods.

Algorithm 2 A single iteration of the JPDAF

```
given:
  - set of $m$ target states $\boldsymbol{X}_{t-1} = \{\boldsymbol{x}_{t-1}^1, \dots, \boldsymbol{x}_{t-1}^m\}$
  - set of $n$ new measurements $\boldsymbol{z}_t = \{z_t^1, \dots, z_t^n\}$
predict:
  - apply a motion model to $\boldsymbol{X}_{t-1}$
associate:
  - calculate association probabilities $H$ for all target-measurement pairs
update:
  - calculate new estimates $\boldsymbol{X}_t$ based on weighted associations $H$
```

2.4.3 Unified Tracking Methods

The classical approach to MTT using such a *detection-association-update* scheme originates from radar applications that differ in several aspects from vision applications. In vision-based tracking systems a classical detection procedure usually requires scanning of the entire image at every position and scale [Okuma et al., 2004], which is a computationally demanding procedure. Moreover, during the thresholding process some of the information is lost, which can be a crucial issue in cases where the signal-to-noise ratio of the sensor is low. This loss of information also occurs when detected objects overlap even to a small extent. In addition, to apply any of the above mentioned methods two assumptions about the origin of the observations are made: first, a single measurement can originate from a single target or clutter only, and second, a single target can cause only one measurement. However, these assumptions are violated in many vision applications requiring tracking of complex non-rigid objects that may overlap.

An attractive alternative to the classical approach is to incorporate raw measurements directly into the tracking procedure. Such a solution based on Bayesian filtering is called unified or *track-before-detect* tracking [Stone et al., 1999]. In this case association between raw measurements and target tracks is done implicitly within the Bayesian framework, and the processes of detection and tracking are carried out simultaneously. Sample-based versions of unified tracking algorithms provide tractable solutions to multi-target tracking problems, and some good examples include the multiple blob tracker BramBLe [Isard and MacCormick, 2001], and the work of [Orton and Fitzgerald, 2002], [Maskell et al., 2003], [Smith et al., 2005b] and [Kreucher, 2005]. Unfortunately these methods have other shortcomings, for example, the inefficient sampling method of [Isard and MacCormick, 2001], the fixed number of targets in [Maskell et al., 2003] and [Orton and Fitzgerald, 2002], or the complex sensor modelling of [Kreucher, 2005]).

2.4.4 Methods Based on a Joint State-Space Representation

Most multi-target tracking methods use a factorial representation of the state space to make the Bayes filter computationally tractable, but to model correlations between different target states a full joint state space representation is required. [Tao et al., 1999] and [Isard and MacCormick, 2001] use a standard particle filter (SIR) which was shown to be inefficient with increasing dimensionality of the state (number of objects). Therefore other, efficient sampling methods such as partitioned sampling [MacCormick and Blake, 2000] or Markov Chain Monte Carlo (MCMC) [Khan et al., 2004] should be used in this case. Partitioned sampling improves the sampling procedure by dividing all particles into partitions corresponding to each target, applying individual target dynamics and updating the final posterior by weighted resampling. A further improvement to this algorithm called distributed partitioned sampling [Smith and Gatica-Perez, 2004] sorts partitions according to their importance. MCMC methods introduce an extra step to the filtering procedure that selects samples from a Markov chain with a stationary distribution corresponding to the final target distribution. A variant of the MCMC method allowing tracking of a varying number of targets was proposed in [Smith et al., 2005b].

2.4.5 Estimating the Number of Objects

The majority of multi-target tracking techniques usually require some external procedure that estimates the number of objects. This can be realised for example by using probabilities of target birth and death to create and delete individual trackers [Bar-Shalom and Fortmann, 1988]. [Sidenbladh, 2003] uses an approximation of the joint state space based on its first moment. The integral of this so called probability hypothesis density gives a direct estimate of the number of targets. Methods that allow for direct estimation of the number of targets include the MHT and the method proposed by [Kreucher, 2005] that estimates the number of targets directly from the posterior within the tracking framework.

Issues such as the varying number of targets, efficiency of sampling methods and associations between measurements and tracks make multiple target tracking an open research problem. Since our application requires high processing speed, the solution presented in this thesis uses a set of individual particle filter-based trackers (factorial representation). Independent trackers are used in order to maintain linear algorithmic complexity with respect to the number of persons. To model correlations between objects, colour information is used to distinguish between different individuals and occlusions are also modelled explicitly. The measurements are incorporated directly into the tracking procedure (track-before-detect). A sequential detector is used to create a new track for a newly appearing target. Colour information is used to help with data association. Fusion of thermal and colour vision information is also carried out within the tracking framework.

2.4.6 Dealing with Occlusions

Since visual tracking involves projection of real-world objects onto the sensor space, a general solution to the multi-target tracking problem must be able to deal with the problem of occlusions. An occlusion occurs when the view of an object is fully or partially blocked by another object. Many types of occlusions cannot be resolved without domain specific knowledge, or models, helping to derive a structural interpretation of the image.

For the multi-person tracking problem addressed in this thesis, the possible sources of occlusion can be categorised into three main types:

- Self-occlusions caused by the persons themselves. This type of occlusion occurs when tracking different body parts that can interfere with each other. Since our system is designed to track the whole human body self-occlusions do not usually present a major problem.

- Occlusions caused by static and dynamic objects in the environment, e.g. tables, walls, moving vehicles, etc. Reasoning about this kind of occlusion in general would require additional knowledge about the environment. In mobile robotic applications, for example, this would require integration of different algorithms for map building, object recognition, etc., which would be outside the scope of this thesis.

- Occlusions caused by other persons when more than one person appears on the scene (see Figure 2.4). This happens especially often when people interact with each other in real crowded environments. Our work is focussed mainly on this type of occlusion.

In a few selected cases it is possible to solve problems related to occlusions by use of special sensors or their special arrangements. One example system uses a camera placed above the observed scene [Intille et al., 1997]. Persons observed from such a view-point cannot occlude each other. Another example is a multi-camera system [Mittal and Davis, 2002] where ambiguities caused by occlusion are resolved by combining information from different cameras placed in different places. However this solution introduces other problems such as sensor fusion, decisions on where to place the cameras, how many of them to use and increased costs. All these solutions can be used only in a few, very controlled scenarios and their use in mobile applications would be especially troublesome if not impossible.

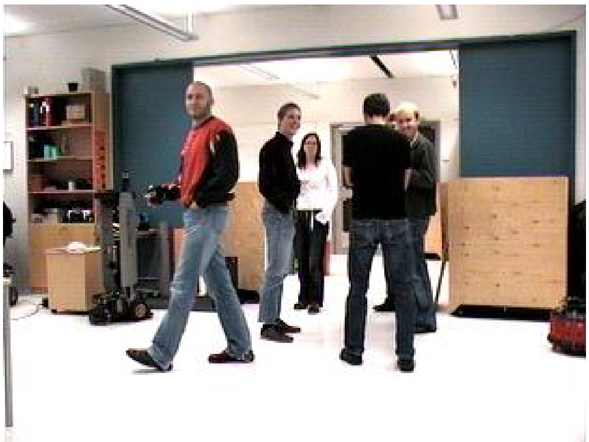

Figure 2.4: How many persons are there? A crowded scene with occluding people.

This thesis is focused on the passive aspect of perception, but it should be mentioned that an active system like a mobile robot could resolve some occlusions by taking appropriate actions. In ambiguous situations the robot could change its position so that occluded persons would become visible. This would require maintaining belief (e.g. probability distributions) over the persons' identities and design of a robust behaviour system that would allow the robot to operate in a populated environment.

In the majority of people tracking systems the problem of occlusion is solved within the tracking framework. Possible approaches handle occlusions either implicitly without reasoning, or model them explicitly. Implicit solutions use kinematic information as well as dedicated measurement models [Wren et al., 1997; Khan and Shah, 2000; Isard and MacCormick, 2001]. However the behaviour of people tends to be highly unpredictable in general, and they may or may not interact. Therefore implicit approaches can deal only with specific cases, i.e., short-term occlusions.

We decided to use an explicit approach in our people tracking system. This reasoning requires domain specific knowledge, i.e., detection of situations when persons appear to merge and split, and making decisions about their behaviour during occlusion (see for example [Elgammal and Davis, 2001; Senior et al., 2001; Mckenna et al., 2000]). We use colour as additional information that helps to detect occluded persons and resolve occlusions when occluded persons appear again on the scene. More details are presented in Chapter 7.

2.5 Identification

After detection and tracking of persons by a mobile robot, one of the next logical steps towards building complete systems for human-robot interaction is person identification. This includes deciding whether a person has been seen before and, if so, which person is present, by comparing the appearance to a database of known persons. If suitable visual features for discriminating different persons can be found, then standard techniques from the field of pattern recognition such as machine learning classifiers can be applied for identification [Duda et al., 2000]. While this thesis does not address the absolute identification problem, it uses similar techniques (comparing thermal and colour features for different persons) for improving data association in the multi-person tracking problem (deciding which observation corresponds to which person). Therefore this section gives a brief review of the literature on person identification, since all of these methods could also be applied to the problem of data association in multi-person tracking.

Most automatic methods for people identification use biometrics. A biometric is a measure based on physiological or behavioral characteristics of a person including appearance, social behaviour, bio-dynamics, natural physiognomy (e.g., skull measurements, fingerprint, retinal scans) and imposed physiognomy (e.g., dog-tags, bracelets). Some of these identification methods require interaction with the subject (e.g., fingerprints, iris, retina, hand-writing). Below we shortly describe the most common features used in person identification. We only consider features that are accessible without subject intervention (non-invasive methods), which makes them suitable for use by autonomous mobile robots.

Face

Face recognition is one of the most reliable methods to recognise humans. Most of the existing face recognition systems are vision-based (using either a single image or sequence). However, there are also techniques using range data (analysis of the 3D shape of the face [Gordon, 1992]) and thermal information [Socolinsky et al., 2001]. Existing algorithms can be divided into:

- *Holistic methods*, that use the whole face region as an input. Different approaches use eigenfaces (principal component analysis) [Turk and Pentland, 1991], fisherfaces [Belhumeur et al., 1996], support vector machines [Osuna et al., 1997], genetic algorithms [Liu and Wechsler, 2000] and artificial neural networks [Rowley et al., 1998].

- *Feature-based methods*, based on local features such as the eyes, nose or mouth. Representative examples include graph matching methods [Lades et al., 1993], hidden Markov models [Samaria and

Harter, 1994] and self-organizing feature maps [Lawrence et al., 1997].

- *Hybrid methods*, this approach is similar to the human perception system, combining analysis of both the whole face and local features [Pentland et al., 1994], [Penev and Atick, 1996].

Despite many advances in this field, a major problem – sensitivity to pose and illumination variation – still exists. Recent trends in this field lead to methods based on 3D geometrical models of the face.

Voice

Speaker recognition is the automatic process of recognizing who is speaking on the basis of characteristics (physiological and behavioural) found in speech waves. This information exists both in the short- and long-term spectral features. Most speech recognition systems are designed for verification of identity. Existing techniques for speaker identification can be divided into text-dependent and text-independent methods. The latter does not rely on a specific text being spoken, which makes it more useful in mobile robotics. The two most successful approaches are based on vector quantization [Soong et al., 1987] and hidden Markov models [Savic and Gupta, 1990]. Variability generated by the speaker, recording conditions and background noise makes the speaker identification problem still an open issue for further research.

Gait

Psychophysiological experiments and biomechanics studies provide evidence that gait signature contains possibly unique characteristics for each individual. Recognition methods can be divided into model-based methods which incorporate kinematics and dynamics of the human body [Cunado et al., 2003], and model-free methods [Little and Boyd, 1998]. The majority of gait recognition systems are based on vision, which leads to problems with segmentation of persons and occlusions.

Other

Other features that can be used in recognition systems are the whole appearance of persons (including face, hair, clothes, shoes, etc.) [Cielniak and Duckett, 2003], shape and proportions of the body, weight and lip movements. [Cielniak and Duckett, 2004] present results that indicate that it is possible to discriminate between different persons even using thermal features. All these features have been less well studied than the features described above, but still they can be used as a complementary input in systems combining multiple cues. Some examples of systems using multiple cues include [Brunelli and Falavigna, 1995] (face, voice), [Ross and Jain, 2003] (face, fingerprint, hand geometry) and [Yang et al., 1999] (whole body appearance, voice, face).

2.6 Applications

2.6.1 Non-Mobile Applications

Many people tracking systems have been designed for applications such as surveillance, video games, virtual reality interfaces, gesture recognition, user interfaces, etc. The majority of these systems use a stationary camera and are usually designed to work in known or partially controlled environments, which allows simplification of the detection task. The most popular models used are appearance models of humans, however in applications for movie industry or motion analysis more complex 3D models are used. Real-time systems usually use tracking techniques aided with heuristics to simplify the problem. Systems working in an off-line manner apply state-of-the-art tracking algorithms allowing to handle multiple persons and occlusions. Below we present few examples of people tracking systems designed for non-mobile applications.

PFinder is one of the first people tracking systems developed [Wren et al., 1997]. The system was used in many successful applications such as video games, virtual reality interface or even for gesture recognition. The system is able to track a single person in real-time (10 fps, 160x120 pixels). PFinder uses an appearance model based on statistics of colour and shape. This model is learned before-hand from data. Different body parts of a person are represented as blobs. This representation allows also for recognition of simple gestures. PFinder uses a stationary camera and background subtraction to detect a person and initialise the respective model. Tracking is realised by predicting position of blobs based on a constant velocity motion model and later updating the respective models based on the classification of pixels by a maximum a posteriori approach.

The system W^4 proposed by [Haritaoglu et al., 1998] is similar in spirit with PFinder. It also uses a blob based representation of the human body and learns appearance models based on histograms. The system can also estimate the rough pose of the body. The most important difference to PFinder is that W^4 system can track multiple persons: single separated persons and groups. The system uses a heuristic approach to tracking based on combination of prediction from the second order motion model together with correlation techniques updating the model. The Reading People Tracker [Siebel, 2003] also allows to track multiple persons. The system uses a Kalman filter based active shape tracker to track detected persons and has the ability to deal with partial occlusions.

Work by [Smith et al., 2005b] presents the latest achievements in the field of people tracking based on stationary cameras. The system is using a particle filter based multi person tracker. It models interactions between occluding persons hence is able to deal with partial and total occlusions. The advanced tracking methods do not allow to implement the system in real time, however.

[Gavrila and Davis, 1995] and [Sidenbladh, 2001] are examples of systems that use complex $3D$ models of persons. Because of their computational complexity they can be used only in specific applications without real-time requirements such as in movie industry or motion analysis of the human body. Both systems allow to track a single person only. The first system in addition requires a specially controlled environment (i.e. two camera set-up, persons wearing tight clothes). The latter system uses a particle filter to robustly track the pose of a person.

2.6.2 Mobile Applications

In the case of mobile robots, people tracking becomes an even more challenging task because the robot is moving and the environment is unpredictable. In addition computational resources are very limited since a robot usually has to perform other tasks such as navigation, planning, object recognition, etc., at the same time. Therefore not all techniques used in non-mobile applications can be directly applied to mobile robotics. Below we present existing mobile robotic applications designed to detect and localise people in the environment. Here the literature has been divided into systems which only detect humans based on the current sensor data and systems which track them using recursive methods for state estimation. Otherwise it is difficult to classify existing systems into distinct categories, because the field is at an early stage and a wide variety of techniques are still being explored. Perhaps two general trends can be observed: First, a few approaches rely on a model of the environment for separating humans from the background, therefore requiring accurate maps and self-localisation by the robot. Second, many approaches apply complementary sensor modalities, e.g., vision and range-finder data, in order to compensate for the limitations of each individual modality. For example, many approaches try to detect human legs from local minima in laser scans, but usually this information alone is not enough to guarantee reliable results, and another sensor (e.g., vision) may be used to confirm the presence of humans. A further observation is that almost no work has been done to objectively evaluate or compare the different methods: this thesis presents a first step in this direction, applying quantitative methods from computer vision to evaluate tracking performance on a mobile robot.

Detection Only Systems

Early mobile robotic systems, despite their simplicity and hardware limitations, showed that people can be detected and localised in the environment even though the platform is moving. They usually made strong assumption about the environment and used very limited and simple models of persons, expecting the user to be somehow aware of the robot.

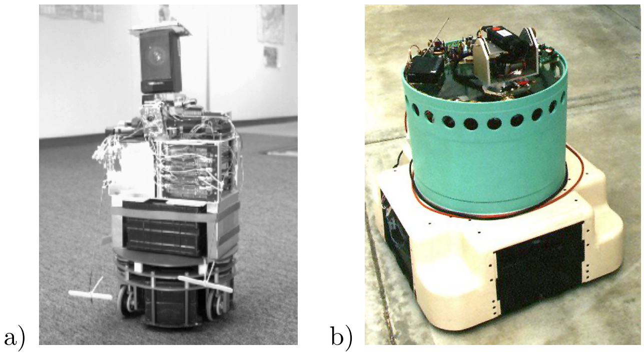

Figure 2.5: First mobile robots able to recognise persons: a) Polly [Horswill, 1993a], b) a biologically inspired system presented in [Blackburn and Nguyen, 1994].

The most popular sensor was a colour camera working in low resolutions to simplify the image processing. They usually could detect only a single person since no tracking procedure was applied.

Most probably the first robot designed to recognise people was Polly [Horswill, 1993a,b], a mobile robot that gave tours in the corridors of the MIT AI Lab (see Fig. 2.5a). It was equipped with a simple vision system (a camera pointing down at the floor with a resolution of 64x48 pixels and frame rate of 15 Hz) that was capable of detecting people in its surroundings. The system could estimate the "depth" of different objects in the environment by filtering out pixels belonging to the floor. This approach assumes that the environment is planar, so that depth can be estimated from height in the image plane, and that the floor has a distinctive texture that can be easily separated from foreground objects. Based on this depth information Polly could detect objects corresponding to people's legs (it was assumed that other similar objects like table or chair legs would not be present). Walls and junctions were detected by the same vision systems using a similar approach. Moreover the system could also recognise simple gestures such as foot waving, allowing simple interaction with the robot by the user.

[Blackburn and Nguyen, 1994] presented a mobile robot equipped with a biologically inspired vision-control system that could separate the motion of a moving object from the motion of the environment caused by the movement of the robot (Fig. 2.5b). The reported speed of the system was 15 fps with images of resolution 128x128 pixels. The system required the speed of the tracked object to be high enough to separate the object from the background, so the approach would only be useful for tracking humans while they move quickly from one place to another.

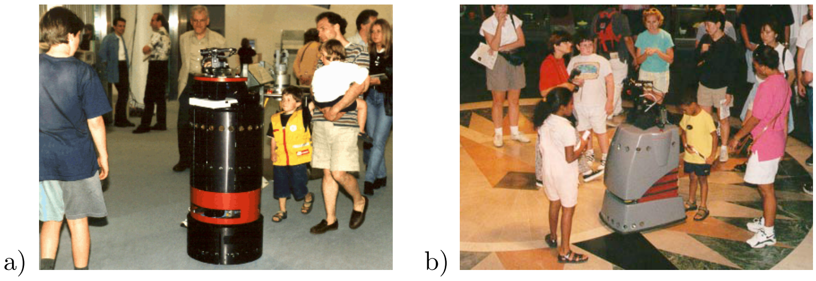

Figure 2.6: Robotic museum guides: a) Rhino [Burgard et al., 1999] b) Minerva [Thrun et al., 1999].

[Huber and Kortenkamp, 1995; Wong et al., 1995] presented a mobile robot with a visual attention system that was able to detect and follow an arbitrary object. The system used both stereo and motion information to detect the first object with enough texture information (assumed to be a person). This information was later used by the robot to pursue the detected object. The system could operate at a speed of 30 fps. Later in [Kortenkamp et al., 1996] a similar system was used for gesture recognition where a simple model of a person was introduced to obtain more reliable detection. Another robotic system recognising gestures of a person was presented by [Kahn et al., 1996].

The robot Rhino [Burgard et al., 1999] and its successor Minerva [Thrun et al., 1999] are examples of very successful mobile platforms designed to work in museums as artificial tour guides (see Fig. 2.6). They use proximity sensors (i.e., sonars and laser scanners) to detect people. The technique used for people detection is based on the entropy gain filter [Fox et al., 1998] which was intended to filter out unexpected objects from the robot readings. Assuming that these unwanted objects correspond to people, one can obtain a simple people detector. This technique therefore requires an accurate, up-to-date map of the environment and reliable self-localisation, otherwise any discrepancies between the map and environment would be detected mistakenly as persons (false positives).

Many mobile robotic systems use techniques based on face detection. In this case it is possible to choose from the vast amount of face detection techniques in computer vision. In addition many techniques could also be used for recognising the face of a potential user. [Barreto et al., 2004] presents a human-robot interface that relies purely on a face detector in combination with face recognition based on PCA. The robots Cog and Kismet from MIT use a face detector aided by speech recognition to localise humans. [Blanco et al., 2003] presents as system based on

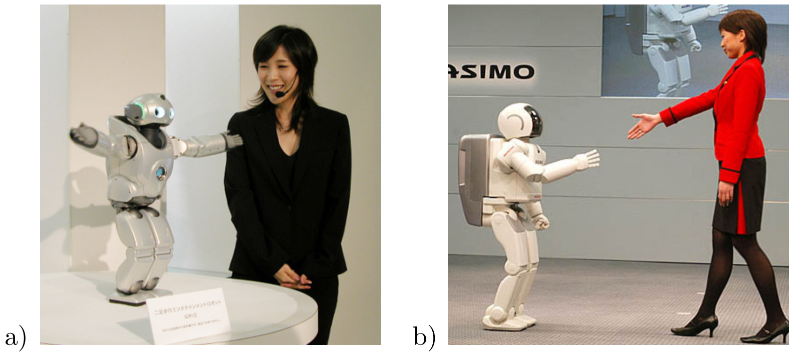

Figure 2.7: Commercially produced robots are able to recognise human faces a) QRIO by Sony, b) Asimo by Honda.

the combination of face detection and laser scanner. Techniques based on face detection are still very popular and have been used in recent commercially produced robots such as Sony's QRIO and Honda's Asimo (see Fig. 2.7).

There are several robotic applications which fuse information from multiple sensors and modalities to assure more reliable detection. The group from Ilmenau in Germany developed a system that can detect persons using different cues including skin colour, face detector, head-shoulder contour, motion and speech [Boehme et al., 1998, 1999]. To combine information from different cues they used a biologically inspired approach based on saliency maps that represent the importance of each individual cue in different scales. [Feyrer and Zell, 1999] proposed a system that combines skin colour, motion, contour and stereo information. This system is based on hierarchical detection: skin colour and motion cues are used to select regions of interest that are later filtered depending on contour and stereo information. [Byers et al., 2003] present another system combining skin colour and laser data in the interesting application of a robotic photographer. In this system hypothetical regions corresponding to people are found by detecting skin colour, and laser data is used to estimate their distances and determine their size. Despite the simplistic techniques this application has been successfully tested in crowded environments such as conferences and public presentations.

Tracking Systems

[Schlegel et al., 1998] present a people tracking system that uses a model of a person including a colour histogram and adaptive contour model that is learned during an initialisation phase. The system can detect and track people in a range of 1 to 5 m. In this system tracking is performed by an adaptive procedure that updates both models. Example of systems where tracking was realised in a similar way include [Waldherr et al., 1998] and [Brèthes et al., 2004]. Both systems use skin colour to select the regions of interest and later update the colour models of these regions. In addition the system in [Waldherr et al., 1998] uses an adaptive colour model of the shirt to increase the robustness of detection.

[Wilhelm et al., 2002] present a robotic shop assistant. The tracking system is based on a particle filter allowing to track a single person (a potential user). The system uses a combination of skin colour, contour-based information and additional range information from the robot's sonars. Information from different modalities is combined by a fuzzy data fusion technique.

[Kleinehagenbrock et al., 2002] present another technique that combines skin colour and laser data. In this case the two modalities are fused by means of symbols and a respective set of rules. The system is able to track one person and the reported speed is around 3-4 fps with resolution 198x139 pixels for the vision system tracking skin colour and 4.6 Hz for the laser. The system is able to combine the different sensor data asynchronously.

[Scheutz et al., 2004] present a system that could track multiple persons with information provided by a laser for leg detection and a camera for skin colour detection. The skin regions are used to verify hypothesis of the existence of persons given by the laser tracker. Despite the ability of the system to track multiple persons, occlusions are not handled.

[Schulz et al., 2001] present a particle filter based approach to track multiple moving objects from a mobile platform. It uses information from a laser scanner to detect the legs of persons, using scan matching between consecutive scans to separate the legs from the background, meaning that the approach cannot detect stationary persons. For each object a single particle filter is used and the JPDAF algorithm is used for data association. This algorithm explicitly models occlusions to increase robustness, relying on the motion cue. This work gives the first rigorous presentation of the multiple person tracking problem in the field of mobile robotics. The sample-based JPDAF algorithm has also been applied in tracking applications in other fields.

[Montemerlo et al., 2002] present a similar system. In this case the tracking system uses information about the environment (an accurate metric map collected beforehand) to increase the robustness and accuracy of the tracker. The people tracking is solved simultaneously with the problem of pose estimation of the robot. However this technique would be sensitive to dynamic environments, increasing the number of false detections, since it relies on having an up-to-date map of the environment. To solve data association the nearest neighbour approach is used.

The system described by [Jensen and Siegwart, 2003] uses a laser scanner and is similar in principle to [Montemerlo et al., 2002] in the sense that the system uses a map of the environment and then selects objects that are outliers. This is realised by the EM algorithm and a feature-based representation of the environment to reduce the complexity of the algorithm. However, the EM algorithm requires many iterations and all previous data to be available, so the scalability of this approach is not clear.

Another laser-based people tracking system mounted on a wheelchair is presented by [Kluge et al., 2001]. Shapes of persons are represented by a set of objects (i.e. vertices and edges) extracted from the laser scan. Tracking of multiple moving persons is realised by matching objects from two consecutive scans represented in a graph-like structure. The association is performed by standard optimisation techniques used in graph theory. However this approach allows to track only well separated, moving objects and data association realised by the graph can fail in cases of simultaneous appearance/disappearance of objects.

[Cielniak et al., 2003] developed a method for people tracking by mobile robots allowing the incorporation of additional knowledge about the behaviour of the people. These behaviours are learned off-line by clustering recorded trajectories of the persons provided by a laser scanner using the EM algorithm. The results from the clustering are later used to construct a person-specific hidden Markov model (HMM). This model can predict the intentions of a person and is used for on-line tracking. Range and colour information is used to update the HMM model where colour allows to distinguish between different individuals and tracking of multiple persons. The issues related to on-line learning of behaviours would need to be investigated further to allow application of this system in practice.

[Zajdel et al., 2005] address the problem of tracking and identification of persons from a mobile platform using vision. The proposed system segments persons from the image in two ways: by a standard background subtraction method when the robot is stationary and motion extraction from optical flow when the robot is moving. The tracking procedure is realised by a colour matching algorithm. The resulting tracks are later used for re-identification of persons entering the field of view of the camera. This so-called global tracking is used to determine whether the observed person has been seen before. The method uses a Bayesian network that associates local tracks using colour and spatio-temporal features extracted from the tracks. However the authors did not investigate or discuss some possible problems of the approach. This would include, for example, the influence of faulty tracks on the performance of the global matching algorithm, or tractability of the approach with a growing number of observed tracks (the complexity of the proposed Bayesian network grows exponentially with the number of tracks).

2.7 Conclusions

In this chapter we presented the theoretical background and state of the art in people tracking for mobile robots. The major challenge for people tracking systems lies in reliable detection and localisation of people. The selection of an appropriate model of a person depends heavily on the application, however in general appearance models seem to be more suitable for mobile robots. The most popular sensors are vision cameras, but other sensors such as a thermal camera or laser scanner can simplify or aid the detection task.

The Bayesian framework allows to specify the tracking problem in a rigorous way. There are efficient sample-based algorithms that allow to solve the general tracking problem. Tracking of multiple persons introduces the further problems of estimating the number and order of persons, as well as their locations. Despite the recent achievements in this field, people tracking remains an open research area due to challenges such as reliable detection of persons, efficiency of the tracking algorithms and successful occlusion handling.

People tracking systems have been used in many interesting applications in different fields. Not all of the techniques used in non-mobile applications can be directly transferred to robotic systems due to the increased amount of noise, movement of the platform, unpredictability of the environment and computational limitations. The presented mobile applications illustrate the need for fast and reliable people tracking systems.

Chapter 3

Experimental Set-up and Evaluation Metrics

This chapter presents the set-up used to conduct the experimental part of the thesis. We first present the mobile platform - an ActiveMedia mobile robot - together with its sensors. The people tracking system described in the thesis was entirely implemented on the robot. We describe also the process of collecting the ground truth data. Finally we present the metrics used for evaluation of the tracking system throughout the remainder of the thesis.

3.1 Experimental Platform

The experimental platform used in the thesis was *PeopleBoy* - an ActivMedia PeopleBot robot (Fig. 3.1), which is a mobile platform especially designed for tasks involving interaction with humans. It is equipped with an array of different sensors including a colour pan-tilt-zoom camera (VC-C4R, Canon) and thermal camera (Thermal Tracer TS7302, NEC). Information from these two cameras was used as input to the people tracking system. The robot was equipped with a computer based on the Intel Pentium III processor (0.85 GHz) running the Linux operating system. The code for the tracker software was implemented in C++.

The colour and thermal cameras are mounted close to each other to allow for easy combination of the information from both cameras (see Section 6.1.1). The cameras are connected to on-board frame-grabbers allowing simultaneous capturing of images with a maximum frequency of 15 Hz. The thermal camera (see Fig. 3.2a) converts infrared radiation into an image where each pixel corresponds to a temperature value (see Fig. 3.2b). In our set-up the visible range in the grey-scale image was

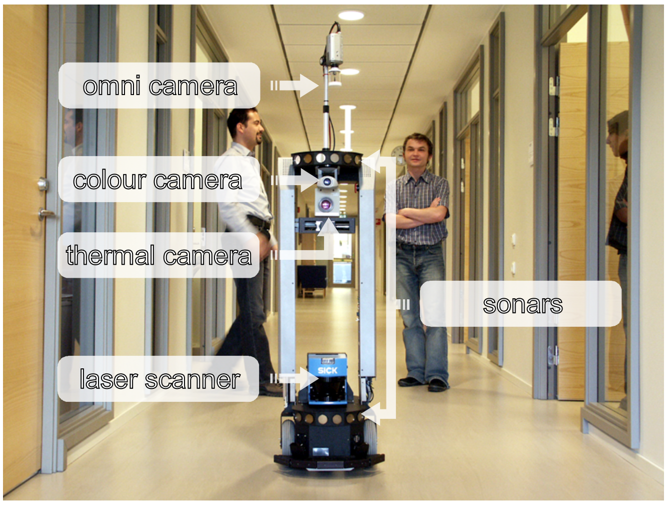

Figure 3.1: The ActivMedia PeopleBot robot *PeopleBoy* equipped with an array of different sensors.

equivalent to the temperature range from 24 to 36 °C.

3.2 Data Collection

Our system was tested on the data collected by the robot during several runs. The robot was operated in an indoor environment (a corridor and lab room at our institute). Persons taking part in the experiments were asked to walk in front of the robot while it performed different autonomous patrolling behaviours: corridor following (based on sonar readings) and person following (using information from the implemented tracker), or while the robot was stationary. At the same time, image data were collected with a frequency of 15 Hz. The person following behaviour was used to collect data for a single person and corridor following was used in the multi-person case. The resolution of the thermal and colour images was 320 × 240 pixels.

The person following behaviour of the robot was designed based on the implemented tracker. The robot starts first in a search mode, rotating continuously at the same time trying to detect a person in the thermal image. After a person is detected the robot tries to get closer and maintain a constant distance to the person. This is realised within an image-based control loop: the direction of the robot is adjusted so that the position of the person provided by the tracker remains in the middle of the thermal image. The velocity of the robot is determined

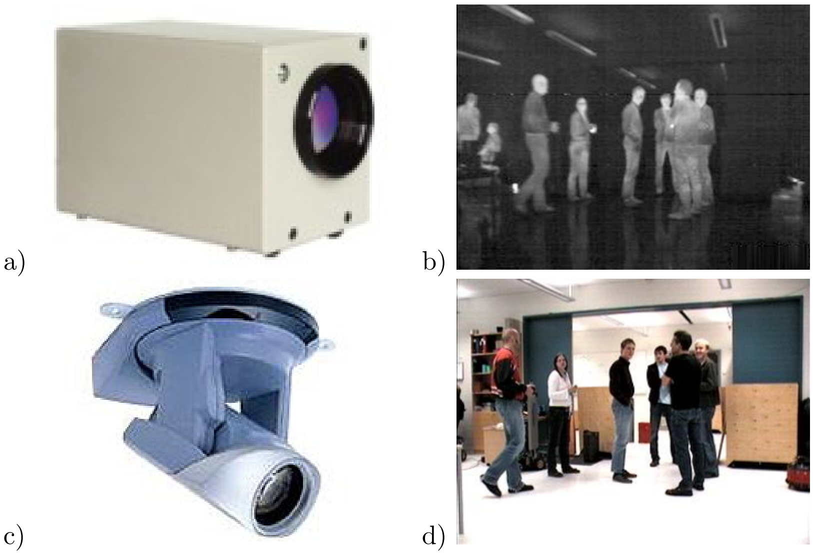

Figure 3.2: The vision-based people tracking system uses information from two cameras: a) thermal camera (Thermal Tracer TS7302, NEC) b) thermal image c) colour camera (VC-C4R, Canon) d) colour image.

by the height of the person which corresponds to the apparent distance between the robot and person. If the height is bigger than a specified threshold, or in other words the robot gets close enough to a person, the robot stops. Any change in the position of a person is immediately compensated by the control loop resulting in an appropriate action of the robot. The high frame rate of the system allowed for smooth operation in our office environments.

3.3 Ground Truth

Obtaining the ground truth in the case of video data is often a difficult, monotonous and labour demanding process. There have been attempts to improve and automate this process by using synthesised ground truth data [Black et al., 2003], systems performing fully automatic evaluation based on colour and motion metrics [Erdem et al., 2001] and approaches that first use some other algorithm to roughly select regions of interest that are refined later by hand [Khan et al., 2004]. We used a similar method in which results from a flood-fill segmentation algorithm were corrected afterwards by hand using the ViPER-GT tool [Doermann and Mihalcik, 2000].

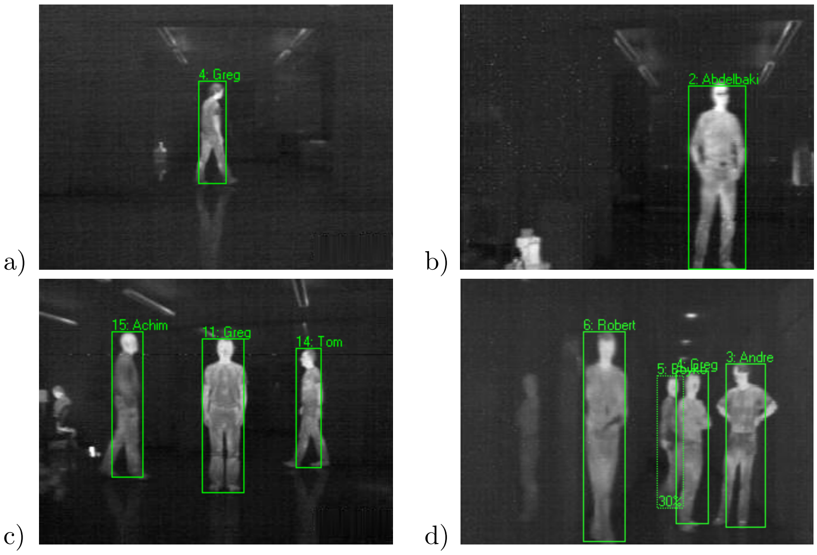

Figure 3.3: Ground truth data: a) single person - standing robot b) single person - moving robot c) multiple persons - standing robot d) multiple persons - moving robot; the percentage value in the bottom of a dashed bounding box indicates the amount of occlusion.

	Single person		**Multiple persons**	
ref. name	dataset1	dataset2	dataset3	dataset4
moving robot	-	+	-	+
frames	590	8431	3476	2131
frames total	1268	14573	4369	2400
tracks	8	45	42	11
detections	590	8431	6425	3831
occlusions	-	-	1130	159
max. persons	1	1	3	4

Table 3.1: Detailed information about the experimental data used in the thesis.

Optimally the ground truth data should consist of true values for each component of the state vector. Then the errors for each state variable could be specified to obtain an indicator of the performance of the tracker. However in our application obtaining such information for an ellipse model would be very difficult and could introduce significant errors. Instead we decided to consider only a bounding box around a person (see Fig. 3.3) in order to simplify the ground truth labelling process. As a result we were able to obtain a substantial amount of ground truth data (see Table 3.3). The top and bottom edges of a bounding box were determined from the contours of the head and feet while the sides were specified by the maximum width of the torso (without arms). The cases when persons appeared too close ($< 3m$) or too far ($> 10m$) to the robot were not taken into account. This type of ground truth information is just an approximation, and the quality of this process is affected by factors such as the naturally blurred appearance of a person in the image, noise caused by the movement of the robot and also the skill of the person labelling the data.

In addition the ground truth data for the multi-person case contain information about the amount of occlusion for each person (see Fig. 3.5d). The order of persons in the image was determined by the position of the bottom edge of the bounding box: closer persons that occlude other persons appear closer to the bottom edge of the image. Detailed information about the ground truth data including the number of persons, tracks, number of occlusions, robot's behaviour and reference name is presented in Table 3.3.

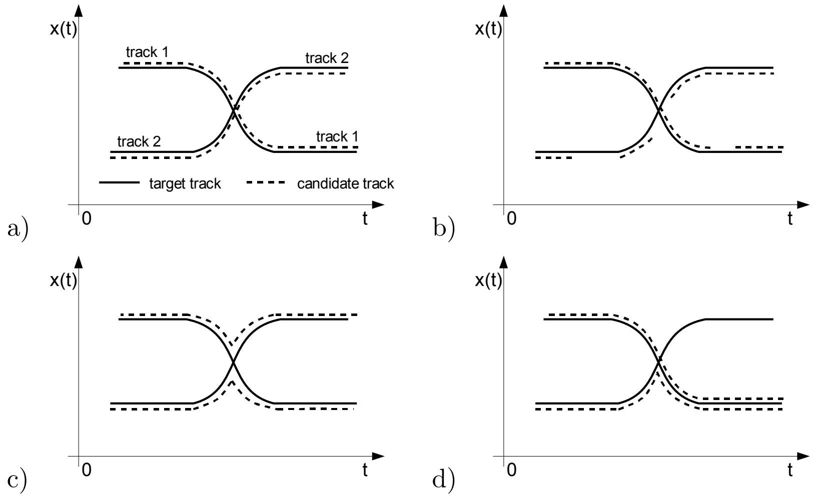

Figure 3.4: Typical tracking errors: a) target and correct candidate tracks b) track scattering c) track swapping d) track fetching.

3.4 Evaluation Metrics

The problem of evaluating tracking systems has been addressed recently by the computer vision community [Ferryman, 2000]. The consensus is that there is no single metric that could indicate sufficiently the quality of the entire system. For a proper evaluation it is important to use different metrics quantifying different performance aspects of the system. Examples of different metrics can be found in [Doermann and Mihalcik, 2000; Needham and Boyle, 2003; Black et al., 2003; Erdem et al., 2001; Smith et al., 2005a]. Having a good set of performance measures allows to optimise algorithm parameters, check performance of the tracker for different kinds of data, quantitatively compare different algorithms, support development of the algorithm, and decide upon trade-offs between different performance aspects.

The output from the tracking system is a set of tracks corresponding to each appearing person. A single track is a collection of estimated values of the state for the corresponding person over some period of time. To compare the output from the tracker with the ground truth data, we first transform the information provided by the elliptic model to match the assumed ground truth data described in the previous section. The size of the bounding box generated by the system was specified as $2 \cdot \mathit{width}$ and $3.5 \cdot \mathit{height}$ of the elliptic model, which is an approximation to the proportions of the human body. Bounding boxes from the ground truth data are referred to as *targets* and those from the tracker as *candidates*.

The output from the tracker should produce results as close to ground truth as possible. However, due to tracking errors, tracks deviate from their true paths, get scattered, missing or swapped (see Fig. 3.4). These phenomena occur especially in the case of tracking multiple persons. Therefore an association procedure is required to match the resulting candidate tracks to the target tracks. However the association procedure itself can produce inaccurate results in ambiguous situations. We decided to use a spatial association technique that produces more reliable results in cases of track fetching but is more sensitive to situations where tracks are swapped. In the spatial association procedure each frame is considered separately. It is assumed that only one candidate can match one target. As a measure of the distance between the target and candidate we used the overlap ratio. Moreover from these associated target-candidate pairs we consider only those pairs that overlap each other by more than a certain threshold (in our case $> 50\%$). Other candidates are considered as false alarms.

We use two kinds of metrics: detection metrics (counting persons) and localisation metrics (area matching). Each type of metric is further divided into three statistics: recall, precision and accuracy. Recall indicates true positives ("hits"), precision indicates the level of false alarms, and accuracy is a combination of both recall and precision. Such a set of metrics allows thorough testing of the properties and performance of the tracker and is similar in spirit to the metrics used in [Doermann and Mihalcik, 2000] and [Smith et al., 2005a]. Figure 3.5 shows an illustrative example of a possible outcome from the tracker, ground truth and a set of calculated metrics for one frame.

3.4.1 Detection Metrics

Detection metrics take into account the number of all correctly detected persons N_R in one frame and compare it with the number of targets N_T and number of candidates N_C. The final result is a weighted average of all frames.

- *Count Recall (CR):*

$$CR = \frac{N_R}{N_T}. \tag{3.1}$$

  This metric indicates how well an algorithm counts persons.

- *Count Precision (CP):*

$$CP = \frac{N_R}{N_C}, \tag{3.2}$$

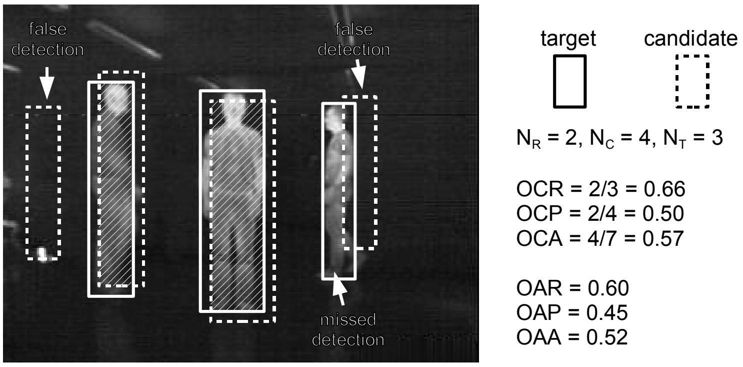

Figure 3.5: An illustrative example showing a possible output from the tracker together with different metrics calculated after applying the spatial association procedure described in Section 3.4. Only the hatched area is taken into account when calculating the localisation metrics.

is a counter-metric to CR and corresponds to the false detection ratio.

- *Count Accuracy (CA):*

$$CA = \frac{2 \cdot N_R}{N_T + N_C} \tag{3.3}$$

This metric is applied to check the accuracy of detection. It penalises both missing recalls (false negatives) and false alarms (false positives).

3.4.2 Localisation Metrics

These metrics express relations between areas corresponding to correctly detected candidates A_R, all candidates A_C and targets A_T. The final result is a weighted average of all frames.

- *Area Recall (AR):*

$$AR = \frac{|A_T \cap A_R|}{|A_T|}. \tag{3.4}$$

This metric measures the proportion of each target area covered by the corresponding correctly detected candidate. Recall is calculated for each target and averaged for the whole frame.

- *Area Precision (AP):*

  $$AP = \frac{|A_T \cap A_R|}{|A_C|}. \tag{3.5}$$

  This metric is a counter-metric to AR and it examines areas of correctly detected candidates with respect to all candidates instead. Precision is computed for each candidate and averaged for the whole frame.

- *Area Accuracy (AA):*

  $$AA = \frac{2 \cdot |A_T \cap A_R|}{|A_T| + |A_C|}. \tag{3.6}$$

  This metric measures how well an algorithm covers the target areas but also penalises for areas that are not covered.

All of the mentioned metrics are normalised to give percentages. If the nominator is greater than the denominator the result is set to 100. If the denominator is 0 then the result is undefined.

3.5 Conclusions

In this chapter we presented the set-up and methodology used in the experimental part of the thesis. Our robot *PeopleBoy* is well suited for tasks requiring interaction with persons. Colour and thermal cameras mounted on the robot provide the sensory information to the implemented people tracking system.

A proper evaluation of the tracking system is an important issue which seems to be neglected in many mobile robotic applications. We address this problem and present a set of metrics for evaluating the detection and localisation performance of the tracking system. This methodology is not limited to people tracking, but could also be used to evaluate more general object trackers.

Chapter 4

The Basic Tracking System

This chapter presents a people tracking system for mobile robots with a thermal camera that uses an elliptic contour model. First an efficient tracking method based on the particle filter is described for tracking a single person. Detailed information is provided about the measurement and dynamic model used by the particle filter. The section is concluded by experiments showing the general performance of the tracker. The parameters and performance of this tracker are used as a reference for further extensions of the system. We also show the performance of the system with respect to different system parameters to determine optimal values and the sensitivity of the tracker to their changes. Extensions to the proposed system for tracking multiple persons and incorporating colour information for increased performance and better occlusion handling are presented in the following chapters.

4.1 Particle Filter

Our system is based on a particle filter that provides an efficient solution to the estimation problem despite the high dimensionality of the state space. The particle filter allows to perform both the detection and tracking procedure simultaneously without exhaustive search through the entire state space. Moreover the measurements are incorporated directly into the tracking framework without any thresholding procedure that can cause loss of information.

The posterior probability $p(\boldsymbol{x}_t|\boldsymbol{z}_{1:t})$ of our system being in state $\boldsymbol{x}_t$ given a history of measurements $\boldsymbol{z}_{1:t}$ is approximated by a set of N

weighted samples such that

$$p(\boldsymbol{x}_t|\boldsymbol{z}_{1:t}) \approx \sum_{i=1}^{N} w_t^i \delta(\boldsymbol{x}_t - \boldsymbol{x}_t^i). \tag{4.1}$$

Each $\boldsymbol{x}_t^i$ describes a possible state together with a weight w_t^i which is proportional to the likelihood that the system is in this state. We use a standard Sampling Importance Resampling (SIR) filter that consist of a prediction, update and resampling step (see Algorithm 3 and Section 2.3.2 for more details about the SIR filter). In the prediction step a motion model (see Section 4.3) is applied to each particle, while the update step calculates new weights for particles by application of the measurement model (see Section 4.2). To avoid negative effects of sample impoverishment the SIR filter applies a resampling step (see Section 2.3.2). Starting with some initial distribution $p(\boldsymbol{x}_0)$ (in our case uniform) these three steps are repeated iteratively, providing at each time step t estimates $\hat{\boldsymbol{x}}_t$ of the state of the system as a weighted mean over all sample states:

$$\hat{\boldsymbol{x}}_t = \frac{1}{N} \sum_{i=1}^{N} w_t^i \boldsymbol{x}_t^i. \tag{4.2}$$

The resampling step is implemented using the systematic resampling algorithm (see Algorithm 4). With this approach the complexity of the resampling step is $O(N)$ with respect to the number of particles. After the resampling procedure all weights of particles are distributed uniformly.

Algorithm 3 A single iteration of the SIR filter

```
predict:
  - draw samples $\boldsymbol{x}_k^i \sim p(\boldsymbol{x}|\boldsymbol{x}_{k-1}^i),\ i = 1, \dots, N$
update:
  - calculate weights $\tilde{w}_k^i = p(\boldsymbol{z}_k|\boldsymbol{x}_k^i),\ i = 1, \dots, N$
  - normalise weights $w_k^i = \frac{\tilde{w}_k^i}{\sum_{j=1}^{N} \tilde{w}_k^j},\ i = 1, \dots, N$
re-sample:
  - generate a new set of samples $\{\boldsymbol{x}_k^{i^*}; i = 1, \dots, N\}$ by using the
    systematic resampling algorithm (see Algorithm 4)
```

4.2 Measurement Model

A thermal camera provides images where the human silhouette is very distinct, which makes the problem of detection easier. Moreover it is possible to segment persons easily despite the sensor movement, which

Algorithm 4 Systematic resampling after [Ristic et al., 2004]

```
$c_1 = w_k^1$
for i=2 to N do
  $c_i = c_{i-1} + w_k^i$
end for
$i = 1$
draw $u_1 \sim \mathcal{U}[0, N^{-1}]$
for j=1 to N do
  $u_j = u_1 + N^{-1}(j-1)$
  while $u_j > c_i$ do
    $i = i + 1$
  end while
  $\boldsymbol{x}_k^j = \boldsymbol{x}_k^i$
  $w_k^j = N^{-1}$
end for
```

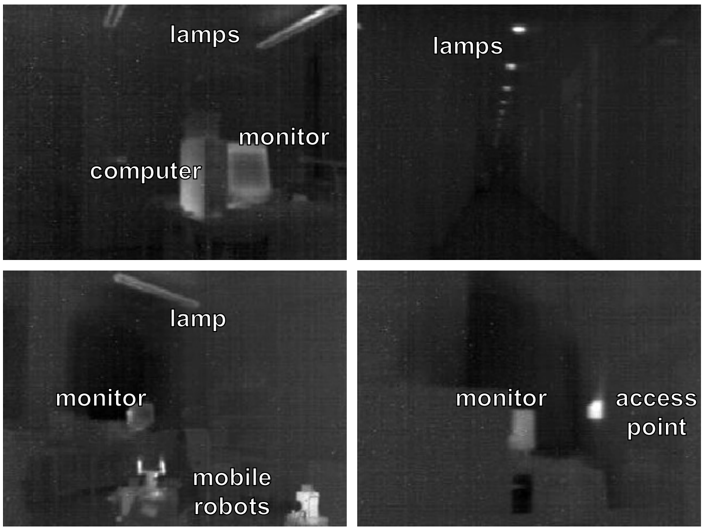

Figure 4.1: Other objects visible in the thermal image.

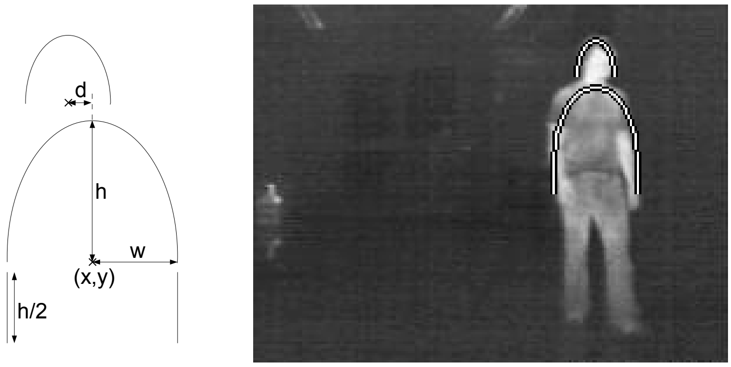

Figure 4.2: The elliptic measurement model in thermal images.

is a crucial property for a mobile robot system. However there are other objects that can be visible in thermal images such as lamps, radiators, monitors, robots, etc. (see Fig. 4.1). We propose an elliptic contour model that allows to distinguish persons from other objects in the environment. The contour measurement model used to estimate the position of a person in the image consists of two elliptic segments: one ellipse describes the position of the body part and the other ellipse measures the position of the head part. Thus we obtain a 9-dimensional state vector: $\boldsymbol{x}_t = (x, y, w, h, d, v_x, v_y, v_w, v_h)$ where (x, y) is the mid-point of the body ellipse with width w and height h. The height of the head is calculated by dividing h by a constant factor. The displacement of the middle of the head part from the middle of the body ellipse is described by d. We also model velocities of the body part as (v_x, v_y, v_w, v_h). The elliptic contour model can be seen in Figure 4.2.

To calculate the weight w_t^i of a sample i with state $\boldsymbol{x}_t^i$ we divide the ellipses into m different regions (see Figure 4.3) and for each region j the image gradient Δ_j^i between pixels in the inner part and pixels in the outer part of the ellipse is calculated. The gradient is maximal if the ellipses fit the contour of a person in the image data. A fitness value f^i for each sample i is then calculated as the sum of all gradients multiplied by a weight α_j for each region:

$$f^i = \sum_{j=1}^{m} \alpha_j \Delta_j^i. \tag{4.3}$$

The weights α_j sum to one and are chosen so that the shoulder parts have lower weight to minimize the measurement error that occurs due to different arm positions (see Figure 4.6). Typical fitness values for situations with and without a person in the thermal image are depicted

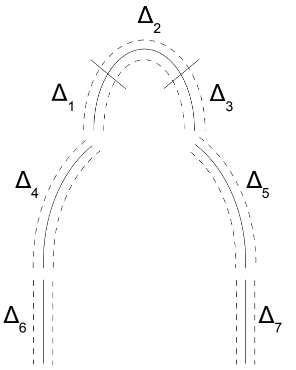

Figure 4.3: Elliptic model divided into 7 sections.

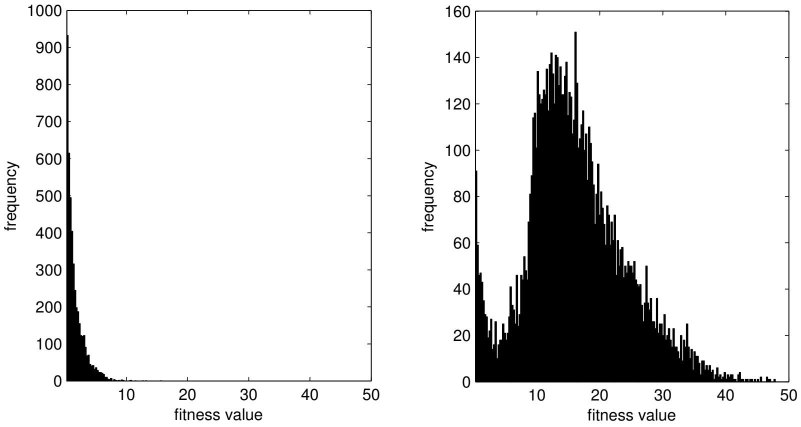

Figure 4.4: Histograms of particle fitness values for 30 selected frames containing no person (left) and a person (right).

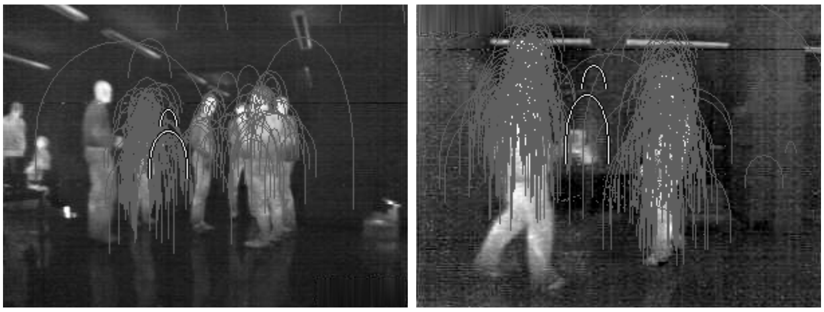

Figure 4.5: Situations with multiple persons leading to wrong estimates.

in Figure 4.4. The fitness value is finally scaled to values in $[0, 1]$ in order to represent a likelihood:

$$p_g(\boldsymbol{z}_t|\boldsymbol{x}_t^i) = \frac{\exp(\kappa \cdot (f^i - \theta))}{\exp(\kappa \cdot (f^i - \theta)) + \exp(\kappa \cdot (\theta - f^i))}, \qquad (4.4)$$

where θ denotes a fitness threshold and the value of κ defines the slope of the likelihood function. With a proper choice of the value κ too strong peaks in the observation model can be avoided. This kind of likelihood function was used in [Li and Wang, 2005] for visual tracking of objects. The weight update equation is then straightforwardly

$$w_t^i = p_g(\boldsymbol{z}_t|\boldsymbol{x}_t^i). \qquad (4.5)$$

When the mean gradient value calculated by Equation 4.4 is greater than 0.5 a person is considered to be detected. However in situations when the posterior probability distribution is highly multi-modal (e.g. multiple persons appearing on the scene) the weighted mean estimate can lead to a detection in wrong regions (see Fig. 4.5). To avoid such situations we also check the uncertainty of the estimate U_t [Karlsson and Gustafsson, 2001] given by equation:

$$U_t = \sum_{i=1}^{N} w_t^i(\boldsymbol{x}_t^i - \hat{\boldsymbol{x}}_t)(\boldsymbol{x}_t^i - \hat{\boldsymbol{x}}_t). \qquad (4.6)$$

If the value U_t is above some threshold we discard the corresponding detections.

Our contour model can be considered as a deformable template [Yuille and Hallinan, 1992] and is similar to the model used by Isard and Blake [Isard and Blake, 1998] for tracking people in grey scale images. However, they use a spline model of the head and shoulder contour which cannot be applied in situations where the person is far away or visible

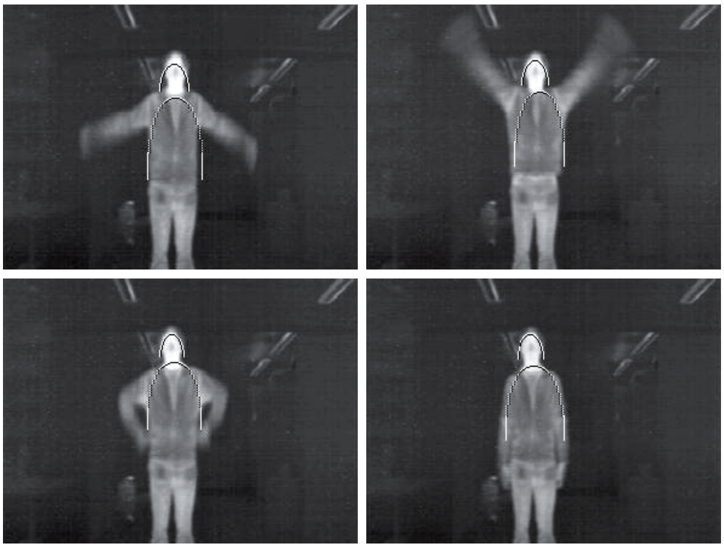

Figure 4.6: Tracking with different arm positions.

in a side view, because there will be no recognisable head-shoulder contour. The elliptic contour model is able to cope with these situations. Another advantage of the contour model used in this thesis is that it can be calculated very quickly due to the fact that we measure only differences between pixel values on the inner and outer part of the ellipse. Figure 4.7 shows the results of tracking a person under different views at different distances: starting with a frontal view the person turns to a side view and then a back view.

An unconstrained model as described so far would require that the whole 9-dimensional state space is explored. This would make it necessary to use an enormous number of particles. However this problem can be greatly alleviated by introducing constraints on some state variables. In our case we check limits of the ratio r_{wh} between the width and height of an ellipse model, the ratio r_{dw} between the state variable d (relative position of the head) and width of a sample, and finally the minimum and maximum values of the state variables y and h. If any of these requirements for a given sample is not fulfilled we discard the particle. Such a sample gets a very low chance to be selected in the next resampling step.

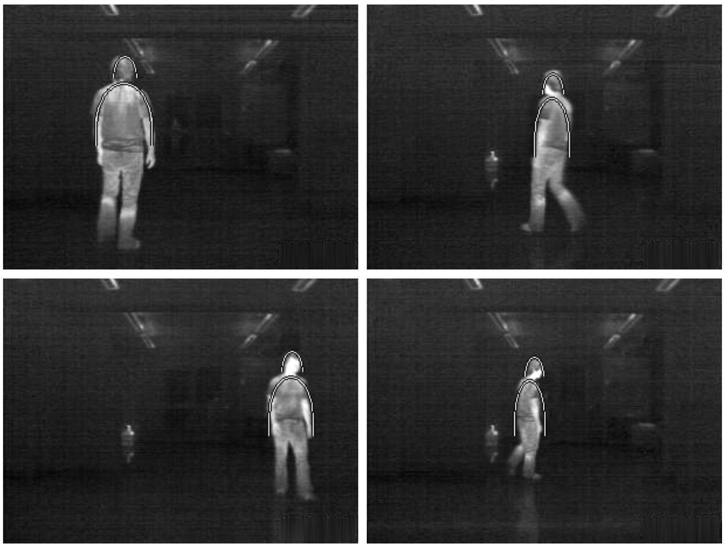

Figure 4.7: Tracking under different views using the elliptic measurement model.

4.3 Motion Model

In our work we consider the movement of a person in the image space. Optimally the motion model of the tracked object should be learned from data [Blake and Isard, 1998]. However the movement of a person, especially persons that interact with other people, can be unpredictable and it would be difficult if not impossible to learn such motion models. If learning is not possible other simpler models can be used such as the linear Gaussian model in the Kalman filter. However the Gaussian assumptions made about the motion are very often not met in visual applications. One possible way to deal with this problem would be to use a multiple model filter [Pavlovic et al., 2000] with a set of different motion models tracking a person in parallel and selecting the best motion model for a given situation. However this approach increases significantly the complexity of the tracker and introduces a decision problem: when to switch between different models.

Another important factor that introduces problems when selecting the motion model is the movement of the mobile platform together with its sensors. This movement further influences the apparent motion of a person in the image. The compensation of this movement would require complex transformations based on geometrical characteristics of the camera and robot, together with additional models or assumptions about the environment (e.g. planarity of the floor), etc. In such situations, when the range of activities of persons varies in addition to being observed from a mobile platform, it is better to use a more general motion model. In our system we use a random walk – a discrete realisation of Brownian motion – that can cope with the above mentioned problems to some extent and at the same time is relatively simple and efficient.

Random walk belongs to the family of first order stochastic processes, and thus depends only on the state of the system from the previous step. Such a motion model can deal with changes of position and shape, and despite some limitations (e.g. it does not allow modelling of oscillatory movements) it is a commonly used model in visual tracking [Blake and Isard, 1998].

The generative form of a Brownian motion model used in our work can be expressed as:

$$\boldsymbol{x}_t = A\boldsymbol{x}_{t-1} + B\boldsymbol{w}_k, \tag{4.7}$$

where $\boldsymbol{w}_k$ is a vector of independent random variables $\mathcal{N}(0,1)$ of the same dimension as the state vector $\boldsymbol{x}_t$. For our elliptic model matrix A has the following form:

$$A = \begin{bmatrix} I_4 & 0 & \delta t I_4 \\ 0 & 1 & 0 \\ 0 & 0 & I_4 \end{bmatrix}, \tag{4.8}$$

where δt is a time interval between two consecutive steps and B is a matrix specifying the amount of randomness introduced into the system.

4.4 Experiments

This section presents results showing the performance of the tracker based on the metrics introduced in Section 3.4. We used data collected for a single person in both a stationary and moving robot scenario (see Table 3.3 for more details).

Thanks to the evaluation metrics, we could optimise all system parameters based on the test data. As the performance criterion we chose an area accuracy metric that reflects the overall performance of the tracker. The influence of each parameter on the performance of the tracker was checked independently. Experiments for each parameter value were repeated 10 times with different random variations in the particle filter (run with $N = 1000$) for each trial. Obtained optimal values for the system parameters were as follows:

Parameter	**Value**
$\alpha_1, \ldots, \alpha_7$	0.18,0.18,0.18,0.07,0.07,0.16,0.16
κ, θ	0.125, 22
U_t threshold	400 pixels
B	$diag\{0, 0, 0, 0, 60, 30, 30, 22, 22\}$
limits	$0.1 < r_{wh} < 0.9$, $-1.5 < r_{dw} < 1.5$, $0.2 \cdot image\_height < y < 0.8 \cdot image\_height$, $h > 0.02 \cdot image\_height$

With this set of optimised parameters we ran experiments including all the data for the single person case. Each experiment was repeated 10 times with different random variations in the particle filter for each trial using $N = 1000$ particles. From the results presented in Figure 4.8 it can be seen that, in general, the tracking system based on the thermal appearance of a person decreases false detections especially well: the detection precision metric for both data sets was 94.61%. The performance of localisation is affected strongly by the fact that we are considering bounding boxes around a person, which results in low recall values especially in the case of distant and very close persons. The movement of the platform sometimes causes blurry images and increased noise, resulting in worse recall metrics compared to the stationary robot scenario. The difference in the area accuracy metric between the data sets for the stationary and moving robot was 1.22%. Using a standard paired t-test, differences between both data sets in recall and accuracy metrics were found to be significant ($p < 0.01$) and differences in precision metrics were not found to be significant at the same confidence level ($p < 0.01$).

We also checked the performance of the system with respect to different parameters of the particle filter including the number of particles and the percentage of samples used in the resampling step. As default values we chose 1000 particles and 20% of re-initialised samples. As in the case of parameter optimisation we chose an area accuracy metric as the performance criterion. Each experiment was repeated 10 times with different random variations in the particle filter for each trial.

Figure 4.9a shows the results for different numbers of samples. The quality of tracking increases with the number of samples and satisfactory results can be obtained with 300 particles. With more than 2000 samples the quality of tracking saturates and there is no significant improvement in the results. With less than 200 samples the tracker often loses tracks and the pose estimates become inaccurate.

To reduce the effects of degeneracy of particles we choose a fraction of all samples for re-initialisation in the resampling step. The best results can be observed for small values around 20% of re-initialised samples (see Fig. 4.9b). With more than 70% the performance of the tracker drops down due to the small number (less than 300) of samples effectively used in the filtering procedure. In the range between 10-50% the results

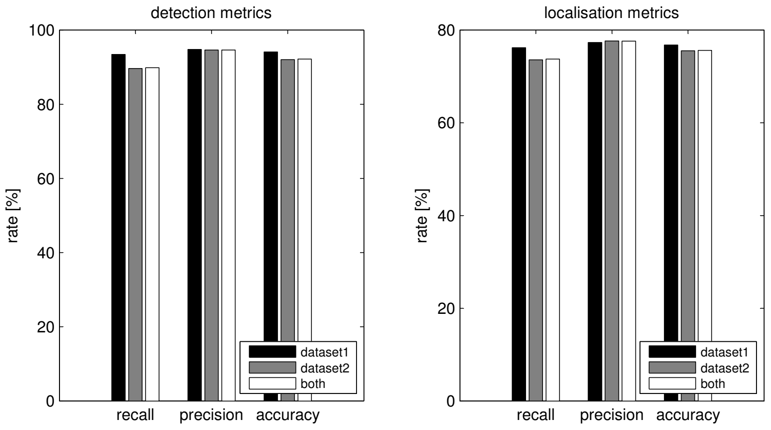

Figure 4.8: Detection and localisation metrics for tracking a single person in both stationary (*dataset1*) and moving (*dataset2*) robot scenario.

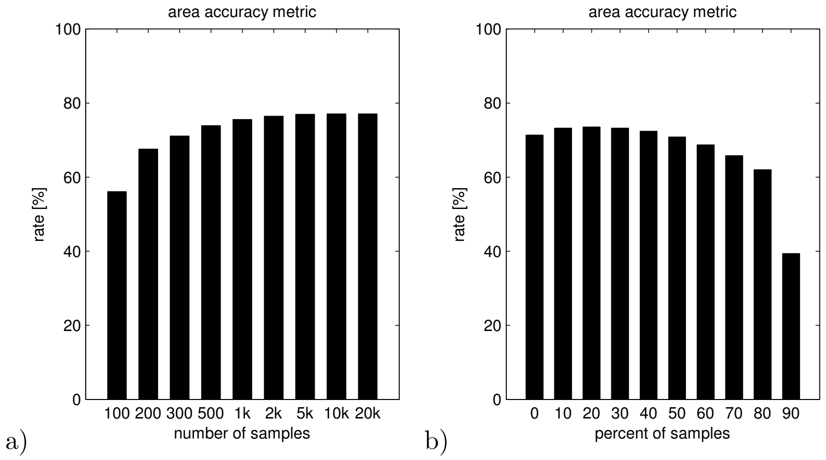

Figure 4.9: Performance measures for different system parameters: a) number of samples b) ratio of re-initialised samples.

Processing step	**Platform**	
	PeopleBoy Intel Pentium III 0.85 GHz [ms]	AMD Athlon XP 2.00 GHz [ms]
resampling	0.84	0.29
motion model	1.69	0.70
measuring	30.69	12.36
weight calculation	0.18	0.07
estimate uncertainty	0.33	0.10
total	33.73	13.52

Table 4.1: Average processing times of consecutive steps of the tracking algorithm calculated for $N = 1000$ samples.

indicate a low sensitivity of the tracker to changes in the ratio of re-initialised samples.

The use of the elliptic model results in low computational requirements. One iteration of the tracking algorithm using $N = 300$ particles on the PeopleBoy robot (Intel Pentium III processor, 0.85 GHz) requires only 11 ms, which is equivalent to a frame-rate of 90 Hz and leaves enough computational resources for other high-level tasks such as planning, navigation, face recognition, etc. Table 4.4 presents average processing times of different processing steps of the tracking algorithm calculated on the robot and modern PC for comparison.

4.5 Conclusions

In this chapter we presented an effective, fast and robust tracking system, allowing a mobile robot to track a single person in real-time. The sensory information is provided by a thermal camera, which enables tracking of people despite the movement of the robot. An efficient tracking method based on a particle filter allows to detect and localise a person without the need to scan the entire image. Measurements are incorporated directly into the tracking framework without thresholding of observations. The elliptic measurement model is fast to calculate and allows detection and tracking of persons under different views. An explicit model of the human silhouette effectively distinguishes persons from other objects in the scene. By contrast, the usual blob representations either make strong assumptions about the detected persons (the only objects in the scene) or involve filtering of non-person objects based on heuristics about the size and proportions, which may result in many false positives. Please note that the elliptic model also provides information about the position

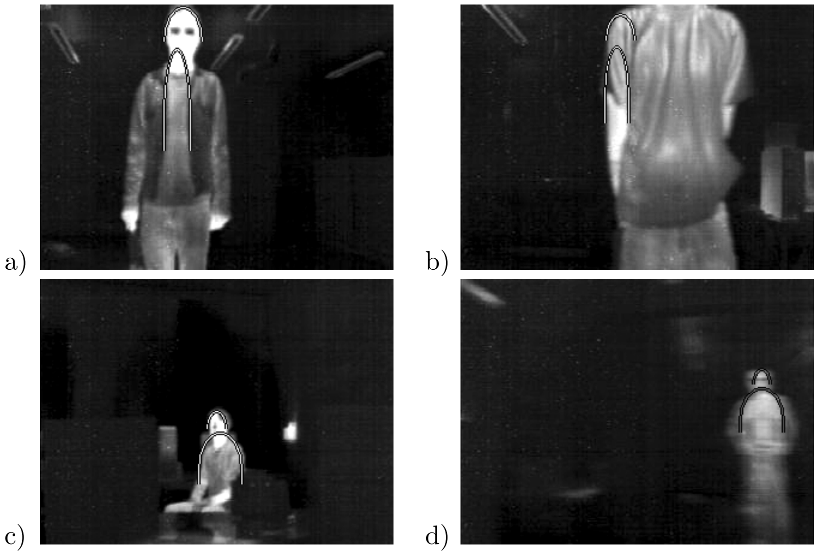

Figure 4.10: Selected problematic situations for an elliptic model: a) wrong estimate, b) incidental false detection, c) sitting person, d) blurred image (dark ellipse indicates an estimate below the detection threshold).

of the person's head, which could be used to provide an initial estimate for a face tracking system, for example.

The first shortcoming of the presented system is that it cannot track multiple persons. If a mobile robot is supposed to interact with different users and operate in crowded environments it is necessary to deal with multiple persons. An extension allowing our system to track multiple persons is described in the next chapter. Another issue is related to the elliptic contour model, which can detect and track persons only in the up-right pose, for which a head-shoulder contour is visible. This pose is most natural and common when interacting with a mobile robot in indoor scenarios. However such a model does not allow to track persons in some special poses: bending, lying, etc. In addition even though the system is able to detect infants and sitting persons, the provided estimates would be incorrect (see Fig. 4.10c). Moreover the contour of a person does not always provide accurate estimates about the position and size of the person (Fig. 4.10a) and sometimes it also happens that the pattern of the person's clothing seen on the thermal image is very similar to a human silhouette, resulting in false positives (Fig. 4.10b). The more sophisticated measurement model that incorporates additional thermal features based on integral images presented in [Treptow et al., 2006] improves the tracking performance in such situations. Fig. 4.10d presents the case when the image from the camera is too blurred to detect a person because of fast rotational movements of the robot. Incorporation of additional colour information could also help with some of above mentioned problems by focusing samples in the right regions of the image. An extension of the basic tracking system introduced in this chapter that incporporates colour information is presented in Chapter 6.

Chapter 5

Tracking Multiple Persons

In this chapter an extension to the people tracking system proposed in the previous chapter that enables detection and tracking of multiple persons is presented. A sequential detector that detects new persons appearing on the scene without the necessity of scanning the whole image is described. Later we present an efficient solution to the multi-person tracking problem based on independent tracking filters that enables tracking in real-time. The performance of the tracker is evaluated in the experiments and possible extensions and improvements are discussed.

We propose an efficient algorithm for tracking of multiple persons. The algorithm uses a factorial representation of the state space but in contrast to other classic tracking methods (i.e. MHT [Reid, 1979], JPDAF [Bar-Shalom and Fortmann, 1988]) does allow for incorporation of the raw measurements into the tracking procedure. Such a solution is better suited for vision-based tracking applications since a time consuming detection step can be avoided. It also removes the loss of information caused by the thresholding procedure in the detection step, which is very important for mobile systems where the movement of the platform introduces a significant amount of noise. This approach, called *unified tracking* [Stone et al., 1999], allows for association between raw measurements and target tracks implicitly within the Bayesian framework, and the processes of detection and tracking are carried out simultaneously. A more thorough discussion of the unified approach to tracking is presented in Section 2.4.3.

5.1 Sequential Detector

We investigate a unified tracking approach that detects new persons incrementally as they appear while maintaining existing tracks of persons. Our system uses a set of independent particle filters to track different persons. We denote by $\boldsymbol{X}_t = \{\boldsymbol{x}_t^{(1)}, \ldots, \boldsymbol{x}_t^{(M)}\}$ the combined state of the tracker where M indicates number of persons.

To assign new filters to new persons we use a sequential detector, i.e. a separate tracking filter with the state $\boldsymbol{x}^{(D)}$ consisting of a set of N_D randomly initialised particles. These particles are used to "catch" a new person entering the scene. To avoid multiple detections in the same or similar regions, the weight of detection particles is penalised by a factor $\psi_D < 1$ in cases where particles cross already detected areas. The weight update equation 4.5 for the i^{th} detection particle is modified to

$$w_t^{(D),i} = p(\boldsymbol{z}_t|\boldsymbol{x}_t^{(D),i})\psi, \tag{5.1}$$

where $\psi = \psi_D$ if particle i overlaps with other detected regions and $\psi = 1$ otherwise. In this way, already existing filters naturally limit the search space for the detector. The weights of the detection particles are normalised after the update procedure.

The penalty factor ψ_D allows us to specify how close a new detected region can be to other regions. In the extreme case where $\psi_D = 0$ only well-separated persons can be detected. However in crowded scenes higher values are more appropriate. The term ψ actually depends on both the detector state $\boldsymbol{x}_t^{(D)}$ and the combined state of the tracker $\boldsymbol{X}_t$. Therefore it could assume a more sophisticated form, for example, taking into account both the position and velocities of the particles. The simplified form of the penalty factor ψ used in our case resulted in a very computationally efficient method. Moreover from our investigations a more complex form (i.e. amount of overlap between particle regions) did not result in a significant increase in performance of the detector. This approach in the worst case requires $N_D \cdot M$ tests for overlap, where M stands for the number of already detected persons. If a particle overlaps with more than one region it is penalised only once. Detection occurs – similar to the single person tracker – when the mean gradient value calculated by Equation 4.2 is greater than 0.5 and the sample uncertainty specified by Equation 4.6 is above a specified threshold. Then the particles from the detector are used to initialise a new tracker before being re-initialised in order to detect the next new person. Figure 5.1 presents an example case when three persons are successively detected.

Such a design of the detector causes problems in cases when a few people appear in the image at the same time. Since they can only be detected sequentially, detection delays can occur. On the other hand, a sequential detector allows to detect each person in constant time, which is a very important consideration in real-time applications.

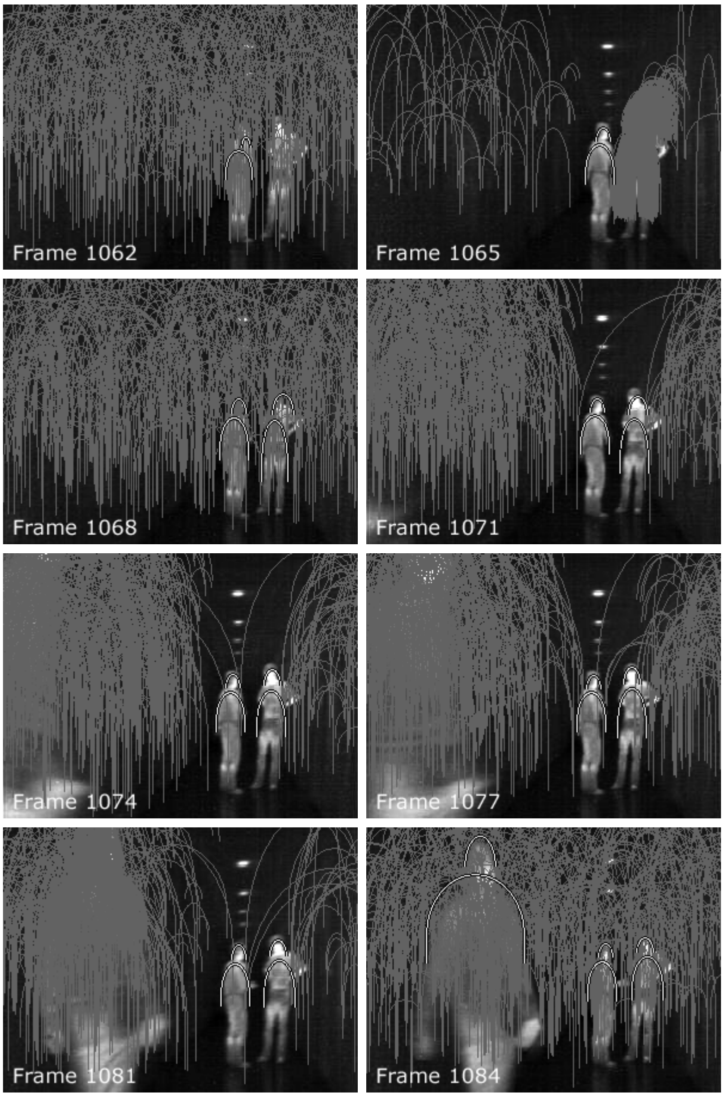

Figure 5.1: The sequential detector.

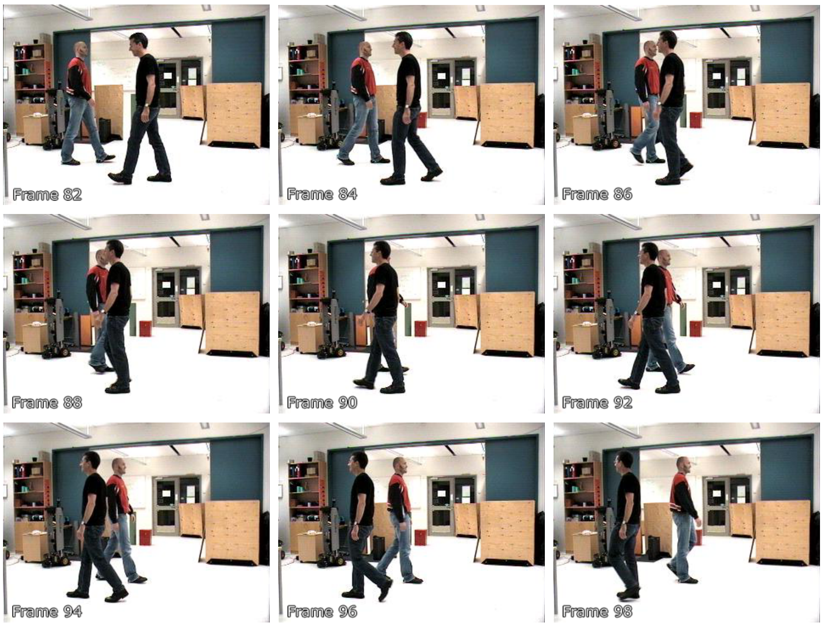

Figure 5.2: Two persons passing by. There is enough kinematic information to solve the tracking problem.

5.2 Heuristic Approach to Track Multiple Persons

The most advanced multi-target tracking algorithms take into account correlations between the targets and use a joint state space representation. This allows to solve the tracking problem including occlusions between targets implicitly within the tracking framework. However it is required that there is enough other information, e.g. from the target kinematics, to distinguish between different targets and assign the right measurement to the right object. An example of this situation is given in Figure 5.2 where two persons pass one another with significantly different velocities. However this is only one possible scenario of peoples' interaction. Persons often stop and interact in different ways: talking, exchanging items, trying to avoid each other, etc. In such situations kinematic information is usually not sufficient and sometimes even misleading. Moreover, in our system we consider a moving platform that further complicates the situation: the movement of the platform combined with the movement of persons can be unpredictable and result in complex apparent motion of persons in the image.

The use of complex multi-target tracking algorithms (e.g., [Kreucher, 2005],[Orton and Fitzgerald, 2002]) would be justified in situations where we could model all possible behaviours of a person. However, this is not possible in our application. The use of such algorithms, where the computational complexity grows exponentially with the number of persons and observations, cannot be fully justified in this case, especially when considering a system designed to work in real-time. Thus we have adopted a computationally simpler solution based on the factorial representation of the state space.

In our system an independent particle filter is assigned to each detected person, so that the total number of particles used by the tracker is $N_D + M \cdot N$, where M is the number of persons detected by the detector described in the previous section. Such a solution is computationally inexpensive and appropriate for on-line applications, but suffers in cases when tracked persons are too close to one another. To reduce these problems we try to explicitly model interactions between persons and prevent the corresponding trackers from being to close to each other. This is realised by penalising the weights of particles that intersect with areas corresponding to other detected persons. The weight update equation for a single tracking filter j is similar to Equation 5.1 used in the detector

$$w_t^{(j),i} = p(\boldsymbol{z}_t | \boldsymbol{x}_t^{(j),i})\psi, \tag{5.2}$$

but this time

$$\psi = e^{(-\rho g_{ij})}, \tag{5.3}$$

where g_{ij} expresses the amount of overlap between particle i and region j multiplied by a penalty factor ρ. The penalty factor ρ allows to specify the "strength" of interactions between persons and the amount of handled partial occlusions (see Fig. 5.3). When $\rho = 0$ no penalty is applied ($\psi = 1$), which leads to situations where stronger filters (i.e. with the more peaked likelihood) "fetch" the others. Target "fetching" (also called "hijacking" [Khan et al., 2004]) occurs when two or more tracking filters in close proximity are attracted by stronger evidence in one region. This causes the filters tracking targets with weaker evidence to collapse into one region, resulting in tracking errors. It appears as if the weaker filters are "fetched" by the stronger filters.

Very high values of ρ ($\psi \approx 0$) do not allow any overlap between persons and weaker filters quickly disappear. The optimal value for the penalty factor ρ depends on a trade-off between the amount of handled overlap (partial occlusions) and the ability to deal with target fetching. The proposed form of the penalty factor ψ is just one possibility, and could also depend on the velocity of persons. However, as previously discussed, velocity information about interacting persons can be unreliable. This solution is similar to the interaction model proposed by [Khan et al., 2004], where the authors propose a Random Markov Field for this pur-

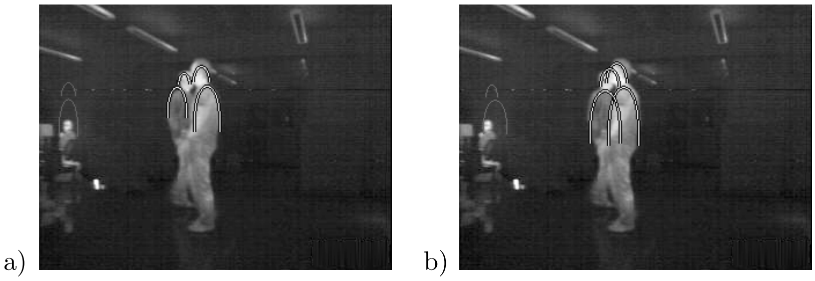

Figure 5.3: Different values of ρ parameter allow to specify the strength of interaction between filters: a) $\rho = 10$ b) $\rho = 1$.

pose, using a joint state space representation. The proposed treatment of interactions has the drawback that in the case of occlusions weaker filters disappear. In the worst case, this approach requires $N \cdot M \cdot (M-1)$ calculations of the amount of overlap between the particles. The overlap calculations for the rectangular regions used in our case can, however, be efficiently implemented. If a particle overlaps with more than one region it is penalised only once.

After weight normalisation we check if the filters still track the persons. When the mean gradient value of a tracking filter calculated by Equation 4.2 is smaller than 0.5 or the sample uncertainty specified by Equation 4.6 is above a specified threshold the filter is deleted.

5.3 Experiments

This section presents results showing the performance of the tracker when tracking multiple persons based on the metrics introduced in Section 3.4. We used data collected for multiple persons both in a stationary and moving robot scenario (see Table 3.3 for more details). Each experiment including all the data was repeated 10 times with different random variations in the particle filter for each trial using $N_D = N = 1000$ particles.

To determine optimal values of the system parameters we used a similar approach as described in the previous chapter choosing an area accuracy metric as the performance criterion. The influence of each parameter on the performance of the tracker was checked independently repeated 10 times with different random variations in the particle filter for each trial run with $N_D = N = 1000$ particles. The obtained optimised values for the system parameters were as follows:

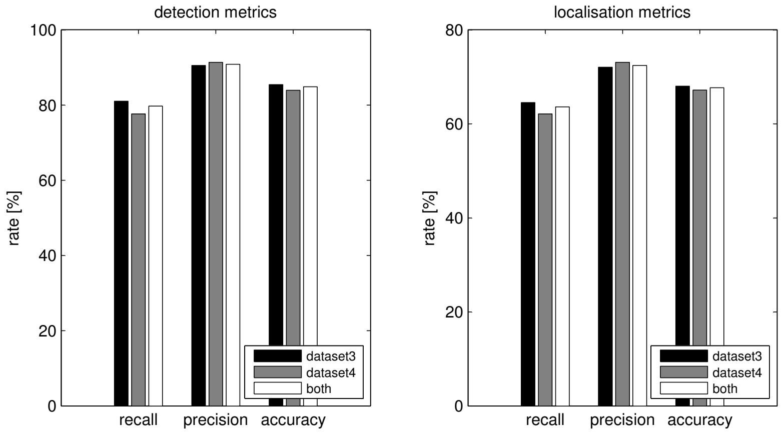

Figure 5.4: Detection and localisation metrics for tracking multiple persons in both stationary (*dataset3*) and moving (*dataset4*) robot scenario.

Parameter	Value
ψ_D	0.01
ρ	2

The other parameters for the measurement and motion model were kept without change.

The results in the case of tracking multiple people are shown in Figure 5.4. In comparison with the single person case there is a significant deterioration in the performance of the system, indicated by lower recall values caused mainly by long crossings and occlusions of persons. The movement of the platform has also a negative influence causing a reduction of 3.40% in the recall detection metric when compared with the stationary robot case. There was also some minor influence on the recall metrics due to the sequential nature of the detector. Despite this, the tracker still keeps the number of false detection low, as indicated by the high precision values. The slightly better results for the precision metrics in the moving robot case can be explained by a lower number of occlusions in this data set (see Table 3.3). The localisation accuracy metrics for both data sets was 67.70%. Using a standard paired t-test, differences between both data sets in all cases were found to be significant ($p < 0.01$).

We also checked how the number of particles assigned to each filter influences the performance of the tracker. Fig. 5.5a presents results for different number of samples N. During this experiment the number of

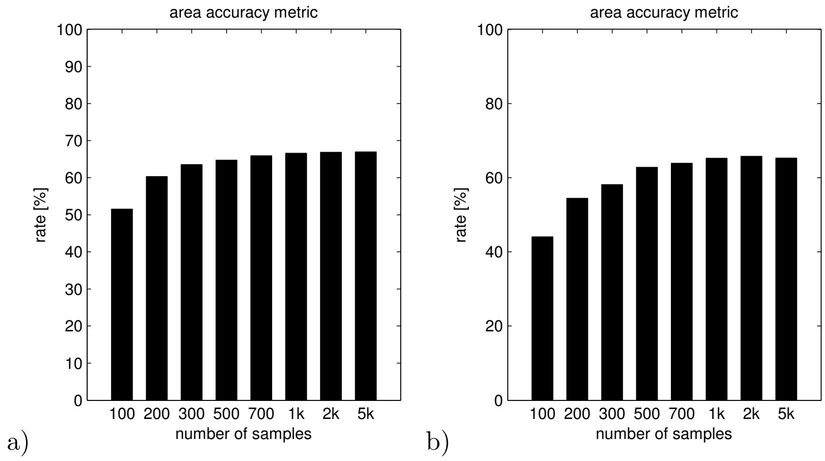

Figure 5.5: Performance measures for: a) number of samples assigned to each individual tracker b) adaptive version assigning an equal fraction of samples to each individual tracker.

samples in the detector filter was kept constant ($N_D = 1000$). The tracker performance is satisfactory even with 200 particles per filter and with more than 1000 samples the performance (around 66%) does not change any more. Each new tracker naturally limits the search space of the other trackers. This fact leads to the conclusion that the number of samples per filter can be reduced every time a new filter is initialised without affecting the performance of the tracker. To check this claim we ran an experiment with an adaptive number of samples where the total number of samples is kept constant and only a fraction of the samples ($\frac{N}{M}$) is assigned to each filter. The results are presented in Figure 5.5b. It can be seen that even with $N = 500$ of total samples the tracker performance is satisfactory (62.67%). This approach allows to limit substantially the total number of samples used by the tracker especially when there are more people appearing on the scene.

To check the performance of the detector we fixed the number of particles used by each tracking filter to $N = 1000$ samples and varied the number of detector particles (see Fig. 5.6a). The overall performance of the tracker becomes satisfactory with about 500 particles (65.77%). Using more than 1000 samples does not result in a significant increase in the performance. The average number of frames needed for detection of a single person was 4, corresponding to a time of 0.26 s with the specified frame rate of 15 Hz.

The computational requirements for all tracking filters are exactly

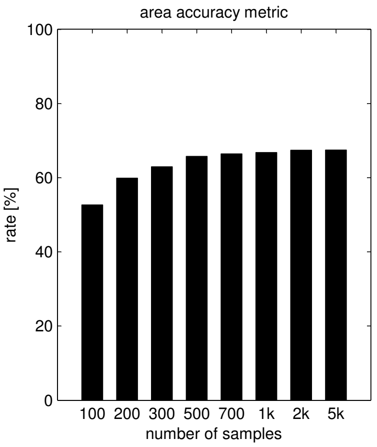

Figure 5.6: Performance measures for different number of samples used in the detector filter.

the same as for a single filter. Additional processing time is needed by the detector to check intersections between particles ($N_D \cdot M$ checks in the worst case) and the amount of overlap between particles of the tracking filters ($N \cdot M \cdot (M-1)$ overlap calculations in the worst case). In Table 5.1 average processing times needed for calculating 1000 particles for different numbers of tracked persons are presented. The number of particles used during these tests was set to $N_D = 1000, N = 1000$. To get the average time needed for calculating 1000 samples the total time needed for the system was normalised by the number of individual trackers (including detector) used at a given time. When one person is tracked, the system requires around 8% more time when compared to the time needed just by the detector. This time is required to calculate intersections between detector and filter particles. When more than one person is tracked the additional time is used to check the amount of the overlap between individual filters. In the case of four people it requires around 36% more time per 1000 samples.

5.4 Conclusions

In this chapter we presented an effective and computationally feasible extension to the basic tracking system allowing a mobile robot to track multiple persons in real-time. The unified tracking approach detects new persons incrementally as they appear while maintaining existing tracks of persons. Different persons are tracked by a set of independent particle filters. The computational efficiency of the tracker was obtained by an

Number of persons	**Platform**	
	PeopleBoy Intel Pentium III 0.85 GHz [ms]	AMD Athlon XP 2.00 GHz [ms]
detector only	33.73	13.52
1	36.40	14.59
2	39.68	15.90
3	43.72	17.52
4	46.17	18.51

Table 5.1: Average processing times for each tracker needed to calculate 1000 samples depending on the number of simultaneously tracked people.

approach that directly manipulates the weights of the particles.

The sequential detector enables detection of new persons in constant time. However it introduces some detection delays depending on the number of simultaneously appearing people. The other major drawback of the tracker is that it cannot cope properly with occlusions. The weaker filters disappear in case of an occlusion even though the tracker quickly recovers as soon as a person is visible again. Proper occlusion handling would allow to track persons despite occlusions. The next chapters therefore present a further extension to the proposed tracker, incorporating additional colour information, that is used to detect cases in which occlusions occur and later to explicitly reason about them.

Chapter 6

Incorporating Colour Information

This chapter presents an extension to the proposed system to incorporate colour information provided by a colour camera mounted on the robot. This approach improves data association and increases the robustness and performance of the tracker. First we present the solution to the correspondence problem between the two cameras. Later a compact and efficient colour representation is described together with an adaptive appearance model. We also present a rapid way to calculate the rectangular colour features used to measure similarity of the region of interest to the appearance model. We also show how to fuse thermal and colour information together. The section is concluded by experiments comparing the performance of the tracking system with and without colour information, using different colour representations and different colour spaces.

6.1 Colour Model

6.1.1 Correspondence Between Cameras

Incorporating colour information into the tracking system requires thermal and colour images to be aligned. Since the baseline between these two cameras is relatively small compared to the distance from the sensors to the persons in the scene, it is possible to model the displacement between the cameras by means of an affine transform. If we describe pixel coordinates in the thermal image as (u, v) and pixel coordinates in the colour image as (x, y) then the affine transform between the thermal and colour images can be described as

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} p_1 + p_2 u + p_3 v \\ p_4 + p_5 u + p_6 v \end{bmatrix}, \tag{6.1}$$

where $\boldsymbol{p}_{aff} = \{p_1, p_2, p_3, p_4, p_5, p_6\}$ is an unknown parameter vector. Usually values of this parameter should be determined by some automatic calibration procedure. Note, however, that in our case we have two cameras with different modalities, meaning that standard stereo-camera calibration methods cannot be directly applied. A robust method for multi-sensor alignment based on multi-scale directional derivative filters, which allows to determine the parameter values $\boldsymbol{p}_{aff}$ automatically was proposed by [Irani and Anandan, 1998]. We used a simpler approach and calculated the parameter values from data provided by the camera manufacturers, and in our case $\boldsymbol{p}_{aff} = \{-0.15, 1.20, 0.00, -0.13, 0.00, 1.20\}$. A visualisation of the aligned images can be seen in Figure 6.1.

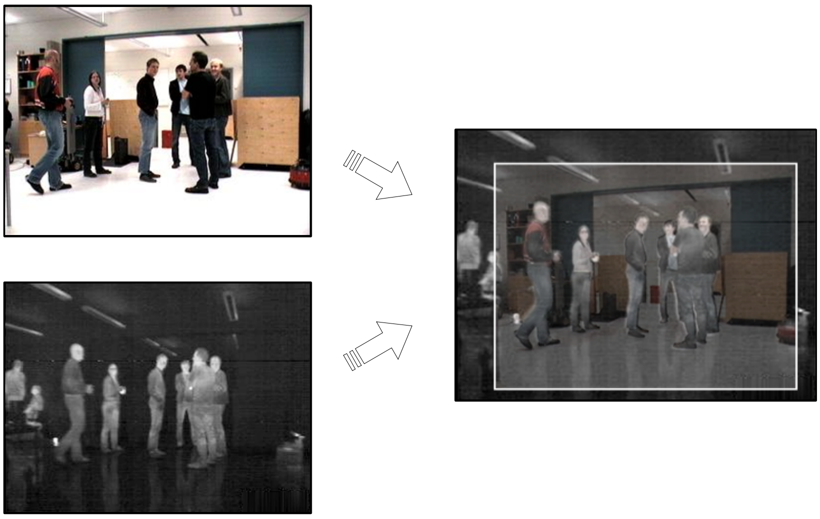

Figure 6.1: Images from the colour and thermal cameras (on the left side of the figure) are aligned by affine transformation. The resulting combined image is shown on the right side of the figure. The white frame specifies the border of the colour image.

6.1.2 Colour Representation

The most popular representation of the colour distribution in visual tracking applications is based on colour histograms [Comaniciu et al., 2003; Perez et al., 2002]. Colour histograms are relatively easy to construct and compare using some of the proposed similarity measures [Swain and Ballard, 1991]. Despite recently proposed efficient algorithms [Porikli, 2005] they are still computationally demanding, especially if a significant number of measurements is required. Choosing the number of bins in the histogram is also a problematic issue. Instead, we use an efficient colour representation proposed in [Stricker and Orengo, 1995] where the authors used the first three moments (mean, variance and skewness) of the colour distribution. The similarity measure based on this representation was shown to be more effective than histogram based methods (e.g., [Swain and Ballard, 1991]) in the domain of image indexing. We use this representation in our system to obtain a very compact model, which can be quickly calculated and evaluated as we show later in this section. In the experimental part we compare results using the proposed approach with a solution based on colour histograms.

Colour distribution does not contain any information about the spatial layout of the colour. However the head, torso and legs of a person can usually be distinguished as differently coloured, and this additional information helps to build a more robust and precise colour model. One possible approach to incorporate the spatial layout of the colour distribution uses a colour correlogram, which is a histogram representation containing spatial relations between pixels [Huang et al., 1997], but the computational complexity of this approach is even higher than in the case of colour histograms. We use an approach that divides the region corresponding to a person's body into different rectangular sub-areas from which we calculate the colour statistics (see Fig. 6.2b). The position and size of these regions is determined from the information provided by the elliptic contour model. The rectangular shape of these regions allows application of a fast method for calculating the colour statistics. A similar approach was used in [Han et al., 2005], where a rectangular region of the person's body was divided into n different parts.

The hardware of our robot provides images in the RGB colour space. In many computer vision applications other colour spaces are used e.g., the HSV (Hue-Saturation-Value) representation where the colour is represented in 3D-polar coordinates [Perez et al., 2002; Okuma et al., 2004; Zhao and Tao, 2005]. With this approach the brightness and saturation are separated and the influence of changing lighting conditions can be reduced. However the additional time needed for conversion between colour spaces and calculation of circular statistics required by the HSV moment based model makes use of this representation questionable in real-time applications.

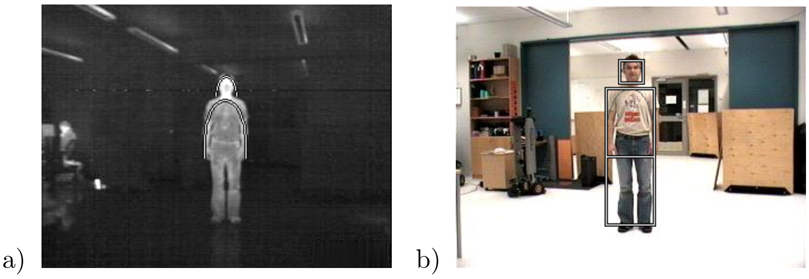

Figure 6.2: Rectangular features: a) thermal image b) colour image with regions corresponding to different body parts from which colour information is extracted.

6.1.3 Adaptive Colour Model

Our appearance model of a person is based on colour moments, thus being very compact yet providing enough information to improve tracking. This model is created every time a new detection occurs, i.e. a new track is initialised on the thermal image. By using the affine transform described earlier we are able to determine the region corresponding to a person in the colour image (see Fig. 6.2). From the three rectangular regions corresponding to the person's head, torso and legs we collect colour statistics $\boldsymbol{c}_t$ about the first three moments (m_1, m_2, m_3) for three colour channels (R, G, B). Thus we obtain a feature vector $\boldsymbol{c}_t$ of size $3 \times 3 \times 3 = 27$. To make the model more robust to changing light conditions we adapt it using the new information as long as a person is tracked. In our current implementation we store information about colour statistics from the last n_k frames and calculate their mean value

$$\boldsymbol{c}_t^\star = \frac{1}{n_k} \sum_{i=t-n_k-1}^{t-1} \boldsymbol{c}_t. \tag{6.2}$$

The size of the parameter n_k influences the robustness and adaptivity of the colour model.

We use the L_2 metric, i.e. Euclidean distance, to measure the distance between the model $\boldsymbol{c}_t^\star$ and region of interest $\boldsymbol{c}_t$ of the form

$$d_t = \sqrt{(\boldsymbol{c}_t^\star - \boldsymbol{c}_t)^2}. \tag{6.3}$$

Finally, the likelihood model for colour information can be described as

$$p_c(\boldsymbol{z}_t|\boldsymbol{x}_t) = exp\left(-\lambda d_t^2\right), \tag{6.4}$$

where λ is a parameter that allows to specify the shape of the colour likelihood. Since λ scales the distance, higher values of λ mean that

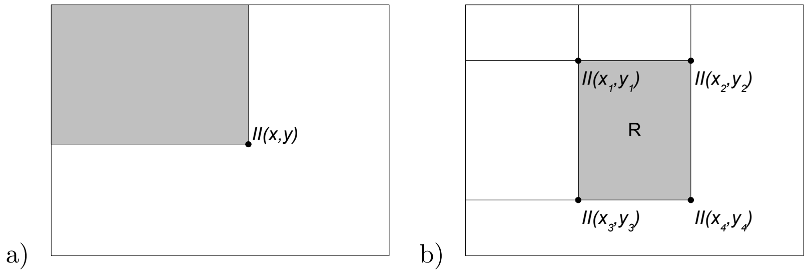

Figure 6.3: Integral image: a) creation of the integral image b) calculating the sum over a rectangular area using the integral image.

the colour-based likelihood model is more peaked, thus having more importance when combined with the gradient information from the ellipse model.

6.1.4 Rapid Rectangular Colour Features

The simple features based on the colour moments can be rapidly calculated using an integral image representation [Viola and Jones, 2001]. The main idea is not to work directly on image intensities but to use some intermediate representation. Each pixel of the integral image contains a sum of all pixels in the rectangular area determined by the left-upper borders of the image and the pixel coordinates. The integral image II can be created from the original image I using the following formula:

$$II(x,y) = \sum_{u \leq x, v \leq y} I(u,v), \tag{6.5}$$

where (x,y) represents the pixel coordinates in the integral image II and (u,v) the pixel coordinates in the original image I. This can be easily realised in one pass over the original image I as follows:

$$\begin{aligned} S(x,y) &= S(x,y-1) + I(x,y) & (6.6) \\ II(x,y) &= II(x-1,y) + S(x,y), & (6.7) \end{aligned}$$

where $S(x,y)$ is the cumulative row sum. The resulting integral image II is one pixel wider and higher since $S(x,-1) = 0$ and $II(-1,y) = 0$. Such a representation can easily be extended to colour images if we treat each colour channel independently. In the case of the RGB colour representation the resulting integral image consists of three integral images: II_R, II_G, II_B.

Having obtained the integral image, only four basic operations are required to calculate a sum over any rectangular region in the original image; if we consider the situation presented in Figure 6.3b the sum over the rectangular region $R = II(x_4, y_4) + II(x_1, y_1) - (II(x_2, y_2) + II(x_3, y_3))$. This makes the method very fast, and in particular, allows the calculation of the sum of any size region in constant time. Grey-scale features based on integral images were also used in [Treptow and Zell, 2004] to track an ordinary football and colour features based on three integral images for the different colour channels were used in [Han et al., 2005] to track different objects including people.

The estimates for the first three moments of the colour distribution can be obtained by means of k statistics. The k statistics can be easily calculated by using sums of the rth powers of the colour data:

$$S_r = \sum_{x=1}^{n_x} \sum_{y=1}^{n_y} I^r(x, y), \tag{6.8}$$

where $I(x, y)$ is a pixel value of the image of size $n_x \times n_y$. Note that this calculation should be performed for each colour channel. For each power of the sum S_r we have to calculate one separate colour integral image II. Then it is easy to obtain the first three k-statistics using the following formulas:

$$\begin{aligned} k_1 &= S_1/n, && (6.9) \\ k_2 &= \frac{nS_2 - S_1^2}{n(n-1)}, && (6.10) \\ k_3 &= \frac{2S_1^3 - 3nS_1S_2 + n^2S_3}{n(n-1)(n-2)}, && (6.11) \end{aligned}$$

where $n = n_x \times n_y$. Finally the normalised values of the estimates for the mean m_1, variance m_2 and skewness m_3 can be obtained as

$$\begin{aligned} m_1 &= k_1, && (6.12) \\ m_2 &= k_2/k_1, && (6.13) \\ m_3 &= k_3/k_2^{\frac{3}{2}}. && (6.14) \end{aligned}$$

The normalisation is performed to balance the influence of each moment on the final score.

6.2 Fusing Thermal and Colour Information

If we assume that the likelihoods for the gradient model $p_g(\boldsymbol{z}_t|\boldsymbol{x}_t)$ (Equation 4.4) and colour model $p_c(\boldsymbol{z}_t|\boldsymbol{x}_t)$ (Equation 6.4) are independent then the data fusion can be realised by taking a product of these two likelihoods as

$$p(\boldsymbol{z}_t|\boldsymbol{x}_t) = p_g(\boldsymbol{z}_t|\boldsymbol{x}_t)p_c(\boldsymbol{z}_t|\boldsymbol{x}_t). \tag{6.15}$$

The parameters κ, θ (gradient model) and λ (colour model) specify the shape of the gradient and colour likelihood functions, thus specifying the importance of the respective features.

When a person is not detected, a colour model cannot be built and only gradient information can be used to update the weight of the particles of a single tracking filter j as

$$w_t^{(j),i} = p_g(\boldsymbol{z}_t|\boldsymbol{x}_t^{(j),i})\psi, \ i = 1, \ldots, N, \ j = 1, \ldots, M. \tag{6.16}$$

However as soon as a person is detected the colour model can be created and the weight update equation changes to:

$$w_t^{(j),i} = p_g(\boldsymbol{z}_t|\boldsymbol{x}_t^{(j),i})p_c(\boldsymbol{z}_t|\boldsymbol{x}_t^{(j),i})\psi, \ i = 1, \ldots, N, \ j = 1, \ldots, M. \tag{6.17}$$

Please note that the sequential detector relies only on gradient information.

If the assumptions about the independence of measurements hold then Equation 6.15 can be easily extended to combine many other features. A similar solution was used in work of [Rasmussen and Hager, 2001; Hayman and Eklundh, 2002; Serby et al., 2004].

6.3 Experiments

This section presents results showing the performance of the tracker when tracking multiple persons with additional colour information based on the metrics introduced in Section 3.4. We used data collected for multiple persons in both stationary and moving robot scenarios (see Table 3.3 for more details). In this chapter all results are presented using a combined data set that includes both scenarios. Each experiment including all the data was repeated 10 times with different random variations in the particle filter for each trial using $N_D = N = 1000$ particles.

To determine optimal values of the system parameters we used a similar approach as described in the previous chapters, choosing an area accuracy metric as the performance criterion. The influence of each parameter on the performance of the tracker was checked independently and repeated 10 times with different random variations in the particle filter for each trial run with $N_D = N = 1000$ particles. The optimised values obtained for the system parameters were as follows:

Parameter	**Value**
n_k	10 frames
λ	50

The other parameters for the measurement model, motion model and penalty terms ψ_D and ρ were kept without change.

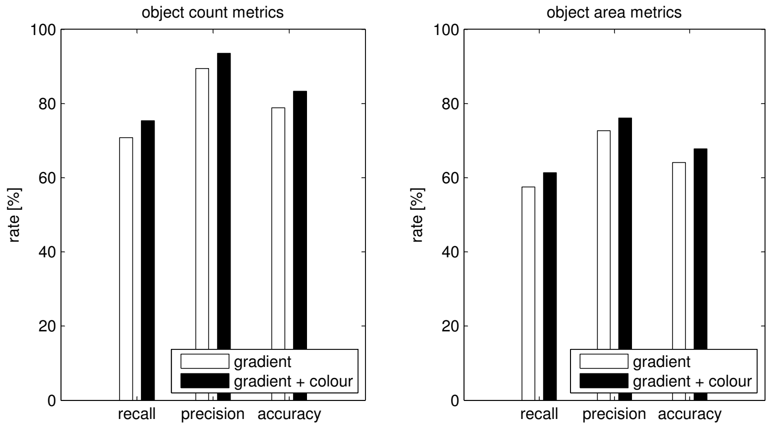

Figure 6.4: Detection and localisation metrics for tracking multiple persons without and with colour information.

Results presented in Figure 6.4 show the difference in performance of the tracker with and without additional colour information. Both detection and localisation metrics indicate a significant improvement due to a better focus of samples around tracked persons. This leads not only to more precise estimates but also decreases the number of cases when the tracker loses track of the person. The overall accuracy is affected by low recall values, which are caused mostly by occluded persons.

Figure 6.5 shows the comparison of different colour representations based on colour moments and histograms. We used a colour histogram with 20 evenly spaced bins for each colour channel. To measure similarity of two histograms we used a method proposed by [Stricker and Orengo, 1995]. This method, instead of comparing standard histograms directly, is based on cumulative histograms. Each bin $\tilde{h}_j$ of the cumulative colour histogram is defined in terms of normal histogram bins h_i such that

$$\tilde{h}_j = \sum_{j \leq i} h_i. \tag{6.18}$$

We use the L_2 metric specified by Equation 6.3 to measure similarity between two cumulative histograms. Thanks to this approach problems caused by sparse elements in histogram bins and effects of quantization are reduced, and the approach was shown to perform better than similarity measures based on standard histograms. The performance of the tracker based on histograms is slightly better than when using moments. However satisfactory results can be obtained even when using only the first moment of the colour distribution.

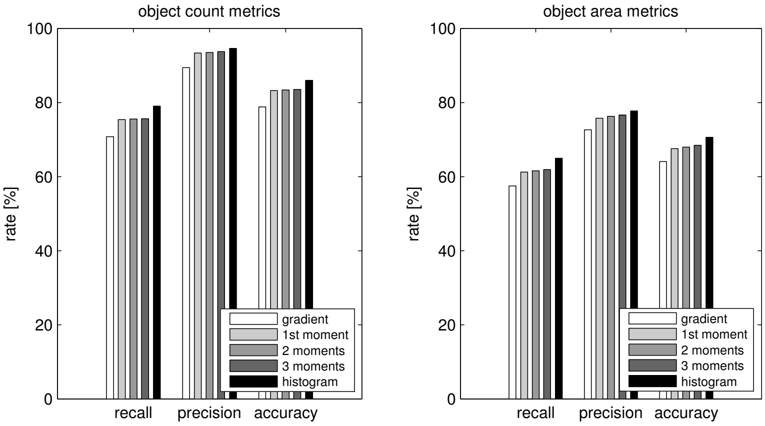

Figure 6.5: Comparison of different colour representations.

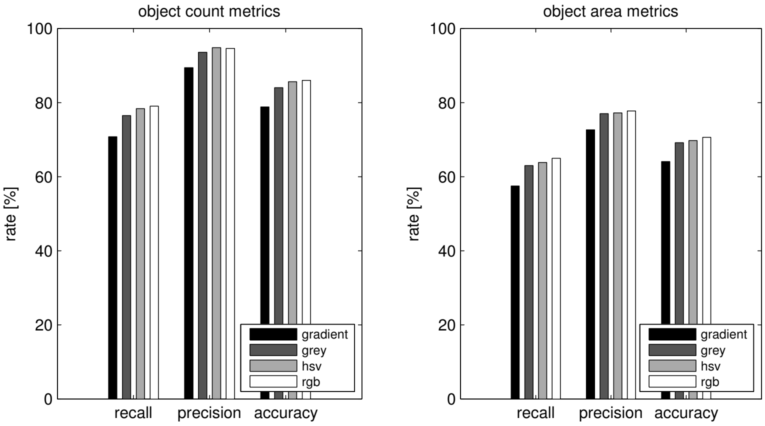

Figure 6.6: Comparison of different colour spaces using histograms with 20 bins.

Type of integral image	Platform	
	PeopleBoy Intel Pentium III 0.85 GHz [ms]	AMD Athlon XP 2.00 GHz [ms]
greyscale	0.86	0.24
colour (RGB)	5.12	2.09
colour (RGB) 2^{nd} degree	12.83	4.35
colour (RGB) 3^{rd} degree	16.09	4.90

Table 6.1: Time requirements for building different variants of integral image.

Representation	Platform	
	PeopleBoy Intel Pentium III 0.85 GHz [ms]	AMD Athlon XP 2.00 GHz [ms]
gradient	33.73	13.52
greyscale	35.96	15.88
colour, RGB, first moment	50.24	17.66
colour, RGB, first two moments	64.19	23.46
colour, RGB, first three moments	68.79	25.89
colour, RGB, histogram (20 bins)	8011.00	2709.00

Table 6.2: Average processing time needed to calculate 1000 samples using different colour representations.

Figure 6.6 shows the performance of the tracker when using colour histograms with different colour spaces including grey scale, RGB and HSV. It can be seen that most of the improvement is due to the intensity information. Moreover using HSV colour representation results in slightly worse performance compared to the RGB space. This fact indicates that for certain tasks (in our case data association) the HSV representation does not necessarily have to lead to better results.

Table 6.1 presents the time required to process different versions of the integral image including one-channel grey-scale and RGB colour images for different power sums used later to calculate features of higher moments. Grey-scale images can be calculated very quickly: it takes approx. 0.86 ms on the PeopleBoy robot. Colour images require more time since it is necessary to calculate three layers. Moreover the integral images needed for calculation of higher moments (2nd and 3rd) require additional multiplications and a 64-bit representation of data, which further increases time demands. Table 6.2 presents the average processing time needed for calculation of 1000 samples when using different colour representations. It takes about two times longer to calculate one step of the tracking procedure when using all three moments compared to the tracker based on gradient information only. A good trade-off between time requirements and performance of the tracker for our set-up is a representation using only the first moment of the colour distribution. However using just intensity information (grey-scale images) is also an attractive alternative, especially when other tasks (e.g., navigation) have to be performed by the robot at the same time. With increasing computational power of robots, use of higher moments should be possible in real-time scenario. For comparison the figure also includes time requirements for the approach based on colour histograms. The average time to calculate 1000 samples is around 2 orders of magnitude higher compared to the representation based on 3 moments. The obtained result of ~2.7 sec. on a 2.00 GHz processor would make histogram based methods unsuitable for real-time applications like ours.

6.4 Conclusions

In this chapter we presented an extension of the tracker to incorporate colour information. The additional colour information increases the robustness and accuracy of the tracker. An adaptive appearance colour model of each tracked person reduces problems related to different light conditions and changes in view. An efficient and compact representation based on statistical moments of the colour distribution was shown to perform similarly to other popular representations based on colour histograms, while requiring much less computational time. A rapid method for calculating rectangular features enables real-time tracking using both thermal and colour information.

The proposed adaptive colour model has several useful properties; for example, it is recognisable from a wide range of distances, and is fairly invariant to different orientations of the persons. However, in situations when people wear similar clothes (uniforms, lab coats) obviously there will be no gain in the performance of the tracker. The tracker still cannot cope with occlusions, but the availability of the colour information allows us to further extend the system to detect and handle occlusions, as presented in the next chapter.

Chapter 7

Handling Occlusions

In this chapter we present a novel approach for handling occlusions in the people tracking system. We treat occlusions explicitly, i.e. we first detect them and then reason about them in the tracking algorithm on the basis of heuristics. Detection of occlusions allows us to determine the occluding persons and treat them differently to the persons that are occluded. We present a classifier built using the AdaBoost algorithm of Freund and Shapire [Freund and Schapire, 1995] to compare visual information for a pair of tracked persons in order to determine which person occludes the other. In the experiments we show the performance of this classifier and determine the visual features which contain the most relevant information for occlusion detection. The results from the trained occlusion detector are then applied to the problem of occlusion handling in a heuristic extension to the tracking algorithm described in the previous chapter. The corresponding experiments demonstrate a further improvement in performance of the multi-person tracker.

7.1 Detecting Occlusions

To detect occlusions we first have to check if regions corresponding to different persons overlap. This requires $M \cdot log(M)$ comparisons, where M is a number of persons in the scene. In crowded environments it may happen that more than two people overlap with each another. Ideally we would have to consider all interacting persons and their relations. To avoid combinatorial explosion we propose a simplified approach that sorts the order of all persons in the image according to pairwise comparisons. The proposed occlusion detector specifies which one of two overlapping persons is in front of the other. Which person occludes the other is determined on the basis of a sort procedure which requires $M_O \cdot log(M_O)$ comparisons, where M_O specifies the number of overlap-

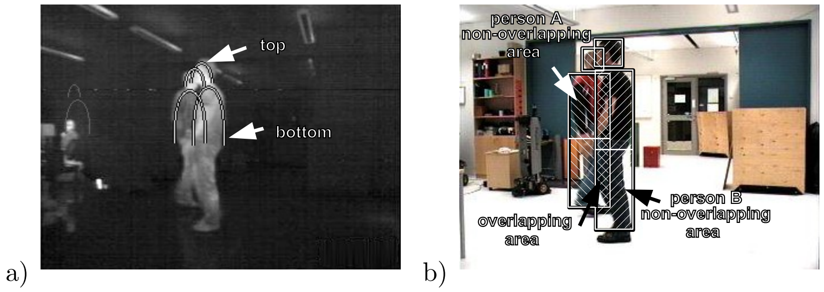

Figure 7.1: a) Top and bottom thermal features. b) Overlapping and non-overlapping areas from which colour features are extracted.

ping persons.

There are several features that could indicate the correct order of two overlapping persons in the image, from which we have chosen a set of three thermal and three colour features:

- The "strength" (i.e., mean gradient value) of a tracking filter, since a person for which the corresponding tracker indicates a higher confidence is more likely to be in the front. This feature is, however, very noisy and is affected by many factors such as movement of the camera, temperature of the environment, etc.

- The top and bottom of the elliptic contour model (see Fig. 7.1a) can also indicate the depth of a person, since closer persons appear taller and closer to the upper and bottom border of the image. However the bottom is affected in situations when persons stand too close to the camera such that their lower part is cut and cannot be properly estimated. The top of a person's head is a more reliable feature which is, however, affected by the different height of persons.

- Another set of features is obtained from the colour similarity of the image region corresponding to a person. We have chosen three such regions including the overlapping, non-overlapping and whole areas of a person (see Fig. 7.1b). Occluded persons should have lower similarity values. These features can be misleading when overlaps are small (overlapping area) or big (non-overlapping area).

Thus six features were obtained: three corresponding to the information from the thermal image and three colour features. Since a single feature cannot easily determine the right order of the persons we propose an application of a boosting algorithm [Freund and Schapire, 1995] to weight

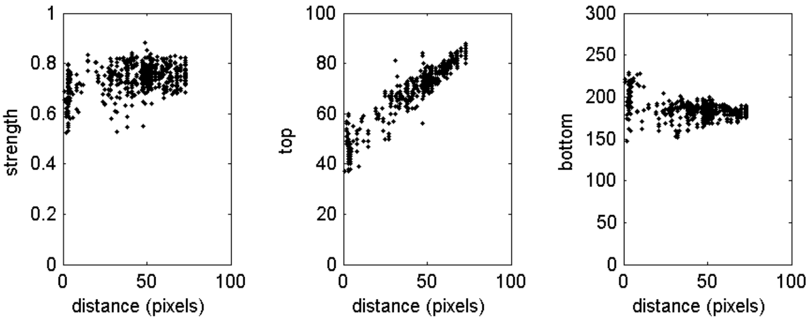

Figure 7.2: Relationship of the different thermal features to the apparent distance of a person taken from the ground truth data.

and combine a number of "weak classifiers" built from these features, resulting in a "strong classifier" with much improved occlusion detection accuracy.

To give an impression of the discriminative power of the thermal features used, we present a graphical representation of their relationship to the apparent distance of a person taken from the ground truth data (see Fig. 7.2). This distance uniquely determines the order of the persons. Note that range information from a laser scanner could also be used to simplify this problem. However in this work we consider an exclusively vision-based system. (It would not be meaningful to provide a similar visualisation for the colour features, since these features are based on comparisons of two tracked persons rather than a single tracked person as in the thermal case.)

7.2 AdaBoost Approach

We use the AdaBoost (Adaptive Boosting) classification algorithm [Freund and Schapire, 1995] for selecting the optimal combination of selected features to detect occlusions. AdaBoost is a linear classifier that has some attractive properties such as good generalisation, simplicity of implementation and can be also considered as a feature selector. AdaBoost combines results from so-called "weak" classifiers $h_t(x)$ into one "strong" classifier $H(x) = sign(f(x))$ given that

$$f(x) = \sum_{t=1}^{T} \alpha_t h_t(x), \tag{7.1}$$

where T is the number of weak classifiers and α_t is an importance weight given to each "weak" classifier $h_t(x)$ according to the performance during an iterative learning process (see Algorithm 5 for details). As a result we obtain a final classifier that performs better than any of these weak classifiers alone. The high performance of the final strong classifier is due to the fact that during the learning process focus is put on the examples from the training set which are most difficult to classify (this process is called "boosting").

Algorithm 5 AdaBoost learning algorithm after [Viola and Jones, 2001]

```
input:
  - training data $(x_i, y_i),\ i = 1, \ldots, N$ with $N_p$ positive $(y_i = 1)$ and $N_n$ negative $(y_i = 0)$ examples
init:
  - set uniform distribution of weights assigned to each input examples $w_1^i = \frac{1}{N_p}, \frac{1}{N_n}$ depending on the value of $y_i$
for $t = 1$ to $T$ do
  - for each feature $j$ train a classifier $h_j$ with error $\epsilon_j = \sum_{i=1}^{N} w_t^i |h_j(x_i - y_i)|$
  - select the best weak classifier $h_j$ with the lowest error $\epsilon_j$ and set $(h_t, \epsilon_t) = (h_j, \epsilon_j)$
  - update weights: $\tilde{w}_{t+1}^i = w_t^i \beta_t^{1-e_i}$ with
    $e_i = \begin{cases} 0 : & x_i \text{ correctly classified} \\ 1 : & \text{otherwise} \end{cases}$ and $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$
  - normalise all weights: $w_{t+1}^i = \frac{\tilde{w}_{t+1}^i}{\sum_{j=1}^{N} \tilde{w}_{t+1}^j},\ i = 1, \ldots, N$
end for
output:
  - the final strong classifier:
    $H(x) = \begin{cases} 1 : & \sum_{t=1}^{T} \alpha_t h_t(x) \geq 0.5 \sum_{t=1}^{T} \alpha_t \\ 0 : & \text{otherwise} \end{cases}$ ,
    where $\alpha_t = log(\frac{1}{\beta_t})$
```

Following the approach presented by [Viola and Jones, 2001] we first developed an approach using simple weak classifiers that are based on a single-valued feature $f_j(x)$

$$h_j(x) = \begin{cases} 1 : & p_j f_j(x) < p_j \theta_j \\ 0 : & \text{otherwise,} \end{cases} \tag{7.2}$$

where θ_j is a threshold and $p_j = \{-1, 1\}$ is a parity indicator determining the direction of the inequality sign. During the training procedure optimal values of θ_j and p_j are determined such that the number of misclassified training examples is minimised as

$$(\theta_j, p_j) = \underset{(\theta_i, p_i)}{\operatorname{argmin}} \sum_{n=1}^{N} |h_i(x_n) - y_n|. \tag{7.3}$$

The number of possible combinations of (θ_j, p_j) is limited since there is a finite set of training examples.

We found that results using the single features directly can be limited when the number of weak classifiers is small. In our case there are only six individual features: the strength of the tracker, the estimate of the upper and lower end of a person, and the colour similarity of the three areas. Boosting with simple weak classifiers in such a case is limited, resulting in relatively low performance of the final strong classifier. One way to increase the number of weak classifiers is to use a weighted combination of T features such as

$$f_j(x) = \sum_{i=1}^{T} \alpha_i f_i(x), \tag{7.4}$$

where α_i specifies a weight for a single valued feature $f_i(x)$. We discretise possible weight values α_i from the range $\{-1, 1\}$ into N_f fractions. As a result we obtain a much larger number of weak classifiers that can be selected by the boosting algorithm. In our experiments, we compared results using weak classifiers built from two and three features ("weighted pairs" and "weighted triplets" respectively). Since the underlying weak classifiers in this case are equivalent to linear networks, and the Adaboost algorithm itself builds a perceptron-type classifier from the weak classifiers, this approach is equivalent to building a multi-layer perceptron rather than a single-layer perceptron as in the approach of Viola and Jones[Viola and Jones, 2001]. The resulting improvement in occlusion detection accuracy can be seen in Section 7.4.

7.3 Dealing with Occlusions

The learned occlusion detector can be used to improve the tracking performance during occlusions. It is used in two different ways: first, to alter the penalising policy between the trackers (as described in Section 5.2), and second, to re-identify occluded persons when they reappear.

Our interaction model for tracking multiple persons allows tracking of people that overlap to a certain degree. This is achieved by modifying the interaction factor ρ to prevent target fetching (i.e., to prevent two filters in close proximity from collapsing around the same tracked object). The proposed pairwise occlusion detector is used to determine which of the tracking filters is occluded. We consider two possible situations:

partial occlusion and total occlusion. During partial occlusion, some part of a person is still visible. However, the gradient along the contour is disturbed, which can cause a quick disappearance of the tracker. To avoid this we change the penalty term for a partially occluded tracker which we denote by ρ_o so the penalty equation 7.5 changes to

$$\psi = e^{(-\rho_o g_{ij})}. \tag{7.5}$$

Interaction with other filters (non-overlapping with this pair) remains unchanged. A modified update procedure for the tracker with improved occlusion handling is presented in Algorithm 6.

Algorithm 6 A modified update procedure for tracking filters (not totally occluded).

```
for each filter
    measure(thermal,colour)

handle occlusions():
  -  determine overlaps between filters
  -  determine occlusions between overlapping filters:
       -  detect occlusions using the AdaBoost detector
       -  assign occluded/occluding filters
       -  assign partial/total occlusions
  -  adjust the penalty term $\rho$ (Eq. 7.5) for each filter according to
     the type of occlusion

for each filter
    penalise()
    calculate estimates()
```

When the head contour of a person becomes occluded the corresponding tracker is considered to be totally occluded. This means that we can only guess the true position of this person. We assume that the state of the occluded person is the same as the state of the occluding person (simply stating that an occluded person is behind the occluding one). No penalty is considered for the occluded tracker. We keep particles of the totally occluded tracker for a short time (we use a threshold of 8 frames here) in situations when quick occlusions occur and the velocity of particles may allow to resolve this occlusion. However after this time has elapsed the particles of the tracker are removed and the only information kept is the colour model. When a new person is detected by the detector this information is used to match the colour model to all occluded trackers. If the colour model is most similar to the closest occluded tracker then the detected person is considered to be an occluded one. Otherwise the person is considered to be a new person. To avoid situations where the occluded tracker stays forever behind the occluding

Feature type	**Results** [%]	**T total**
thermal	76.39 ± 4.49	243
colour	69.07 ± 1.94	243
both	89.38 ± 2.48	1206

Table 7.1: Classification results for different type of features used to create weak classifiers (resulting in *T total* weak classifiers).

one, we also specify a maximal duration of occlusion (in our case 10 sec.). This minimises errors in the case where an occluded person disappears from the scene in some other way (e.g., through a door or a corridor behind an occluding person) or in cases of missed assignments to newly detected persons.

7.4 Experiments - Detecting Occlusions

We extracted the described thermal and colour features from data collected for multiple persons in both stationary and moving robot scenarios (see Table 3.3 for more details). We considered only cases when two or more people were overlapping. Moreover since the behaviour of the tracker without proper occlusion handling is unpredictable after a total occlusion occurs, we took only those examples that preceded the moment of total occlusion. During partial and total occlusions, the colour models of the respective persons were not updated. In this way we obtained $N_p = 121$ positive and $N_n = 121$ negative examples giving in total $N = 242$ examples.

We created additional weak classifiers based on weighted sums of pairs of features with $N_f = 20$ fractions giving, in the case of all six thermal and colour features used, 1200 new weak classifiers. We used 60% of randomly selected input examples as a training set and the remaining part as a test set. Each training procedure was repeated 10 times.

The strong classifier learned from the combination of thermal and colour features was able to predict correctly around 89% of all cases (see Table 7.1). This gives a significant advantage over classification results obtained when thermal and colour features were used separately ($p < 0.01$). Thermal features provided significantly better results than colour features alone.

Table 7.2 shows results for different methods of combining features into weak classifiers. The comparatively bad results when using single features are caused by the low number of weak classifiers. The proposed method of using a weighted combination of pairs of features increased the

Combination of features	**Results** [%]	**T total**
single	74.94 ± 4.88	6
weighted pairs	89.36 ± 2.48	1206
weighted triplets	89.38 ± 1.82	129206

Table 7.2: Classification results for different combination of features to create weak classifiers (resulting in *T total* weak classifiers).

Single feature	**Results** [%]
strength	50.10 ± 4.91
top	72.99 ± 3.85
bottom	56.49 ± 4.25
colour	67.62 ± 3.04
colour_o	45.56 ± 2.69
colour_no	67.42 ± 2.92

Table 7.3: Classification results for single features (colour_o and colour_no labels stand for colour of the overlapping and non-overlapping area respectively).

performance of the final classifier by around 15%. We also made tests with weighted triplets of features for comparison. Despite the much higher number of possible weak classifiers the difference in performance compared to weighted pairs was not found to be significant (based on a paired t-test with confidence level $p = 0.01$).

From the results presented in Table 7.3 we can get an impression about how much information is provided by a single feature. The most reliable features are the top of a person's head, colour similarity of the whole region and of the non-overlapping area. Weak classifiers based on combinations of these features had the highest importance (see Table 7.4). Other features also contributed to the final classifier (e.g., the feature based on the position of the bottom of the elliptic model) even though their individual performance was relatively poor.

7.5 Experiments - Dealing with Occlusions

This section presents results showing the performance of the tracker when tracking multiple persons with improved occlusion handling using the metrics introduced in Section 3.4. We used data collected for multiple persons in both stationary and moving robot scenarios (see Table 3.3 for more details). In this chapter all results are presented using a combined

Place	Weight of a feature					
	strength	top	bottom	colour	colour_o	colour_no
1	-0.05	-	-	-	-	1.00
2	-	-0.05	-	1.00	-	-
3	-	-1.00	0.45	-	-	-
4	-	-0.75	1.00	-	-	-
5	-	-	0.05	-	-	1.00
6	-	-0.80	1.00	-	-	-
7	-	-	0.10	-	-	1.00
8	-0.55	1.00	-	-	-	-
9	-1.00	0.05	-	-	-	-
10	-	-	-0.05	1.00	-	-

Table 7.4: 10 best weak classifiers with their respective weights (colour_o and colour_no labels stand for colour of the overlapping and non-overlapping area respectively).

data set that includes both scenarios. Each experiment including all the data was repeated 10 times with different random variations in the particle filter for each trial using $N_D = N = 1000$ particles.

To determine optimal values of the system parameters we used a similar approach as described in the previous chapters, choosing an area accuracy metric as the performance criterion. The influence of each parameter on the performance of the tracker was checked independently and repeated 10 times with different random variations in the particle filter for each trial run with $N_D = N = 1000$ particles. The optimised values obtained for the system parameters were as follows:

Parameter	Value
ρ_o	0.5

Other parameters for the measurement model, motion model and penalty terms ψ_D and ρ were kept without change.

Results presented in Figure 7.3 show the difference in performance of the tracker using only thermal gradient information, with additional colour information, and with both colour information and the occlusion handling procedure. Both detection and localisation metrics indicate a significant improvement when using the occlusion detector and enhanced tracking algorithm, which diminishes problems related to occlusions giving an increase of 6.83% in the area recall metrics and 3.12% in the area accuracy metrics. The output from the tracker can be seen in Figure 7.4.

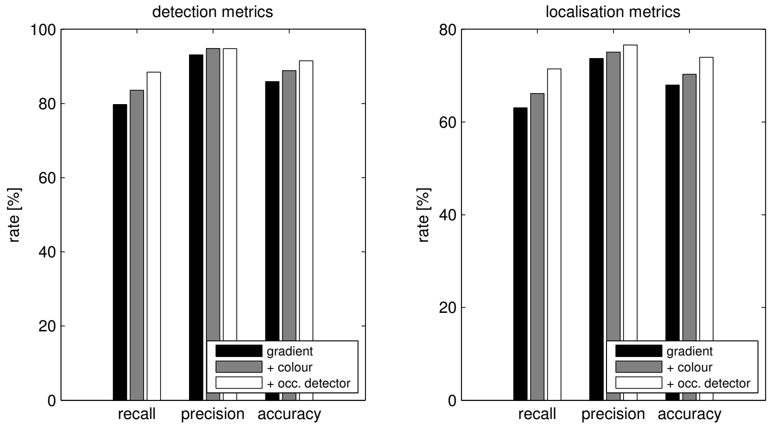

Figure 7.3: Detection and localisation metrics for tracking multiple persons without and with colour information and with occlusion handling procedure.

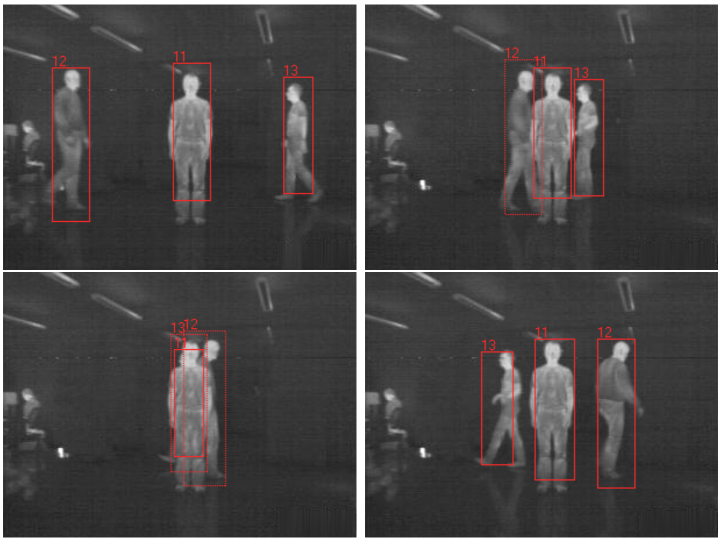

Figure 7.4: Selected thermal images from the sequence showing the output from the tracker before, during and after the occlusion of three simultaneously tracked persons. The bounding boxes corresponding to occluded persons are marked by a dotted line.

7.6 Conclusions

In this chapter we presented an algorithm for detecting occlusions in the people tracking system. The algorithm is able to determine the order of persons in the image using a combination of thermal and colour cues. The order is determined by a classifier built using the AdaBoost algorithm which performs a comparison between two selected persons. This procedure determines which person is occluding the other one. We have shown that using weak classifiers constructed from a weighted combination of pairs of features gives the best performance while keeping the computational demands low. During the learning procedure AdaBoost automatically selects the best combination of features, creating a high performance "strong" classifier. Using this classifier, an extension to the multi-person tracker was proposed for handling occlusions, further improving the performance of the system.

Of course such a method of dealing with occlusions can be considered only as a proposal in order to demonstrate the concept of explicit occlusion handling based on a learned classifier. It was based on observation of the occlusion problem in our specific scenario and application. We believe that the question of how to handle occlusions is impossible to answer in a general way, i.e. independently of a particular application. Our solution demonstrates that it is plausible to deal with occlusions to some extent, and through experiments we showed that this increases the overall performance of the tracker. Such a solution has obvious pitfalls that should be considered in future work, such as proper handling of misclassification errors, wrong assignments after occlusions, dealing with uniformly dressed people, etc. A mobile robot itself could be designed to check if the occluded person is really behind another person by taking appropriate actions (e.g. driving to look closer of from a different angle). Recognition of human behaviour could also help to solve this kind of problem.

Chapter 8

Conclusions and Future Work

This chapter presents a summary of the main contributions of the thesis and an analysis of their significance. Open questions are then discussed together with possible improvements to the presented system.

8.1 Summary of Contributions

In this thesis a people tracking system designed for mobile robots was presented. The system is entirely vision-based, using sensory information provided by thermal and colour cameras. The use of a thermal camera helped to overcome the main difficulty for vision-based systems, especially for those mounted on a moving platform, namely reliable and efficient segmentation of persons. The elliptic contour model based on thermal gradient information ensures both high robustness and computational efficiency of the method. The results indicate that the basic tracker using the proposed model minimises false alarms especially well.

Our tracking system follows recent trends in the design of tracking algorithms. Many previous vision systems required an exhaustive scanning procedure for the detection task which limited their use in real-time applications. In contrast our system uses an efficient sample-based tracking algorithm that avoids excessive scanning of the whole image. In addition the system is built in the spirit of the unified tracking framework which provided a combined solution to the detection and tracking problems. Such a solution is especially suitable for mobile systems where the movement of the platform causes increased measurement noise. Our system incorporates unthresholded measurements directly into the tracking framework, avoiding the loss of information that can occur in a separate pre-processing stage for people detection.

To track multiple persons we proposed an efficient heuristic tracking algorithm for tracking a varying number of persons that mitigates the problems with combinatorial explosion associated with other multi-target tracking methods. Our method operates directly on the sample weights and uses an efficient method of calculating interactions between individual filters. For each new appearing person an individual tracking filter is assigned which is initialised by a sequential detector that detects consecutive persons in predictable time. The experiments showed that it is possible to adjust the number of samples for each individual filter depending on the number of tracked persons to further improve the computational efficiency of tracking without affecting tracking quality.

Through the experiments we showed that the data association in our system can be improved when the additional colour information is incorporated. The fusion of thermal and colour information is done within the tracking framework by combination of the respective likelihoods. Our system uses an efficient colour representation based on the integral image representation to speed up processing. It was found that use of three colour moments (in RGB colour space) showed the best performance while using just one moment gave the best compromise between performance and computational requirements. In addition the results showed that use of the HSV colour space, which is popular in many computer-vision applications, does not necessary lead to better results.

A major difficulty for all tracking systems involving multi-target tracking is the problem of occlusions. The occlusions in our system are treated explicitly. To achieve this we propose a new approach for detecting occlusions using a machine learning classifier for pairwise comparison of persons (classifying which one is in front of the other). While we found that the final classifier learned by the AdaBoost algorithm based on thermal features gave better results than a classifier using colour features, the combination of thermal and colour features produced the best classification performance. We also incorporated the occlusion detection classifier into the tracking algorithm, adjusting the penalty policy in occlusion handling, which resulted in a significant increase in the performance of the whole system.

Finally the thesis provided a comprehensive, quantitative evaluation of the whole system and its different components using a set of well defined performance measures. The behaviour of the system was investigated on different data sets including different real office environments and different appearances and behaviours of persons. Moreover the influence of all important system parameters on the performance of the system was checked and their values optimised based on the area accuracy metric, which proved to be a good indicator of the overall performance of the system. A list and description of all system parameters is presented in Appendix A. The proposed methodology can be easily extended and used in future research for comparisons and further improvements to the system.

8.2 Limitations and Possible Improvements

To the best knowledge of the author, this thesis presents the first system using a thermal camera for people detection and tracking that was designed for mobile robots. The experience gained during this work allows us to expect that the popularity of thermal sensors in mobile robotic applications should increase in the near future. This is related to decreasing prices and the high potential of possible applications.

A thermal sensor not only simplifies the problem of detection but also allows for work in darkness which could be especially important in security applications. It could also be used for other tasks since the thermal signature of objects such as lamps or radiators can be used in other localisation and object recognition tasks.

The presented tracking framework allows easy incorporation of other cues and modalities and also to use different extensions to the proposed measurement and motion model.

The efficiency of the proposed method allowed for the implementation of the person following behaviour on the robot that was also used for data collection in this work. The high frame rate of the system allowed for smooth operation in our office environments.

When designing the elliptic contour model the main emphasis in our work was placed on its efficiency. This simple model could be extended in various ways. The presented elliptic model allows to detect and track persons in the up-right position only. A more sophisticated model would be needed to deal with other situations when persons are sitting, bending or lying down, for example. This could be realised by a completely new design of the contour model allowing for better flexibility of the shape or by creating several separate contour models for each possible behaviour. The latter approach could also be used for human behaviour recognition. In some situations the generality of the presented elliptic model causes inaccurate estimates resulting in low localisation performance. This problem could be diminished by incorporating an additional feature based model. An approach that combines our gradient model and a learned feature model based on the integral image representation was presented in [Treptow et al., 2006].

The proposed algorithm for tracking of multiple persons was also designed to run on real-time systems. It would be interesting to investigate the difference in performance with other tracking algorithms to compare their strengths and weaknesses. This would include other solutions based on individual trackers and techniques such as MCMC-based multiple target tracking described in [Khan et al., 2004]. Also more sophisticated forms of the interaction model could be investigated, for example, models that take into account the velocity of tracked persons. Further improvements to the interaction model would require solving problems of behaviour and intention recognition [Cielniak et al., 2003]. This information could be used to determine possible interactions between people and to select an appropriate interaction model accordingly. In addition our occlusion detector could also be incorporated.

During the experiments we used data containing a maximum of 4 persons. However this number does not limit the maximum capabilities of the tracker. The maximum number of persons depends heavily on the image resolution and kind of interaction between persons. With the resolution of 320 × 240 our tracker would be able to track up to 10 persons. In more crowded scenes the tracker most probably would loose some of them. To deal with this problem the system could be extended to detect and track crowds of people, perhaps with a single tracker. A hypothetical application would be a crowd of school students being given a tour of the robot lab.

The presented appearance colour model was shown to be very effective, despite its limiting assumptions. It could be used in other applications requiring very fast processing times. However the simple rectangular features that can be rapidly calculated can be quite noisy because of the assumed approximations of the human body shape. To decrease this noise other methods for collecting features from more complicated shapes should be investigated.

The colour appearance model could be also used to re-identify persons when they re-appear on the scene, as in the approach proposed by [Zajdel et al., 2005]. It could also be used to identify tracked persons. This colour appearance model has several useful properties; for example, it is recognisable from a wide range of distances, and is fairly invariant to different orientations of persons. However, it has a number of obvious drawbacks, e.g., people often change their clothes on a regular basis! To overcome this problem, combination with other people recognition techniques with complementary strengths and weaknesses should be investigated. Accurate (but not so robust) techniques such as face or speech recognition could be used at the start of each day to obtain a confident initial estimate of the identity of a person. Then the clothing model could be re-acquired or added to an existing database of clothes for that person, and used for general identification purposes later in the day, and in situations where faces and voices cannot be easily recognised. Combination with other recognition techniques would also help to overcome problems when the appearance of persons is very similar, e.g., when they wear uniforms.

The proposed method for dealing with occlusions would also need further investigation. Our approach is based on the observation of the occlusion problem in a specific scenario and application. Obvious pitfalls such as improper handling of misclassification errors, wrong assignments after occlusions, uniformly dressed people, etc. should be considered in future work.

The proposed system is not limited only to specific hardware, environments and scenarios. Further extensions would open many possible topics for investigation and future research. For example an interesting extension would include mounting an omnidirectional mirror on top of a colour camera. The system could learn appearance colour models of persons when they are visible in the thermal camera. This information could be used further for tracking persons in the colour omni-camera, providing information about persons in the whole surrounding of the robot. The use of thermal camera is also not mandatory: in preliminary investigations we also applied the elliptic measurement model to difference images obtained from a colour camera using background subtraction techniques. Our robot was especially designed for tasks involving cooperation with humans. However it should be straightforward to use the system on other mobile robots. Possible problems could include the change in perspective, which would require re-tuning of some system parameters. Our system was designed for indoor environments. In outdoor environments there are several complications that would have to be considered: constantly changing temperature of the environment, heavily dressed people, more difficulties with colour information, etc.

One day, however, robots may take over the planet, destroying all the people and making the proposed method redundant – then we will know we really succeeded!

Appendix A

System Parameters

N	number of particles
$\alpha_1, \ldots, \alpha_7$	weights of the segments of the elliptic contour model
κ, θ	gradient likelihood parameters
U_t	estimate uncertainty threshold
B	amount of randomness in the motion model
r_{wh}, r_{dw}, y_{min}, y_{max}, h_{min}	geometrical constrains for the ellipse model
N_D	number of detector particles
ψ_D	penalty factor for the detector
ρ	interaction factor for multiple tracking filters
ρ_o	interaction factor for an occluded tracking filter
λ	importance factor of the colour likelihood
n_k	number of recent frames used to construct a colour model
$\boldsymbol{p}_{aff}$	affine transformation coefficients

Bibliography

D. L. Alspach and H. W. Sorenson. Nonlinear bayesian estimation using gaussian sum approximation. *IEEE Transactions on Automatic Control*, 17:439–447, 1972.

Y. Bar-Shalom and T. Fortmann. *Tracking and Data Association.* Academic Press, 1988.

Y. Bar-Shalom, X. R. Li, and T. Kirubarajan. *Estimation with Applications To Tracking and Navigation.* John Wiley and Sons, Inc., 2001.

J. Barreto, P. Menezes, and J. Dias. Human-robot interaction based on Haar-like features and eigenfaces. In *Proc. of the IEEE International Conference on Robotics and Automation*, New Orleans, USA, 2004.

A. M. Baumberg and D. C. Hogg. An efficient method for contour tracking using active shape models. Technical Report 94.11, University of Leeds, April 1994.

P. N. Belhumeur, J. Hespanha, and D. J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. In *Proc. of the European Conference on Computer Vision*, pages 45–58, 1996.

M. Bertozzi, A. Broggi, P. Grisleri, T. Graf, and M. Meinecke. Pedestrian detection in infrared images. In *Proc. of the IEEE Intelligent Vehicles Symposium*, pages 662–667, Columbus, USA, 2003.

C. Berzuini and W. Gilks. Resample-move filtering with cross-model jumps. In *Sequential Monte Carlo Methods in Practice.* Springer-Verlag, New York, 2001.

J. Black, T. J. Ellis, and P. Rosin. A novel method for video tracking performance evaluation. In *Proc. of the Joint IEEE Int. Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pages 125–132, Nice, France, October 2003.

M. R. Blackburn and H. G. Nguyen. Autonomous visual control of a mobile robot. In *Proc. of Image Understanding Workshop*, pages 1143–1150, Monterey, Canada, November 1994.

A. Blake and M. Isard. *Active Contours.* Springer, 1998.

A. Blake, B. Bascle, M. Isard, , and J. MacCormick. Statistical models of visual shape and motion. *Proc. of the Royal Society of London, A*, 356:1283–1302, 1998.

J. Blanco, W. Burgard, R. Sanz, and J.L. Fernandez. Fast face detection for mobile robots by integrating laser range data with vision. In *Proc. of the International Conference on Advanced Robotics*, 2003.

H. J. Boehme, U. D. Braumann, A. Brakensiek, A. Corradini, M. Krabbes, and H. M. Gross. User localisation for visually-based human-machine interaction. In *Proc. of the IEEE Int. Conf. on Face and Gesture Recognition*, pages 486–491, Nara, Japan, 1998.

H. J. Boehme, U. D. Braumann, A. Corradini, and H. M. Gross. Person localization and posture recognition for human-robot interaction. In *Proc. of the 3rd International Gesture Workshop*, pages 105–116, Gif-sur-Yvette, France, 1999.

L. Brèthes, P. Menezes, F. Lerasle, and J. Hayet. Face tracking and hand gesture recognition for human-robot interaction. In *Proc. of the IEEE International Conference on Robotics and Automation*, pages 1901–1906, New Orleans, LA, USA, 2004.

R. Brunelli and D. Falavigna. Person identification using multiple cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17 (10):955–966, 1995.

W. Burgard, A.B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 114(1-2):3–55, 1999.

Z. Byers, M. Dixon, W.D. Smart, and C.M. Grimm. Say cheese!: Experiences with a robot photographer. In *Proc. of the Conference on Innovative Applications of Artificial Intelligence*, Acapulco, Mexico, August 2003.

J. Carpenter, P. Clifford, and P. Fernhead. An improved particle filter for non-linear problems. Technical report, Department of Statistics, University of Oxford, 1997.

Z. Chen and H. J. Lee. Knowledge-guided visual perception of 3D human gait from a single image sequence. *IEEE Trans. On Systems, Man, and Cybernetics*, 2(22):336–342, 1992.

G. Cielniak and T. Duckett. Person identification by mobile robots in indoor environments. In *Proc. of the IEEE International Workshop on Robotic Sensing (ROSE)*, page 5pp., 2003.

G. Cielniak and T. Duckett. People recognition by mobile robots. *Journal of Intelligent and Fuzzy Systems*, 15(1):21–27, 2004.

G. Cielniak, M. Bennewitz, and W. Burgard. Where is . . . ? Learning and utilizing motion patterns of persons with mobile robots. In *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, Acapulco, Mexico, 2003.

D. Comaniciu, V. Ramesh, and P. Meer. Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25 (5):564–577, 2003.

J. H Connell. A colony architecture for an artifcial creature. Technical Report TR-1151, MIT AI, 1989.

I. J. Cox and S. L. Hingorani. An efficient implementation of Reid's multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(2):138–150, 1996.

I. Craw, D. Tock, and A. Bennett. Finding face features. In *Proc. of the European Conference on Computer Vision*, pages 92–96, 1992.

D. Cunado, M. S. Nixon, and J. N. Carter. Automatic extraction and description of human gait models for recognition purposes. *Computer Vision and Image Understanding*, 90(1):1–41, 2003.

P. Del Moral. Non-linear filtering: Interacting particle solution. *Markov Processes and Related Fields*, 2(4):555–581, 1996.

D. S. Doermann and D. Mihalcik. Tools and techniques for video performance evaluation. In *Proc. of the International Conference on Pattern Recognition*, volume 4, pages 4167–4170, Barcelona, Spain, 2000.

R. Douc, O. Cappe, and E. Moulines. Comparison of resampling schemes for particle filtering. In *Proc. of the 4th International Symposium on Image and Signal Processing and Analysis*, pages 64–69, 2005.

A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice.* Springer-Verlag, New York, 2001.

R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification.* Wiley, New York, 2nd edition, 2000.

A. Elgammal and L. S. Davis. Probabilistic framework for segmenting people under occlusion. In *Proc. of the International Conference on Computer Vision*, Vancouver, Canada, 2001.

C. E. Erdem, A. M. Tekalp, and B. Sankur. Metrics for performance evaluation of video object segmentation and tracking without ground truth. In *Proc. of the Int. Conf. on Image Processing*, Thessaloniki, Greece, October 2001.

J. M. Ferryman, editor. *Proc. of the 1st IEEE Workshop on Performance Evaluation of Tracking and Surveillance*, Grenoble, France, March 2000.

S. Feyrer and A. Zell. Robust real-time pursuit of persons with a mobile robot using multisensor fusion. In *International Conference on Intelligent Autonomous Systems*, pages 710–715, Venice, Italy, 2000.

S. Feyrer and A. Zell. Detection, tracking, and pursuit of humans with an autonomous mobile robot. In *Proc. of the IEEE International Conference on Intelligent Robots and Systems*, pages 864–869, Kyongju, Korea, 1999.

D. Fox, W. Burgard, S. Thrun, and A.B. Cremers. Position estimation for mobile robots in dynamic environments. In *Proc. of the National Conference on Artificial Intelligence*, 1998.

D. Fox, W. Burgard, and S. Thrun. Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, 11:391–427, 1999.

Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *Computational Learning Theory: Eurocolt*, pages 23–37. Springer-Verlag, 1995.

D. Gavrila and L. Davis. 3d model-based tracking of humans in action: A multi-view approach. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 73–80, San Francisco, CA, USA, 1996.

D. M. Gavrila. The visual analysis of human movement: A survey. *Computer Vision and Image Understanding*, 73(1):82–98, 1999.

D. M. Gavrila and L. S. Davis. Towards 3-D model-based tracking and recognition of human movement. In Martin Bichsel, editor, *Int. Workshop on Face and Gesture Recognition*, pages 272–277, June 1995.

G. Gordon. Face recognition based on depth and curvature features. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 108–110, Champaign, Illinois, USA, June 1992.

N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *Proc. Inst. Elect. Eng. F*, 140(2):107–113, April 1993.

B. Han, C. Yang, R. Duraiswami, and L. Davis. Bayesian filtering and integral image for visual tracking. In *Workshop on Image Analysis for Multimedia Interactive Services*, Montreux, Switzerland, 2005.

I. Haritaoglu, D. Harwood, and L. Davis. Who, when, where, what: A real time system for detecting and tracking people. In *Proc. of the Third Face and Gesture Recognition Conference*, pages 222–227, 1998.

E. Hayman and J. Eklundh. Figure-ground segmentation of image sequences from multiple cues. In *Proc. of the European Conference on Computer Vision*, 2002.

I. Horswill. Polly: A vision-based artificial agent. In *Proc. of the National Conference on Artificial Intelligence*, pages 824–829, Menlo Park, CA, USA, 1993a.

I. Horswill. *Specialization of Perceptual Processes.* PhD thesis, MIT, Cambridge, MA, USA, 1993b.

J. Huang, S. Kumar, M. Mitra, W. Zhu, and R. Zabih. Image indexing using color correlograms. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 762–768, 1997.

E. Huber and D. Kortenkamp. Using stereo vision to pursue moving agents with a mobile robot. In *Proc. of the IEEE International Conference on Robotics and Automation*, pages 2340–2346, 1995.

S. S. Intille, J. Davis, and A. Bobick. Real-time closed-world tracking. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 697–703, 1997.

M Irani and P. Anandan. Robust multi-sensor image alignment. In *Proc. of the International Conference on Computer Vision*, pages 959–966, Washington, DC, USA, 1998.

M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1): 5–28, 1998.

M. Isard and J. MacCormick. Bramble: A bayesian multiple-blob tracker. In *Proc. of the International Conference on Computer Vision*, volume 2, pages 34–41, Vancouver, British Columbia, Canada, 2001.

B. Jensen and R. Siegwart. Using EM to detect motion with mobile robots. In *Proc. of the IEEE International Conference on Intelligent Robots and Systems*, Las Vegas, Nevada, October 2003.

S. J. Julier and J. K. Uhlmann. A new extension of the kalman filter to nonlinear systems. In *Proc. of AeroSense: The 11th Int. Symp. on Aerospace/Defence Sensing, Simulation and Controls*, 1997.

R. E. Kahn, M. J. Swain, P. N. Prokopowicz, and R. J. Firby. Gesture recognition using the perseus architecture. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 734–741, 1996.

K. Kanazawa, D. Koller, and S. Russell. Stochastic simulation algorithms for dynamic probabilistic networks. In *Proc. of the Conference on Uncertainty in Artificial Intelligence*, pages 346–351, San Francisco, CA, USA, 1995.

R. Karlsson and F. Gustafsson. Monte carlo data association for multiple target tracking. In *IEEE Target tracking: Algorithms and applications*, The Netherlands, 2001.

S. Khan and M. Shah. Tracking people in presence of occlusion. In *Proc. of the Asian Conference on Computer Vision*, Taipei, Taiwan, January 2000.

Z. Khan, T. Balch, and F. Dellaert. An mcmc-based particle filter for tracking multiple interacting targets. In *Proc. of the European Conference on Computer Vision*, 2004.

M. Kleinehagenbrock, S. Lang, J. Fritsch, F. Lömker, G. A. Fink, and G. Sagerer. Person tracking with a mobile robot based on multimodal anchoring. In *Proc. of the IEEE Int. Workshop on Robot and Human Interactive Communication (ROMAN)*, pages 423–429, Berlin, Germany, September 2002. IEEE.

B. Kluge, C. Köhler, and E. Prassler. Fast and robust tracking of multiple moving objects with a laser range finder. In *Proc. of the IEEE International Conference on Robotics and Automation*, Seoul, Korea, May 2001.

D. Kortenkamp, E. Huber, and R. P. Bonasso. Recognizing and interpreting gestures on a mobile robot. In *Proc. of the National Conference on Artificial Intelligence*, pages 915–921, 1996.

C. M. Kreucher. *An information-based approach to sensor resource allocation.* PhD thesis, The University of Michigan, 2005.

M. Lades, J. C. Vorbrüggen, J. Buhmann, J. Lange, C. v.d. Malsburg, R. P. Würtz, and W. Konen. Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on Computers*, 42:300–311, 1993.

S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back. Face recognition: A convolutional neural network approach. *IEEE Transactions on Neural Networks*, 8(1):98–113, 1997.

P. Li and H. Wang. Probabilistic object tracking based on machine learning and importance sampling. In *Proc. of the Iberian Conference on Pattern Recognition and Image Analysis*, volume 1, pages 161–167, 2005.

A. Lipton, H. Fujiyoshi, and R. Patil. Moving target classification and tracking from real-time video. In *Proc. of the IEEE Image Understanding Workshop*, pages 129–136, 1998.

J. J. Little and J. E. Boyd. Recognizing people by their gait: The shape of motion. *Videre*, 1(2):2–32, 1998.

C. Liu and H. Wechsler. Evolutionary pursuit and its application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(6):570–582, 2000.

X. Liu, Y. Zhuang, and Y. Pan. Video based human animation technique. In *Proc. of the 7th ACM International Multimedia Conference*, pages 353–362, Orlando, FL, USA, 1999.

J. MacCormick and A. Blake. A probabilistic exclusion principle for tracking multiple objects. *International Journal of Computer Vision*, 39(1):57–71, 2000.

S. Maskell, M. Rollason, N. Gordon, and D. Salmond. Efficient particle filtering for multiple target tracking with application to tracking in structured images. *Image and Vision Computing*, 21(10):931–939, 2003.

S. Maskell, M. Briers, and R. Wright. Fast mutual exclusion. In *Proc. of SPIE Conference on Signal Processing of Small Targets*, 2004.

S. Mckenna, S. Jabri, Z. Duric, and A. Rosenfeld. Tracking groups of people. *Computer Vision and Image Understanding*, 1(80):42–56, 2000.

U. Meis, W. Ritter, and H. Neumann. Detection and classification of obstacles in night vision traffic scenes based on infrared image. In *Proc. of the IEEE Intelligent Transportation Systems*, pages 1140–1144, Shanghai, China, 2003.

A. Mittal and L. S. Davis. M2tracker: A multi-view approach to segmenting and tracking people in a cluttered scene using region-based stereo. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 18–36, 2002.

A. Mohan, C. Papageorgiou, and T. Poggio. Example-based object detection in images by components. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(4):349–361, 2001.

M. Montemerlo, W. Whittaker, and S. Thrun. Conditional particle filters for simultaneous mobile robot localization and people-tracking. In *Proc. of the IEEE International Conference on Robotics and Automation*, Washington, DC, USA, 2002.

H. Nanda and L. Davis. Probabilistic template based pedestrian detection in infrared videos. In *IEEE Intelligent Vehicle Symposium*, Versailles, France, 2002.

C. J. Needham and R. D. Boyle. Performance evaluation metrics and statistics for positional tracker evaluation. In *Proc. of the International Conference on Computer Vision Systems*, pages 278–289, Graz, Austria, April 2003.

S. A. Niyogi and E. H. Adelson. Analyzing gait with spatiotemporal surfaces. In *IEEE Workshop on Nonrigid and Articulated Motion*, pages 64–69, Austin, Texas, USA, November 1994.

K. Okuma, A. Taleghani, N. De Freitas, J. J. Little, and D. G. Lowe. A boosted particle filter: Multitarget detection and tracking. In *Proc. of the European Conference on Computer Vision*, volume 1, pages 28–39, 2004.

M. Orton and W. Fitzgerald. A bayesian approach to tracking multiple targets using sensor arrays and particle filters. In *IEEE Transanctions on Signal Processing*, volume 50(2), pages 216–223, 2002.

E. Osuna, R. Freund, and F. Girosi. Training support vector machines:an application to face detection. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 1997.

N. Oudjane, C. Musso, and F. Legland. Improving regularised particle filters. In *Sequential Monte Carlo Methods in Practice.* Springer-Verlag, New York, 2001.

C. Panagiotakis and G. Tziritas. Recognition and tracking of the members of a moving human body. In *Proc. of Int. Workshop on Articulated Motion and Deformable Objects*, pages 86–98, Palma de Mallorca, Spain, 2004.

V. Pavlovic, J. M. Rehg, and J. MacCormick. Learning switching linear models of human motion. In *Neural Information Processing Systems*, pages 981–987, Denver, CO, November 2000.

P. Penev and J. Atick. Local feature analysis: A general statistical theory for object representation. *Neural Systems*, 3(7):477–500, 1996.

A. Pentland, B. Moghaddam, and T. Starner. View-based and modular eigenspaces for face recognition. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, Seattle, WA, June 1994.

P. Perez, C. Hue, J. Vermaak, and M. Gangnet. Color-based probabilistic tracking. In *Proc. of the European Conference on Computer Vision*, volume 1, pages 661–675, Copenhagen, Denmark, 2002.

M. K. Pitt and N. Shephard. Auxiliary variable based particle filters. In *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New York, 2001.

F. Porikli. Integral histogram: A fast way to extract histograms in cartesian spaces. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 829–836, 2005.

C. Rasmussen and G. D. Hager. Probabilistic data association methods for tracking complex visual objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(6):560–576, 2001.

D. B. Reid. An algorithm for tracking multiple targets. In *Proc. of the IEEE Trans. Autom. Control*, volume 6, pages 843–854, 1979.

B. Ristic, S. Arulampalam, and N. Gordon. *Beyond the Kalman Filter - Particle Filters for Tracking Applications.* Artech House, Boston, 2004.

K. Rohr. Towards model-based recognition of human movements in image sequences. *CVGIP: Image Underst.*, 59(1):94–115, 1994.

A. Ross and A. K. Jain. Information fusion in biometrics. *Pattern Recognition Letters*, 24(13):2115–2125, September 2003.

H. A. Rowley, S. Baluja, and T. Kanade. Neural network-based face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(1):23–38, 1998.

F. Samaria and A. Harter. Parameterisation of a stochastic model for human face identification. In *Proc. of the IEEE Workshop on Applications of Computer Vision*, Sarasota (Florida), December 1994.

M. I. Savic and S. K. Gupta. Variable parameter speaker verification system based on hidden markov modeling. In *ICASSP*, pages 281–284, April 1990.

M. Scheutz, J. McRaven, and G. Cserey. Fast, reliable, adaptive, bimodal people tracking for indoor environments. In *Proc. of the IEEE International Conference on Intelligent Robots and Systems*, 2004.

C. Schlegel, J. Illmann, H. Jaberg, M. Schuster, and R. Worz. Vision based person tracking with a mobile robot. In *The British Machine Vision Conference*, 1998.

D. Schulz, W. Burgard, D. Fox, and A. B. Cremers. Tracking multiple moving objects with a mobile robot. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

A. Senior, A. Hampapur, Y.-L. Tian, L. Brown, S. Pankanti, and R. Bolle. Appearance models for occlusion handling. In *Proc. of the IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, Kauai, Hawaii, USA, 2001.

D. Serby, E. Koller-Meier, and L. Van Gool. Probabilistic object tracking using multiple features. In *Proc. of the International Conference on Pattern Recognition*, volume 2, pages 184–187, 2004.

H. Sidenbladh. *Probabilistic Tracking and Reconstruction of 3D Human Motion in Monocular Video Sequences.* PhD thesis, Dept. of Numerical Analysis and Computer Science, KTH, Stockholm, Sweden, 2001.

H. Sidenbladh. Multi-target particle filtering for the probability hypothesis density. In *International Conference on Information Fusion*, pages 800–806, Cairns, Australia, 2003.

N. T. Siebel. *Design and Implementation of People Tracking Algorithms for Visual Surveillance Applications.* PhD thesis, The University of Reading, Reading, UK, March 2003.

K. Smith and D. Gatica-Perez. Order matters: A distributed sampling method for multiple-object tracking. In *Proc. British Machine Vision Conference*, London, UK, 2004.

K. Smith, D. Gatica-Perez, J. M. Odobez, and S. Ba. Evaluating multi-object tracking. In *Proc. of the Workshop on Empirical Evaluation Methods in Computer Vision*, San Diego, CA, USA, 2005a.

K. Smith, G. Gatica-Perez, and J. Odobez. Using particles to track varying numbers of objects. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, CA, USA, 2005b.

D. A. Socolinsky, L. B. Wolff, J. D. Neuheisel, and C. K. Eveland. Illumination invariant face recognition using thermal infrared imagery. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 527–534, 2001.

F. K. Soong, A. E. Rosenberg, and B. H. Juang. A vector quantization approach to speaker recognition. Technical Report 3, AT&T, March 1987.

L. D. Stone, T. L. Corwin, and C. A. Barlow. *Bayesian Multiple Target Tracking*. Artech House, 1999.

M. A. Stricker and M. Orengo. Similarity of color images. In *Storage and Retrieval for Image and Video Databases*, pages 381–392, 1995.

M. Swain and D. Ballard. Color indexing. *International Journal of Computer Vision*, 7:11–32, 1991.

H. Tao, H. Sawhney, and R. Kumar. A sampling algorithm for detecting and tracking multiple objects. In *Proc. of the IEEE ICCV Workshop on Vision Algorithms*, 1999.

S. Thrun, M. Bennewitz, W. Burgard, A.B. Cremers, F. Dellaert, D. Fox, D. Hähnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. Minerva: A second generation mobile tour-guide robot. In *Proc. of the IEEE International Conference on Robotics and Automation*, 1999.

A. Treptow and A. Zell. Real-time object tracking for soccer-robots without color information. *Robotics and Autonomous Systems*, 48(1): 41–48, 2004.

A. Treptow, G. Cielniak, and T. Duckett. Real-time people tracking for mobile robots using thermal vision. *Robotics and Autonomous Systems*, 54(9):729–739, 2006.

M. Turk and A. Pentland. Face recognition using eigenfaces. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, Maui, Hawaii, 1991.

P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, 2001.

S. Waldherr, S. Thrun, R. Romero, and D. Margaritis. Template-based recognition of pose and motion gestures on a mobile robot. In *Proc. of the National Conference on Artificial Intelligence*, pages 977–982, Madison, Wisconsin, USA, 1998.

T. Wilhelm, H. J. Böhme, and H. M. Gross. Sensor fusion for vision and sonar based people tracking on a mobile service robot. In *Int. Workshop on Dynamic Perception*, pages 315–320, Bochum, Germany, 2002.

C. Wong, D. Kortenkamp, and M. Speich. A mobile robot that recognizes people. In *Proc. of the IEEE International Conference on Tools with Artificial Intelligence*, 1995.

C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland. Pfinder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780–785, 1997.

F. Xu, X. Liu, and K. Fujimura. Pedestrian detection and tracking with night vision. *IEEE Transactions on Intelligent Transportation System*, 5(4):1901–1906, 2004.

J. Yang, X. Zhu, R. Gross, J. Kominek, Y. Pan, and A. Waibel. Multimodal people ID for a multimedia meeting browser. In *Proc. of the ACM International Conference on Multimedia (1)*, pages 159–168, 1999.

M. H. Yang, D. Kriegman, and N. Ahuja. Detecting faces in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1):34–58, 2002.

B. H. Yoshimi and P. K. Allen. Active, uncalibrated visual servoing. In *Proc. of the IEEE International Conference on Robotics and Automation*, volume 4, pages 156–161, 1994.

A. Yuille and P. Hallinan. Deformable templates. In A. Blake and A. Yuille, editors, *Active vision*, pages 20–38. MIT, 1992.

W. Zajdel, Z. Zivkovic, and B. J. A. Kröse. Keeping track of humans: have i seen this person before? In *Proc. of the IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005.

L. Zhao and C. Thorpe. Stereo- and neural network-based pedestrian detection. In *Proc. of the IEEE Intelligent Transportation Systems*, Tokyo, Japan, 1999.

Q. Zhao and H. Tao. Object tracking using color correlogram. In *Proc. of the IEEE International Workshop on Performance Evaluation of Tracking and Surveillance*, pages 263–270, Beijing, China, October 2005.