8-2009

Automated Surveillance Systems with Multi-Camera and Robotic Platforms

Chung Hao Chen
University of Tennessee - Knoxville

To the Graduate Council:

I am submitting herewith a dissertation written by Chung Hao Chen entitled "Automated Surveillance Systems with Multi-Camera and Robotic Platforms." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Electrical Engineering.

Mongi A. Abidi, Major Professor

We have read this dissertation and recommend its acceptance:

Hairong Qi, Seddik M. Djouadi, Andreas Koschan, Frank M. Guess

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

To the Graduate Council:

I am submitting herewith a dissertation written by Chung-Hao Chen entitled "Automated Surveillance Systems with Multi-Camera and Robotic Platforms." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor, with a major in Electrical Engineering.

Mongi A. Abidi, Major Professor

We have read this dissertation
and recommend its acceptance:

Hairong Qi

Seddik M. Djouadi

Andreas Koschan

Frank M. Guess

Accepted for the Council:

Carolyn R. Hodges
Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Automated Surveillance Systems with Multi-Camera and Robotic Platforms

A Dissertation
Presented for the
Doctor of Philosophy
Degree
The University of Tennessee, Knoxville

Chung-Hao Chen
August 2009

Acknowledgement

First and foremost, I am deeply indebted to my parents, Kuei-Lan Hsieh and Eing-Ming Chen, who have always encouraged me to pursue higher education and supported me without reservation, to my beloved wife, Chiung-Ju Yeh, who has always stood beside me during this journey, and to my sister, Li-Wen Chen, and her family for their love.

I would like to thank my advisor, Dr. Mongi Abidi, who has made me who I am and who I will be in my professional career. His willingness to support my work and his guidance throughout my studies has allowed me to develop my abilities to think through the problems and skills to find the answers. I sincerely thank him for this opportunity and everything he had done for me. Special thanks indeed go to Dr David Page and Dr. Andreas Koschan, whose regular technical comments provided me valuable guidance throughout my research. Also, I would like to thank, Dr. Frank Guess, Dr. Hairong Qi, and Dr. Seddik Djouadi. Their advice and counsel have been of equal importance. I greatly appreciate their time and input to this dissertation.

Within the IRIS group, I express my sincerest gratitude for the opportunity and many conversations and assistances that have had tremendous impacts on my research.

To my mentor, Dr. H. Lee Martin, your encouragement and advice to my research, career pursuit, and personal life are beyond any words that I can express for my appreciation.

Finally, to my dear friends, Paul Slay and your family, Kevin Zinn and you family, Jon Blosser and your family, Mike Rogers and your family, Brent Wood and your family, and Tom Wilson and your family, I cannot adjust to the life here easily without your care.

Thank you all.

Abstract

This dissertation addresses automated surveillance systems focusing on four topics: (1) spatial mappings of omnidirectional and PTZ cameras, and PTZ and PTZ cameras; (2) target hopping application for dual camera systems; (3) camera handoff and placement; (4) the mobile tracking platform. The four topics represent four contributions in this dissertation.

Dual camera systems have been widely used in surveillance because of the ability to explore the wide field of view (FOV) of the omnidirectional camera and the wide zoom range of the PTZ camera. Most existing algorithms require a priori knowledge of the projection models of omnidirectional and PTZ cameras to solve the spatial mapping between any two cameras. The proposed methods not only improve the mapping accuracy by reducing the dependence on the knowledge of the projection model but also improved flexibility in adjusting to varying system configurations. The omnidirectional camera is capable of multi object tracking while the PTZ camera is able to track one individual target at one time to maintain the required resolution. It becomes necessary for the PTZ camera to distribute its observation time among multiple objects and visit them in sequence. In comparison with the sequential visiting and nearest neighbor methods, the proposed adaptive algorithm requires less computational and visiting time.

Tracking with multiple cameras is mainly the consistent labeling or camera handoff problem. An automatic calibration procedure combined with Wilcoxon Signed-Rank Test is proposed to solve the consistent labeling problem. Meanwhile, we introduce an additional constraint to search for optimal cameras' overlapped field of views (FOVs) and resource management approach to improve camera handoff performance. Experiments show that our proposed camera handoff and placement can outperform existing approaches. .

However, in the majority of surveillance systems, their cameras are stationary. These stationary systems often require the desired object to stay within the surveillance range of the system. Thus, the robotic platform we propose uses a visual camera to sense the movement of the desired object and a range sensor to help the robot detect and then avoid obstacles in real time while continuing to track and follow the desired object. Experiment shows this robotic and intelligent system can fulfill the requirements of tracking an object and avoiding obstacles simultaneously when the object moves in speed of 4 km/hr.

Table of contents

List of figures

List of tables

1 Introduction

Large area surveillance in the context of physical security is a high priority for the Department of Energy (DOE) National Nuclear Security Administration (NNSA) and as such requires constant vigilance for both safeguarding US nuclear assets and fostering nonproliferation on the international stage. As the incremental demand seeks to increase capability, to become more agile, and to reduce costs—"better, faster, cheaper," the enhancement and adoption of emerging technologies in automated surveillance systems for physical security are necessary. Figure 1.1 illustrates applications including crime prevention, border control, asset monitoring, and airport/security that automated surveillance systems cover and its potential problems. In Figure 1.1, we divide automated surveillance systems into two categories, stationary camera and mobile camera platforms and both platforms have the same subsystems, dual camera and multi camera systems, heterogeneous and homogeneous mapping, resource management, camera handoff and placement.

With the increase of the scale and complexity of an automated surveillance system, it becomes increasingly difficult for a single camera to accomplish object tracking and monitoring with the required resolution and continuity. Camera networks emerge and find extensive applications. One popular example is the pair of an omnidirectional and a PTZ camera, referred to as dual camera system. Omnidirectional cameras, equipped with a FOV of 180°×360°, are promising candidates for monitoring multiple latent activities in the area of interest. However, omnidirectional cameras have non-uniform resolution and are unable to provide close observations of particular targets. This is where PTZ cameras augment the surveillance systems. With high mobility and zoom ability, PTZ cameras compensate for the deficiencies of omnidirectional cameras and provide flexible observation of the target at adjustable detail levels. The combination of these two types of cameras facilitates a continuous monitoring of the whole surveillance area and detailed observations of specific targets simultaneously. Since the performance of a dual camera system heavily depends on the accuracy of the spatial mapping between the omnidirectional and PTZ cameras, how to obtain the spatial mapping by a "better, faster, and cheaper" method yields a challenging question.

Similarly, due to the capacity of pan-tilt-zoom (PTZ) cameras to simultaneously cover a panoramic area and maintain high resolution imagery, researches in automated surveillance systems with multiple PTZ cameras have become increasingly important. Most existing algorithms require the prior knowledge of intrinsic parameters of the PTZ camera to infer the relative positioning and orientation among multiple PTZ cameras. To overcome this limitation, we propose a novel mapping algorithm that derives the relative positioning and orientation between two PTZ cameras based on a unified polynomial model. This reduces the dependence on the knowledge of intrinsic parameters of PTZ camera and relative positions.

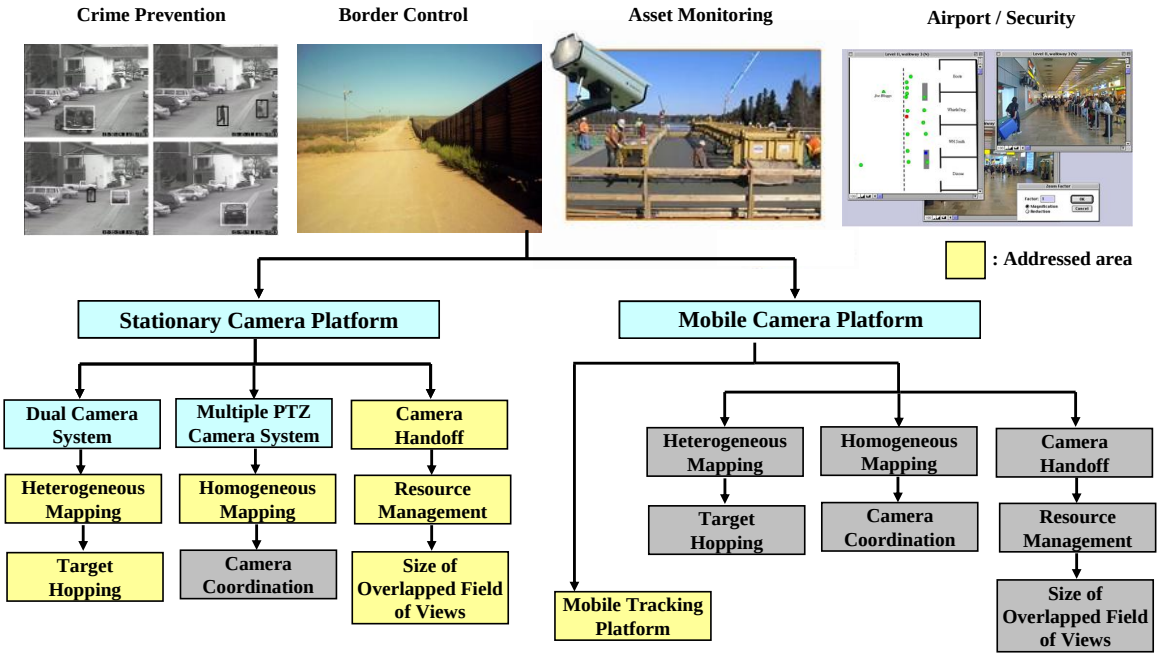


Figure 1.1 Illustration of applications that automated surveillance systems cover and its potential problems. In general, we divide automated surveillance systems into two categories, stationary camera and mobile camera platforms, and both platforms have the same subsystems including dual camera and multi camera systems, heterogeneous and homogenous mapping, resource management, and camera placement and handoff.

The discrepancies in the FOV and resolution levels of the omnidirectional and PTZ cameras not only lead to difficulties in deriving an accurate spatial mapping, which is to be resolved by our geometry and homography calibration methods, but also bring in difficulties in multiple object tracking. Most existing algorithms are only able to handle cameras with similar FOV and resolution levels. Under this condition, the correspondences are of *1*-to-*1* mapping. In a dual camera system, the omnidirectional camera monitors and detects all latent targets in the environment while the PTZ camera only sees a small portion of the environment. Frequently, to achieve the required detail level, the PTZ camera can only monitor a single target at any given time. Therefore, there exists the problem of *N*-to-*1* mapping for a pair of dual cameras, which is referred to as the next best target problem or target hopping application.

In order to expand the availability and applicability of a dual system, we introduce a multi-omnidirectional camera tracking system to improve overall coverage and configuration flexibility relative to commonly used single dual system where single omnidirectional and PTZ camera are used. Tracking with a single camera is a correspondence problem among the tracks of the same object seen from the same camera at different time instances. Tracking with multiple cameras, on the other hand, is a correspondence problem among tracks of the same object seen from different cameras at the same time instance. On the other hand, the goal of multi-omnidirectional camera system is to continuously track the objects of interest without interruptions, which leads to the question: how to manage multiple omnidirectional cameras in terms of camera handoff and placement approaches to achieve the goal.

Even though camera handoff is a crucial step to obtain a continuously tracked and consistently labeled trajectory of the object of interest in multi-camera surveillance systems, most existing camera handoff algorithms concentrate on data association, namely consistent labeling, where images of the same object are identified across different cameras. However, there exist many unsolved questions in developing an efficient camera handoff algorithm. For example, most existing real-time object tracking systems see a decrease in the frame rate as the number of tracked objects increases. To address this issue, our handoff algorithm employs an adaptive resource management mechanism to dynamically allocate cameras' resources to multiple objects with different privileges so that the required minimum frame rate is maintained

In addition, in the majority of surveillance systems or even the dual camera system, their cameras are stationary. These stationary systems do not only require the desired object such as an intruder to stay within the surveillance range of the system, but also need the layout of a monitored area not to change frequently. If the desired object goes beyond this range, it no longer becomes traceable. Similarly, if the layout of a monitored area such as: armory, hazardous materials storage, etc. is changed frequently, the coverage of a secured area will be dependent on the new layout. The last question can be addressed as: how to design a mobile robotic platform capable of tracking the object of interest and avoiding obstacles in real-time for the sake of persisted and automated object tracking. This dissertation work described herewith resolves the aforementioned questions and addresses automated surveillance systems related researches.

The remainder of this chapter outlines the motivation for this research in section 1.1. Section 1.2 gives the pipeline of our system and contributions of our work. Section 1.3 concludes this chapter with the document organization.

1.1 Motivation

In general, the performance of a dual camera system heavily depends on the accuracy of the spatial mapping between the omnidirectional and PTZ cameras. The spatial mapping between the omnidirectional and PTZ cameras can be formulated into the following statement. Given the tracked object's coordinates in the omnidirectional image (x, y), solve for the corresponding pan, tilt, and zoom (θ_P, θ_T, f) for the PTZ camera so that the object can be placed at the image center of the PTZ camera according to a set of equations, ϕ_P, ϕ_T and ϕ_Z, obtained from calibration:

$$\theta_P = \phi_P(x, y),\ \theta_T = \phi_T(x, y),\ f = \phi_Z(x, y). \tag{1.1}$$

Most existing algorithms [Cui98, Scotti05] calibrate the omnidirectional camera assuming a known projection model and relative positions. Based on the projection model, camera calibration is conducted to estimate the camera's intrinsic and projection parameters. These estimated parameters along with the known relative position are used to generate the transformation functions defined in Equation (1.1). Since the calibration results heavily depend on the accuracy of the projection model and professional personnel to carry out, this impedes their direct application to surveillance system with changing configurations. Figure 1.2 shows our intention to obtain the spatial mapping. In Figure 1.2(a), the conventional approaches require the knowledge of both camera's projection models and relative positions to obtain the spatial mapping. In Figure 1.2(b), the intermediary approach relaxes the requirement of known camera's projection model. A polynomial approximation with automated model selection mechanism is incorporated into camera calibration. However, the requirement of known relative position is still needed. In Figure 1.2 (c), the ultimate approach proceeds without the knowledge of either the camera's projection model or the relative position. Polynomial approximation is used to directly model the relation between (x, y) and (θ_P, θ_T, f). The ultimate approach has the highest flexibility in comparison with the conventional and intermediary approaches.

Similarly, surveillance systems with multiple PTZ cameras became popular in the past decade, because of their capacity to simultaneously cover wide area and maintain high object resolution imagery. Due to the time-varying relations among PTZ cameras, how to coordinate multiple PTZ cameras by means of changing their poses to achieve a better observation of the object of interest remains challenging. Thus, the works of Chen and Wang [Chen07A, Chen07B] and Everts et al. [Everts07] proposed to use known intrinsic parameters of PTZ cameras to direct their poses, namely pan, tilt, and zoom values, whenever a change is needed. In other words, we have to individually calibrate each PTZ camera to obtain their intrinsic parameters beforehand. This impedes their direct application to automated surveillance systems with changing configurations and a larger number of PTZ cameras. In particular, due to errors in the estimation of intrinsic parameters of PTZ camera, the works of Chen and Wang need one more optimization process, sensitivity analysis, to obtain the pose relation between PTZ cameras. This increases the system's computational complexity in the calibration process. To overcome their limitations, we propose a novel mapping approach that directly derives a unified polynomial model between

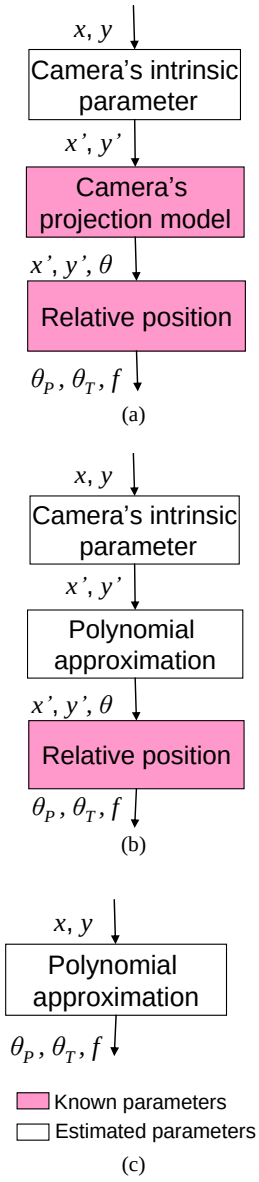


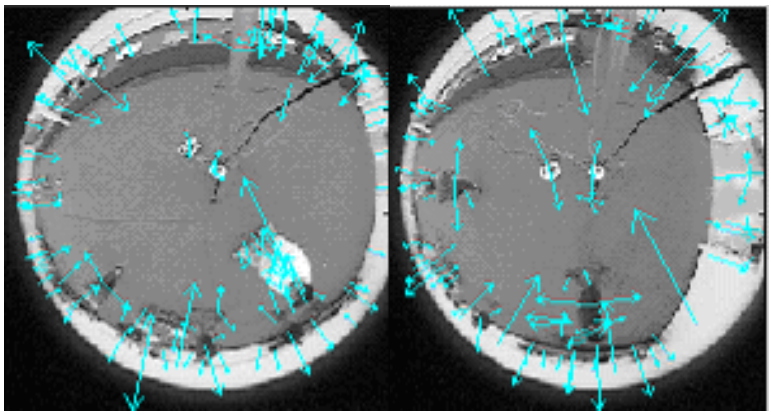
Figure 1.2 Illustration of our intention to obtain the spatial mapping. (a) The conventional approaches require the knowledge of camera's projection model and relative position. (b) The intermediary approach requires the knowledge of relative position. Polynomials are used to approximate various projection models. (c) The ultimate approach directly utilizes polynomials to approximate the relation between (x, y) and (θ_P, $\theta_{i,T}$, f) with no prior knowledge of the cameras' projection models and relative position.

the pan, tilt, and zoom values of PTZ cameras with unknown intrinsic parameters and setups in the scene.

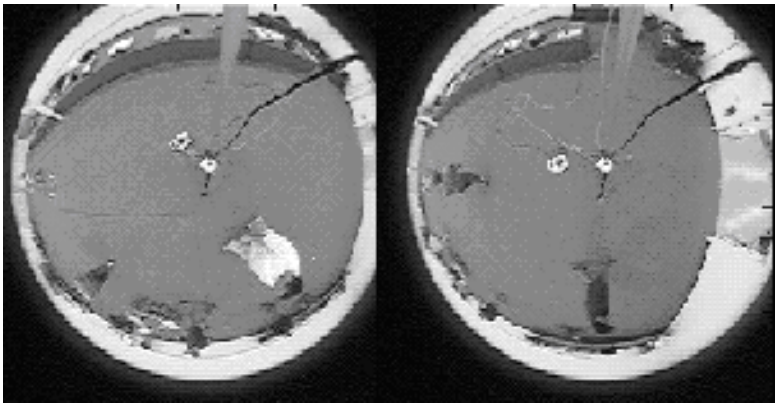
To upgrade the *N*-to-*1* mapping to a *1*-to-*1* mapping, one possible solution is to employ multiple PTZ cameras with the number of PTZ cameras N_c larger than or equal to the maximum number of targets, N_{max}, that may appear in the environment. However, this is usually not practical due to substantially increased cost and idle time for any individual PTZ camera. With limited number of PTZ cameras ($N_c<N_{max}$), an individual PTZ camera is responsible for observing multiple targets that may not fall into its FOV simultaneously. Therefore, a target-hopping scheme is exploited so that the PTZ camera cyclically visits multiple targets according to an automatically adjusted sequence. Each target is assigned a varying number of frames according to its priority. In doing so, all targets in the environment can be monitored with the required object resolution without necessitating more PTZ cameras. The remaining question, referred to as the next best target problem, is how to generate an effective and dynamic visiting sequence for the PTZ camera. The next best target problem is similar to the traveling salesman problem [Applegate06], but with two major differences. (1) The location of each tracked target changes frequently and (2) the computational and traveling time of generating and implementing the optimal visiting sequence affects the system performance significantly. For example, a difference of one or two seconds in the computational and traveling time is marginal in the traveling salesman problem. However, the same amount of time substantially affects a surveillance system with the scale of 10 frames per second because the system may fail to detect abnormal events in real-time.

Most existing consistent labeling approaches have been proved inefficient in some cases. The geometry-based approach usually needs an expensive process in real life surveillance applications to fully calibrate each camera in order to derive 3D information, as pointed out by the paper [Khan03]. This is unnecessary for a camera surveillance system, because most of the information needed can be extracted by observing motion over a period of time [Khan03]. A feature-based approach is not reliable, when the disparity increases. This includes illumination, viewpoint angle [Tuytelaars04], object with different color or textile on front and back, and so forth. Since omnidirectional cameras have non-uniform resolution, it increases the difficulty in finding pixel-to-pixel homography and object-to-object features between two omnidirectional camera images. Consider the work [Lowe04] as an example, which is generally abbreviated as SIFT. It does not only decrease the stability of detection for keypoint location when viewpoint angle increases, but also is not robust to severely distorted images such as images acquired by omnidirectional cameras. Figure 1.3 illustrates that even though SIFT can respectively find keypoints in two omnidirectional images, it cannot find any comparable keypoints in both images. This is caused by the lens distortion and low object resolution. Thus our motivation stems from building an automatic calibration procedure to obtain correspondence information between overlapped omnidirectional cameras without human interventions.

Most existing consistent labeling methods need a certain amount of time to be executed successfully. The size of overlapped field of view (FOV), therefore, should be reserved enough for successfully carrying out consistent labeling, before the object falls out of the FOV of the observing camera. Even though the works [Javed03, Javed05, Kang05, Lim07] can consistently label the object in the disjoint views of two cameras, those algorithms still need a mount of time to recover an untracked object after the camera sees objects. In particular, constraints in their works, such as the size of disjoint view is restrained and the locations of exits and entrances



(a)



(b)

Figure 1.3 Illustration of the SIFT approach for images acquired by two omnidirectional cameras with the same scene. (a) Keypoint locations in the image taken by omnidirectional camera one and two. (b) No keypoints in both images taken by camera one and two are found comparable.

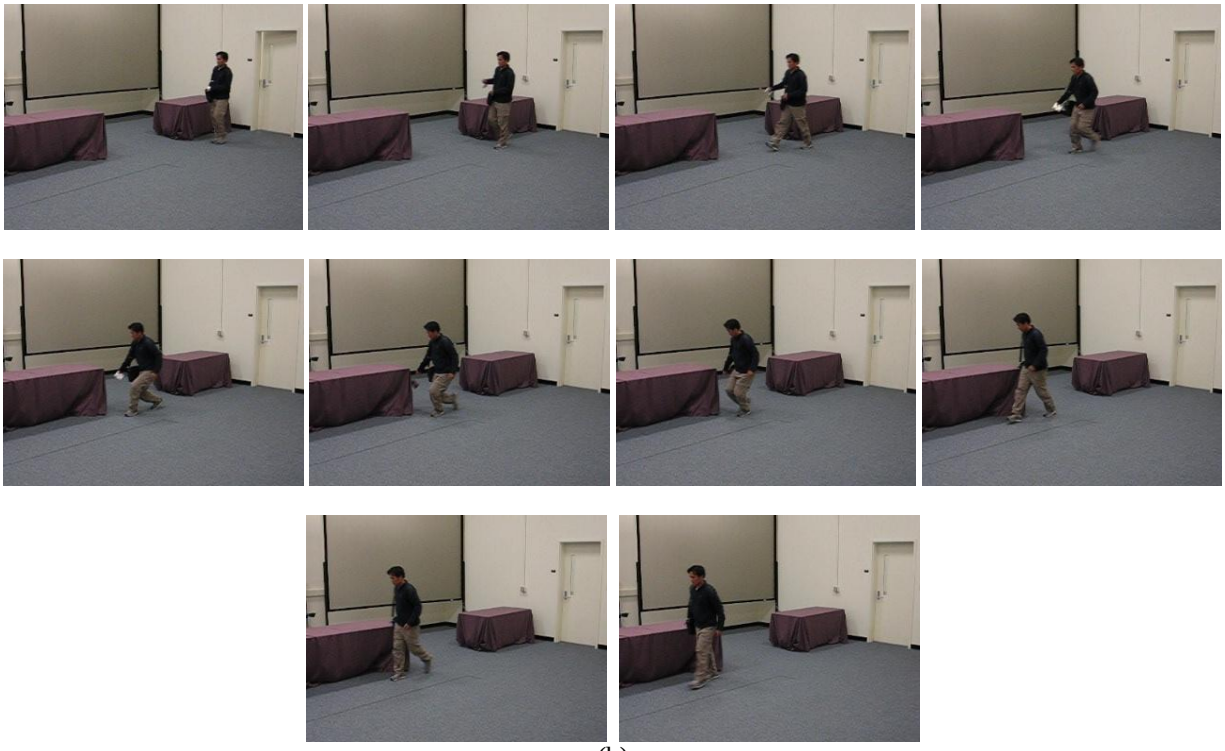
across the cameras have to be correlated, lead to more complicated questions, how to determine the size of disjoint views and where you can have disjoint views in the monitored environment without deteriorating the performance of the automated surveillance system in term of the continuity of the tracked object. Thus a camera placement approach needs to find an optimized tradeoff between the overall coverage and the size of overlapped FOVs to maximize the performance of the automated surveillance system in term of the continuity of the tracked object, which is ignored by most existing camera placement algorithms such as art gallery problem [O'Rouke87] and other most current works [Erdem06, Mittal04].

In addition, most multiple objects tracking systems [Beleznai06, Luo05, Yao06] find it difficult to maintain a constant frame rate given limited resources. Note that frame rates in this paper represent the number of processed frames per second by the tracking system for executing functions such as tracking, crowd segmentation, and behavioral understanding, instead of the number of read-in frames by cameras themselves. This difference occurs due to the tracking system incapable of processing each read-in frame for accommodating the execution of all functions in real-time given limited resources, even though cameras themselves are capable to acquire more frames. Herewith, resources include (I) CPU capacity for executing object tracking, crowd segmentation, and behavior understanding in a automated manner [Hu04], and (II) network bandwidth for exchanging camera handoff information. The computational complexity of most existing tracking systems [Beleznai06, Luo05, Yao06] is of the order from $N_pO(n)$ to $N_pO(n^3)$ [Sebe05], where N_p is the number of tracked objects and n represents the number of steps to execute the algorithm. There inherently exists an upper bound on the number of objects that can be tracked simultaneously without deteriorating the system's frame rate.

Those unprocessed read-in frames may be dropped immediately or reserved for future reference. Regardless of those unprocessed read-in frames being dropped immediately or reserved for future reference, it is crucial for a tracking system to be able to maintain a reasonable frame rate in real-time. A lower frame rate may result in the following problems: (I) the surveillance system's real-time ability to automatically detect a threatening event degrades, causing possible observation leaks. This dangerous loophole impedes the practical application of these real-time multi-camera multi-object tracking systems [Shah03]; (II) the decreased frame rate also affects the performance of consistent labeling and consequently camera handoff, because a successful execution of consistent labeling requires accumulated information of the object of interest over a period of time [Khan03, Fluret08, Guler03]. The reduced frame rate leads to a decreased number of available frames/information for carrying out consistent labeling successfully.

Figure 1.4 illustrates the scenario where an overloaded surveillance system fails to discern a suspicious event. The example system has a frame rate of 4fps when performing multi object tracking, as shown in Figure 1.4(a). The surveillance system suffers from observation leaks and fails to detect the threatening event. When performing single object tracking, the system's frame rate is increased up to 10fps as shown in Figure 1.4(b). The surveillance system successfully detects a threatening event that one person drops a suspicious object and tries to hide it behind the table. From the illustration of Fig. 2, the study of camera overload, in addition to consistent labeling, is another important step to be incorporated into camera handoff. In this paper, we model a multi-object tracking system as a Markov chain and derive the probability of camera overload according to objects' dynamic distributions and priorities. Based on the probability of

(a)



(b)

Figure 1.4 Illustration of observation leak for an overloaded surveillance system to track a different number of targets in the environment. (a) Frames samples at a frame rate of 4fps when performing multiple object tracking. The surveillance system fails to detect the threatening event, causing observation leak. (b) Frames sampled at a frame rate of 10fps when performing single object tracking. The surveillance system successfully detects a threatening event that one pedestrian drops off an object and tries to hide it behind the table. An observation leak occurs because of the degraded frame rate caused by performing multiple object tracking.

camera overload, the camera's limited resources are adaptively assigned to objects with different priorities. In so doing, we are able to dynamically select the objects with higher priority to track and avoid latent camera overload that may lead to a decreased frame rate. Equipped with adaptive resource management, our camera handoff algorithm is capable of not only minimizing the number of dropped objects but also maintaining a constant frame rate to avoid possible observation leaks. The resulting surveillance system has an improved competency in situational awareness and threat assessment. If the object goes beyond the FOV or depth of view (DOV) of the camera in a surveillance system, it no longer becomes tractable. One solution to this problem is to design the system as a mobile system that uses a laser range sensor, and a visual-spectrum camera, to track the moving object and avoid obstacles in real-time. This research topic has been partially studied in several different areas. Studies made by the automotive industry in this area develop systems that assist a human driver for safety or comfort [Philomin00, Perez04]. NASA has applied this to help astronauts carry more equipment while walking on the moon [Graham03]. These systems are primarily concerned with object tracking, and are not concerned with the obstacle avoidance problem. Thus designing a mobile robotic platform capable of tracking the object of interest and avoiding obstacles in real-time is necessary for the sake of persisted and automated object tracking.

1.2 Contributions

The pipeline of this dissertation work is illustrated in Figure 1.5. An automated surveillance system using multiple cameras is developed including heterogeneous mapping of omnidirectional and PTZ cameras, target hopping for the dual camera system, camera handoff and determination of size of overlapped view for multiple omnidirectional cameras, and the mobile tracking platform. Accordingly, our research contributions are listed as follows.

- **Heterogeneous mapping of omnidirectional and PTZ cameras**: Two spatial mapping methods, geometry and homography calibration, are proposed. The geometry calibration method can approximate various projection models, features automatic model selection and straightforward implementation for off-the-shelf cameras, and eliminates the requirement of a known projection model. The homography calibration method directly derives a unified polynomial model between the pan, tilt, zoom values of the PTZ camera and the projected point on the image plane of the omnidirectional camera. In comparison with the reference algorithms [Cui98, Scotti05] that require the knowledge of the camera's projection model, our methods select the optimal model according to a statistical metric or test considering both uncertainty in estimation and modeling. Therefore, the proposed methods feature improved mapping accuracy, reduced computational complexity, and ability to accommodate varying camera configurations.

- **Homogenous mapping of PTZ cameras:** One spatial mapping method is proposed to derive the relation of pan, tilt, and zoom values between any pair of PTZ cameras without prior knowledge of their intrinsic parameters and relative positions. In comparison with the reference algorithm [Chen07A], our proposed approach not only reduces the

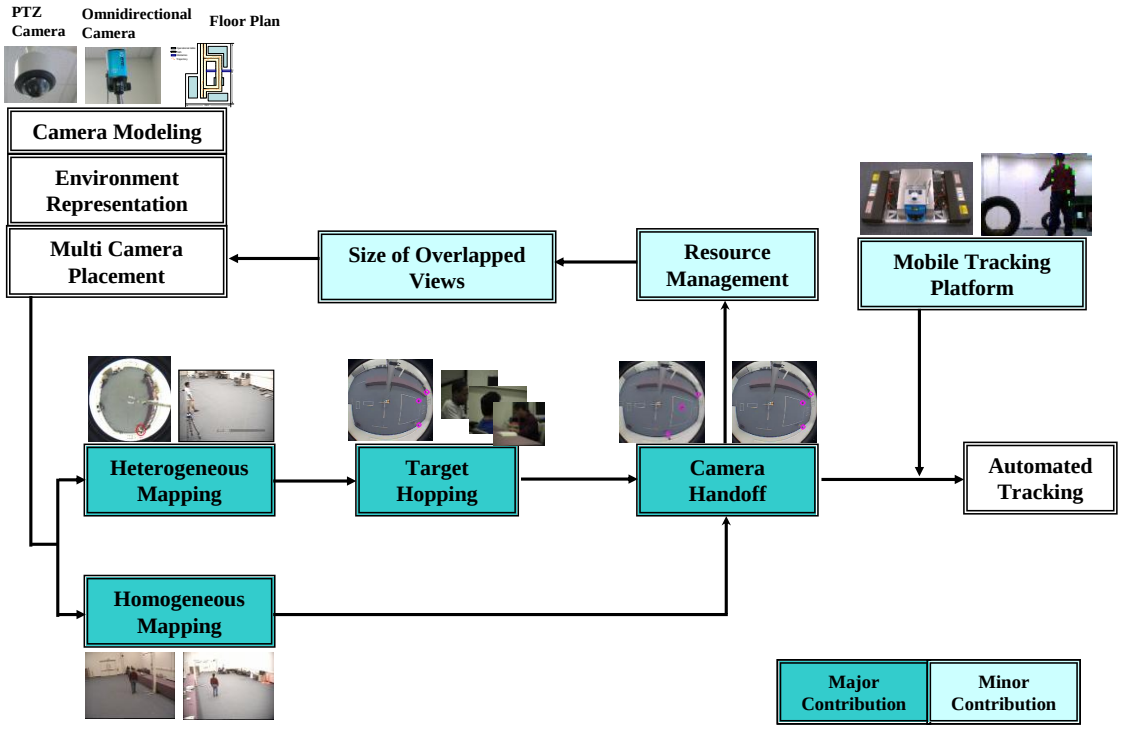


Figure 1.5 The block diagram for automated surveillance systems with multi camera and mobile platforms. During the offline process, our system takes different camera projection models and floor plans as inputs to optimally place cameras. Heterogeneous mapping is used to find the relationship between omnidirectional and PTZ cameras, namely a dual camera system, and homogeneous mapping is used to find the relationship between PTZ cameras. Target hopping is used to guide the coordination in the dual camera system. Multiple omnidirectional cameras are used to cover wide range area, which camera handoff solves the consistent labeling problem and determines the size of overlapped views between omnidirectional cameras. Plus, a resource manage mechanism is added to camera handoff to maintain a fixed frame rate. If the suspicious object is falling out of FOV of currently observing camera, a mobile tracking platform is sent out to continuously track the object.

dependence on the knowledge of intrinsic parameters of PTZ camera, but improves the degree of autonomy and reduces the system's computational complexity at the cost of slightly decreased pixel accuracy. In general, this slightly decreased pixel accuracy does not affect the overall performance for the application of automated surveillance systems, as long as the desired object can be seen within the field of view and can be compensated by consistent labeling approaches [Lowe04] without added cost.

- **Target hopping for the dual camera system:** The next best target (NBT) problem is addressed, which exemplifies a typical problem in multiple object tracking using cameras with different FOVs and resolutions. An adaptive algorithm is designed for a minimized computational and traveling time. The proposed algorithm studies the targets' dynamic distribution in the environment and generates the optimal visiting sequence for the PTZ camera. In comparison with the sequential visiting and nearest neighbor methods, the proposed adaptive algorithm requires less computational and visiting time, which is critical to real-time applications.
- **Camera handoff and determination of size of overlapped view for multiple omnidirectional cameras:** We present a novel solution to the consistent labeling problem in omnidirectional cameras. An automatic spatial mapping procedure considering both the noise inherent to the tracking algorithms used by the system and the lens distortion introduced by omnidirectional cameras is proposed to obtain the correspondences between the trajectories of the same object seen in different omnidirectional cameras without human interventions. For the purpose of automated and persistent object tracking, typical of most surveillance requirements, we propose to use the Wilcoxon Signed-Rank Test for the trajectory association. Our proposed consistent labeling algorithm can perform as accurately as the geometry-based approach without tedious calibration processes and outperform Calderara's homography-based approach [Calderara05]. In the meantime, our proposed camera placement approach that optimally reserves sufficient cameras' overlapped FOVs accomplishes the task of automated and persistent object tracking. As a result, it features a significant increase in the handoff success rate at the cost of slightly decreased coverage as compared to Erdem and Sclaroff's method [Erdem06] without considering the necessary overlapped FOVs.

- **Resource management mechanism:** our handoff algorithm employs an adaptive resource management mechanism to dynamically allocate cameras' resources to multiple objects with different privileges so that the required minimum frame rate is maintained. In other words, the overload probability is one important criterion to evaluate the performance of a multi-camera system fulfilling multiple object tracking. It determines the number of objects that may be dropped due to limited resources. Therefore, in practice, it is desirable to distribute the resources dynamically according to the system's current load and the object's priority rank. Experimental results illustrated that our handoff algorithm outperforms Khan and Shah's method [Khan03] by keeping a higher overall tracking rate and a more stable frame rate. This improves the reliability of the tracking system for continuously tracking multiple objects across multiple cameras

- **Mobile tracking platform:** We describe a robotic application that tracks a moving object by utilizing a mobile robot with multiple sensors. The robotic platform uses a visual camera to sense the movement of the desired object and a range sensor to help the robot detect and then avoid obstacles in real time while continuing to track and follow the desired object. In terms of real-time obstacle avoidance capacity, we also presents a modified potential field algorithm called Dynamic Goal Potential Field algorithm (DGPF) for this robotic application specifically. Experiment shows this robotic and intelligent system can fulfill the requirements of tracking an object and avoiding obstacles simultaneously when the object is moving at about 4 km/hr.

1.3 Document organization

According to the aforementioned topics, the remainder of this document is organized as follows:

- Chapter 2 reviews existing work relevant to this dissertation, including multi-camera surveillance system including camera handoff and its applications, camera placement and robotics.
- Chapter 3 describes the theory for the heterogeneous mapping of omnidirectional and PTZ cameras in a dual system.
- Chapter 4 addresses the theory for the homogeneous mapping of PTZ cameras.
- Chapter 5 introduces the theory for the target hopping application in a dual system.
- Chapter 6 discusses the theory for the camera handoff and the determination of size of overlapped view in multiple omnidirectional cameras.
- Chapter 7 details the resource management mechanism.
- Chapter 8 covers the theory for the mobile tracking platform.
- Chapter 9 illustrates experimental results for the heterogeneous mapping of omnidirectional and PTZ cameras, homogeneous mapping of PTZ cameras, target hopping application, camera handoff and determination of size of overlapped view for multiple omnidirectional cameras, resource management mechanism, and the mobile tracking platform.
- Chapter 10 concludes this dissertation with a summary of accomplished and future works.

2 Related work

This chapter discusses research work in three relevant areas: section 2.1 reviews various types of multi-camera systems in terms of camera handoff algorithms and their coordination. Camera placement algorithms are addressed in section 2.2; an overview of robotic system is given in section 2.3.

2.1 Multiple camera surveillance systems

Various types of cameras and their combinations are used in multi-camera surveillance systems to fulfill the task of persistent and automated tracking. In this section, multi-camera surveillance systems are reviewed according to their camera configurations such as perspective cameras, omnidirectional cameras, PTZ cameras, and dual cameras (omnidirectional and PTZ cameras).

2.1.1 Systems using perspective cameras

Tracking with a single camera is a correspondence problem among the tracks of the same object seen from the same camera at different time instances. Tracking with multiple cameras, on the other hand, is a correspondence problem among tracks of the same object seen from different cameras at the same time instance. Solving this correspondence problem is referred to as camera handoff.

In general, a complete camera handoff scheme regulates the collaboration among multiple cameras and answers the questions of When and Who: when a handoff request should be triggered to secure sufficient time for a successful consistent labeling and who is the most qualified camera to take over the object of interest before it falls out of the FOV of currently observing camera. In order works, camera handoff should comprise three fundamental components, time to trigger handoff process, the execution of consistent labeling, and the selection of the next camera. Nevertheless, most existing camera handoff algorithms discussed in systems based on multiple perspective cameras concentrate on the execution of consistent labeling and ignore the interrelation among those three fundamental components. Thus, our related works start with the consistent labeling problem in systems based on multiple perspective cameras. In addition, in order to completely understand the need for a visual surveillance of

object motion and behaviors, its survey can be found in [Hu04], where the issue of multiple camera tracking is addressed.

In the literature, consistent labeling methods could be grouped into five categories: Feature-based, geometry-based, alignment-based, homography-based, and hybrid approaches. In the feature-based approach, color or other distinguishing features of the tracked objects are matched, generating correspondence among cameras. [Metas02, Mikolajczyk03, Nelson 98, Pope00] uses object contours or region boundaries as features to match objects. [Bay06, Lowe04] use the features that are invariant to image scaling and rotation, and partially invariant to change in illumination and 3D camera viewpoint. The work of Li et al. [Li02] simply finds a color space invariant to illumination. The works of [Balcells05, Kovalev96, Krumn00] are based on an appearance based correlogram model which is combined with histogram information to model color distributions of people and objects in the scene.

In the geometry-based approach [Black01, Kelly95, Mittal01, Tan94], consistent labeling can be established by projecting the trace of the tracked object back into the world coordinate system, and then establishing equivalence between objects projected onto the same location. In the alignment-based approach, the tracks of the same object are recovered across different cameras after aligned by the geometric transformation between cameras. [Caspi00] has extended the alignment problem from a frame alignment point of view to incorporate time information. This work can be deployed when disparity between cameras is small. Thus, [Guler03, Khan03, Lee00] computes the field of view lines (FOV lines) to establish correspondence between trajectories. The homography-based approach [Calderara05, Lee00, Yue04] is to position correspondences between overlapped views in the 2D image plane. As its name suggests, the hybrid approach [Chang01, Kang03] in general is a combination of geometry and feature-based methods. However, the work of Kang et al. [Kang03] is based on probabilistic information mapping or on Bayesian Belief Networks (BBN) [Chang01, Dockstader01], and sometimes a learning is required [Chang00].

Those consistent labeling approaches mentioned above have been proved inefficient in some cases. The geometry-based approach usually needs an expensive process in real life surveillance applications to fully calibrate each camera in order to derive 3D information, as pointed out by the papers of Khan and Shah [Khan03]. This is unnecessary for a camera surveillance system, because most of the information needed can be extracted by observing motion over a period of time. However, the work of Khan and Shah [Khan03] in the alignment-based approach has its limit that if two or more objects cross simultaneously, an incorrect labeling can be established, as pointed out by the homography-base approach of Calderara et al. [Calderara05]. A feature-based approach is not reliable, when the disparity increases. This includes illumination, viewpoint angle [Tuytelaars04], object with different color or textile on front and back, and so forth. Consider the work of Lowe [Lowe04] as an example, which is generally abbreviated as SIFT. It does not only decrease the stability of detection for keypoint location when viewpoint angle increases, but also is not robust to severely distorted images such as images acquired by omnidirectional cameras. Figure 1.3 illustrates that even though SIFT can respectively find keypoints in two omnidirectional images, it cannot find any comparable keypoints in both images. This is caused by the lens distortion and low object resolution.

2.1.2 Systems using omnidirectional cameras

Due to their panoramic views, omnidirectional cameras have been widely used in surveillance systems [Boult04, Boult99, Iwata06, Morita03, Zhu00]. Literature mentions intensive research interest in projection modeling, object tracking, and stereo vision of omnidirectional cameras. The main purpose of projection modeling is to obtain intrinsic and extrinsic parameters for the application such as 3D reconstruction. Thus, the non-parametric approaches [Sturm04] have been studied to learn the projection model without any known parameters. This leads to the situation that a stable calibration in practice is difficult to obtain. [Brauer-Burchardt01, Fitzgibbon01] use auto-calibration techniques to calibrate omnidirectional camera by one image. However, the limitation of these auto-calibration approaches comes from the precision reported. With the increased scale of practical surveillance systems, even equipped with a 360°×90° view, a single omnidirectional camera is incapable of monitoring the entire environment. There has been considerable works on optic flow [Beauchemin95] for handling object tracking in an omndirectional camera. However, most of them can only handle the slow motion and size of targets. Motion parameters are then used as the primary feature to distinguish small objects [Ayer94, Blostein91, Lipton98]. In [Rosin91], a system is presented to use both motion parameters and target size/shape information to classify targets.

However, a network of multiple omnidirectional cameras emerges for improved overall coverage and configuration flexibility. Even though the use of multiple perspective cameras is popular, the discussion of the consistent labeling problem regarding systems using multiple omnidirectional cameras is relatively underdeveloped. In particular, the non-uniform resolution and sever distortion from non-perspective projection result in considerable difficulties in establishing correspondence in adjacent omnidirectional cameras. As we mentioned before, the feature-based approach is not robust in most situations, especially severely distorted images in omnidirectional cameras. Even though the accuracy of fully calibrated cameras is promising for the consistent labeling in omnidirectional cameras, the luxury of fully calibrating cameras is impractical in a real-time case where multiple cameras are used. The works of Calderara et al. [Calderara05] and Lee et al. [Lee00], therefore, are a good start for solving the consistent labeling problem in omnidirectional cameras. Nevertheless, they require considerable manual interventions involved to obtain the homography between two images and neglect both the noise inherent to the tracking algorithms and the lens distortion introduced by omnidirectional cameras. This will reduce the accuracy of consistent labeling in a real-time case. In addition, since omnidirectional cameras have non-uniform resolution, it increases the difficulty in finding pixel-to-pixel homography between two omnidirectional camera images.

2.1.3 Systems using PTZ cameras

For a surveillance system with multiple PTZ cameras, they may change the cameras' pan and tilt angle, and zoom degree from time to time to achieve better monitoring results. However, when the pose of a PTZ camera is changed, we need to know what angle we pan and tilt to, and what degree we zoom to, which is generally called as extrinsic and intrinsic parameters. The need to change their poses in a network of PTZ cameras may be triggered by two objects

walking in an opposite direction so that two objects can be separately tracked by two PTZ cameras for a better and continuous monitoring result. The consistent labeling problem in this case is equivalent to the problem introduced by multiple perspective cameras, since both project models are similar with a drastic case of varying focuses in PTZ camera. On the other hand, the focus of systems using PTZ cameras is the coordination of multiple PTZ cameras. Hence, how to quickly and efficiently calibrate extrinsic and intrinsic parameters of multiple PTZ cameras has become an important issue.

This calibration lays a foundation to the coordination of multiple PTZ cameras. The tradition off-line calibration methods [Hemayed03, Zhang04, Zhang00] are not practical due to the dynamic changes in intrinsic and extrinsic parameters of PTZ camera. The first auto-calibration method [Faugeras92] is due to Faugeras et al. who considers a freely moving camera with unknown but constant internal parameters. Since then, several methods have been proposed to consider pure translation [Moons96] or pure rotation [Hartley94]. More recent methods [Basu97, Junejo07, Hartley99, Heyden97, Heyden99, Kahl00] start considering auto-calibration under varying internal parameters. However, for surveillance systems with multiple PTZ cameras, those elaborate processes and the constraints do not seem to be practical choices. Thus, Remagnino and Jone [Remagnino02] propose an approach that recovers the transformation between the image and the ground plane to find the look-down angle. Chen and Wang [Chen07] propose an efficient approach that the tilt angle and altitude of each PTZ camera are estimated first based on the observation of some simple objects lying on a horizontal plane.

2.1.4 Systems using dual cameras

With the rapidly growing demands in monitoring public areas, substantial developments have been made in multiple camera surveillance systems. One popular example is the pair of an omnidirectional and a PTZ camera, referred to as dual camera system. Omnidirectional cameras, equipped with a FOV of 180°×360°, are promising candidates for monitoring multiple latent activities in the area of interest. However, omnidirectional cameras have non-uniform resolution and are unable to provide close observations of particular targets [Boult04]. This is where PTZ cameras augment the surveillance systems. With high mobility and zoom ability, PTZ cameras compensate for the deficiencies of omnidirectional cameras and provide flexible observation of the target at adjustable detail levels [Trivedi02]. The combination of these two types of cameras facilitates a continuous monitoring of the whole surveillance area and detailed observations of specific targets simultaneously.

Dual camera systems have been widely used surveillance and tracking applications [Boult03, Cui98, Horaud06 Scotti05, Thirthala05, Wang06]. For example, Cui et al. [Cui98] uses background differencing and radial profile for target detection and tracking. A polynomial of degree three is applied to approximate the camera's projection model. Confidence coefficients are assigned to the tracking decisions formed by both cameras. The output follows the one with the higher confidence coefficient. In doing so, tracking ambiguity and occlusion can be resolved. Scotti et al. [Scotti05] pays more attention to the non-uniform resolution of the omnidirectional camera and the corresponding calibration Multiple cues, such as color, shape, and position, are selected as tracking features. The omnidirectional camera performs as a secondary tracker and becomes dominant only when the PTZ camera fails.

In general, the performance of a dual camera system heavily depends on the accuracy of the spatial mapping between the omnidirectional and PTZ cameras. Most existing algorithms calibrate the omnidirectional camera assuming a known projection model [Cui98, Scotti05] which impedes their direct application to surveillance system with changing configurations. The spatial mapping needs to be re-calibrated with sufficient knowledge of camera modeling and environment geometry.

2.2 Determination of size of overlapped views

Regardless of what approaches are used to solve the consistent labeling problem, either multi-frame solutions such as alignment-based and homography-based approaches or the time-consuming solutions such as feature-based and hybrid-based based approaches need a certain amount of time to execute consistent labeling successfully. The size of overlapped FOV, therefore, should be reserved enough for successfully carrying out consistent labeling before the object falls out of the FOV of the currently observing camera. Even though the works of Javed et al [Javed03, Javed05], Kang et al. [Kang05], and Lim et al. [Lim07] can consistently label the object in the disjoint views of two cameras, those algorithms still need a mount of time to recover an untracked object after the camera sees objects. In particular, constraints in their works, such as the size of disjoint view is restrained and the locations of exits and entrances across the cameras have to be correlated, lead to more complicated questions, how to determine the size of disjoint views and where you can place them in the monitored environment without deteriorating the performance of the automated surveillance system in terms of the continuity of the tracked object. In addition, those tracking systems cannot detect the occurrence of unusual events due to the lack of clear and continuous views on the object. This may cause a serious loophole in a surveillance system. Nevertheless, most existing camera placement algorithms do not pay an attention to the problem of how to determine the optimized tradeoff between coverage and size of overlapped FOVs to secure the continuity of tracked objects.

The Art Gallery Problem (AGP) [Chyatal75, Fisk78, O'Rourke87,] are to determine the minimum number of guards to maximize the coverage of observed polygon. A variant of the AGP is known as Watch-men Tour Problem [Carlesson91, Efrat00, Guibas99] where the objective is to find the optimal number and route for guards to guarantee the detection of an intruder. Suzuki et al. [Suzuki01] introduces another variant where guards are allowed to more only the boundary of the polygon. Similarly, Floodlight Illumination Problems [Bose97, Estivill-Castro95] deal with the illumination of planar regions by light source. There is also related work in robotic motion control for video surveillance. Cortes et al. [Cortes02] computes optimal locations for mobile sensors to acquire information. Task-based computer vision such as object inspection has a long history in finding the next best camera view [Arbel01, Maver93, Pito99, Tarabanis95]. An automated surveillance system deploys a network of camera to select the best view of the tracked object [Batista98, Cai99, Doubek04, Mikic00].

Mittal and Davis [Mittal04] propose a probabilistic camera planning framework with a visibility analysis, where the visibility of potential subjects over possible camera configurations is evaluated. Since their methods use a local optimization method to solve a highly non-linear constrained optimization problem, they cannot guarantee to be optimal up to the current sample

space. Erdem and Sclaroff [Erdem06] propose the global optimization to solve the problem. In the meantime, they additionally allow modeling of visibility obstructions and cost constraints. Even though the works of Erdem and Sclaroff [Erdem06] and Mattel and Davis [Mittal04] consider camera constraints such as: FOV, spatial resolution, depth of field (DOF), minimal cost, etc. to solve the camera placement problem in a surveillance system, they still do not provide a solution to optimally determine the size of overlapped FOV for carrying out camera handoff successfully.

2.3 Mobile tracking platform

The stationary camera systems require the desired object to stay within the surveillance range of the system. If the object goes beyond this range, it no longer becomes tractable. One solution to this problem is to design the system as a mobile system to track the moving object and avoid obstacles in real time. In this section, mobile robotic platform is reviewed according to path planning and object tracking.

2.3.1 Path planning

Lozano-Pkrez [Lozano-Pkrez81, Lozano-Pkrez87] first proposes the configuration space approach, where the workspace, obstacles, and path are mapped into the joint space of the manipulator in the area of path planning. There are several ways to find the obstacle-free path in the configuration space. The potential field method [Khatib86] is widely used. The goal position is represented by a positive potential, when the obstacles are represented by negative potentials. However, the robot may be trapped into a local minimum, especially when the environment is cluttered. Several method attempts to address the problem in the potential field method. Warren et al. [Warren89] modifies the selected trial path to reduce the probability of be trapped. Barraquand et al. [Barraquand91] suggests that using several potentials concurrently can effectively deal with local minima. Borenstein and Koren [Borenstein91] develop a vector field histogram, which is based on the earlier certain grid and occupancy grid, to pick up a heading direction.

However, this problem can be completely avoided if harmonic potential functions [Kim92] are used. On the other hand, once a path is generated, the required joint angles can be calculated by inverse kinematics which can be accomplished by pseudoinverse formulations [Maciejewski85, Namamura91], extended Jacobian [Baillieul86], or transpose of the Jacobian [Das88]. These concepts are summarized by Latombe [Latombe91].

Some researches such as sequential search strategy [kant90], configuration control [Seraji91], multiplex joint method [Hennessey99], and so forth see alternate methods to the configuration method. Ulrich et al. [Ulrich00] develops VFH* algorithm to guide the robot around the obstacle. In particular, Jing et al. [Jing04] indicates the simple repulsive and attractive information model cannot completely and accurately reflect the actual states ad real motion purpose of the mobile robot. Thus, Cen et al. [Cen07] proposes to use a new artificial coordinating potential to fulfill the task.

2.3.2 Object tracking

The problem of estimating the positions of moving objects is an important problem in mobile robotics. Knowledge about the position of moving objects can be used to improve the behavior of the surveillance system. Joint Probabilistic Data Association Filters (JPDAFs) [Bar-Shalom88, Black98, Cox93] is a very popular approach to track the moving object. They compute a Bayesian estimate of the correspondence between features detected in the sensor data and the different objects to be tracked. More recently, particle filters have been introduced to estimate non-gaussian, non-linear dynamic processes [Fox99, Gordon93]. They have been applied with great success to different state estimation problems including visual tracking [Black98, Isard96], mobile robot localization [Fox99], and dynamic probabilistic networks [Kanazawa95]. The advantage of this method lies in the ability to represent multi-modal state densities, a property which has been shown to intrease the robustness of the underlying state estimation process [Gutmann98, Maccormick99]. Unfortunately, the computation complexity of the method is high [Schulz01].

The active vision system [Barreto98, Das95, Shibata02, Shibata04] exemplifies the ability for visual tracking for moving targets with active vision robot. In many of conventional approaches, the target motion is estimated in the work space, and then the camera motion is controlled with the prediction of the target position to avoid the tracking delay. Several effective methods are proposed, Kalman filter [Corke96], α-β-γ filter [Allen93], AR model [Koivo91], and so on. Studies made by the automotive industry in this area develop systems that assist a human driver for safety or comfort [Perez04, Philomin00]. NASA has applied this to help astronauts carry more equipment while walking on the moon [Graham03]. The robot BIRON [Haasch04] generates person candidates based on audible clues combined with rule-based detection. Albert [Blanco03] uses a background subtraction method to dynamically detect moving legs for tracking. Human detection without using laser scanner is done by PERSES [Bohme03].

3 Heterogeneous mapping theory

In general, the performance of a dual camera system heavily depends on the accuracy of the spatial mapping between the omnidirectional and PTZ cameras. Most existing algorithms calibrate the omnidirectional camera assuming a known projection model [Cui98, Scotti05], which impedes their direct application to surveillance system with changing configurations. The spatial mapping needs to be re-calibrated with sufficient knowledge of camera modeling and environment geometry. In this paper, we investigate approaches to improve the performance of a dual camera system in terms of both accuracy and flexibility of adapting to various system configurations. Two methods, (a) geometry calibration and (b) homography calibration, are proposed. The common setup of a dual camera system is shown in Figure 3.1.

The geometry calibration method exploits a unified polynomial imaging model discussed in [Kannala04]. From an automatic parameter selection method based on the Akaike information criterion (AIC) [Akaike74], a unified and mathematically precise relation between the omnidirectional and PTZ camera is formulated. The proposed approach, therefore, can approximate various projection models, features automatic model selection and straightforward implementation for off-the-shelf cameras, and eliminates the requirement of a known projection model. The homography calibration method directly derives a unified polynomial model between the pan, tilt, zoom values of the PTZ camera and the projected point on the image plane of the omnidirectional camera and chooses significant predictor variables based on the F-test [Neter04, Wackerly02]. Compared with the geometry calibration method, it further reduces the requirement of known cameras' relative positions. This approach maintains comparable accuracy with improved robustness to environment constraints.

The remainder of this chapter is organized as follows. The problem definition is given in Section 3.1. Section 3.2 describes our geometry and homography calibration methods.

3.1 Problem definition

In Figure 3.1, the optical centers of the omnidirectional and PTZ cameras are at (0, 0, Z_o), and (X_P, Y_P, Z_P), respectively. The pan axis of the PTZ camera is parallel to the optical axis of the omnidirectional camera. Given the i^{th} point $\mathbf{P}_i(X_i, Y_i, Z_i)$ in world coordinates, the pan $\theta_{i,P}$, tilt $\theta_{i,T}$, and zoom f_i required for the PTZ camera to be directed at $\mathbf{P}_i$ with the given object resolution are:

$$\theta_{i,P} = \tan^{-1}\left(\frac{X_i - X_P}{Y_i - Y_P}\right), \tag{3.1}$$

$$\theta_{i,T} = \tan^{-1}\left(\frac{R_i}{Z_i - Z_P}\right), \tag{3.2}$$

and

$$f_i = \lambda\sqrt{R_i^2 + (Z_i - Z_P)^2}\,, \tag{3.3}$$

where $R_i = \sqrt{(X_i - X_P)^2 + (Y_i - Y_P)^2}$ and λ is a scalar depending on the given object resolution.

The imaging process of an omnidirectional camera does not comply with the traditional perspective projection. Let θ_i denote the angle between the incoming ray and the optical axis. The following relation holds:

$$\theta_i = \tan^{-1}\left(\frac{\sqrt{X_i^2 + Y_i^2}}{Z_i - Z_o}\right), \tag{3.4}$$

Let r_i denote the distance between the projected point $\mathbf{p}_i(x_i, y_i)$ and the principle point. The perspective projection is characterized by $r_i = f_o \tan\theta_i$, where f_o represents the camera's focal length. To realize a wider opening angle, this relation is changed. Various projection models exist in literature [Kannala04], such as the equidistance projection $r_i = f_o\theta_i$, the equisolid angle projection $r_i = 2f_o \sin(\frac{\theta_i}{2})$, the stereographic projection $r_i = 2f_o \tan(\frac{\theta_i}{2})$, *etc*. For automatic calibration, it would be useful to have one unified model for different types of cameras.

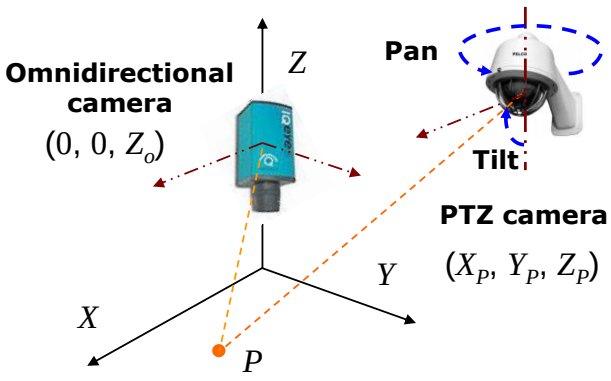


Figure 3.1 A typical setup of a dual camera surveillance system, where one omnidirectional camera is used to obtain all latent objects and then directs one PTZ camera to acquire a better monitoring result.

Therefore, a general form for dioptric omnidirectional cameras is proposed in [11], where a polynomial approximation of degree C is used:

$$r_i = f_o \sum_{j=1,odd}^{C} c_j (\theta_i)^j, \quad (3.5)$$

where c_j represents the approximation coefficient. In a dual camera system, the omnidirectional camera monitors the whole environment and detects multiple motions. Afterwards, the PTZ camera is directed to the detected motions for a close-up observation. Since the omnidirectional camera sees all objects in the environment while the PTZ camera can only place one object at its image center at one time due to its limited FOV for the required object resolution, a sequence of pan/tilt/zoom values should be issued so that the PTZ camera can visit all the objects of interest in a given period of time. To fulfill such functionality, we need to answer two questions. (1) Given a detected motion, where to direct the PTZ camera? (2) Given multiple detected motions, in what sequence to direct the PTZ camera?

The first question, referred to as the spatial mapping between the omnidirectional and PTZ cameras, can be formulated into the following statement. Given the i^{th} tracked object's coordinates in the omnidirectional image (x_i, y_i), solve for the corresponding pan, tilt, and zoom $(\theta_{i,P}, \theta_{i,T}, f_i)$ for the PTZ camera so that the object can be placed at the image center of the PTZ camera according to a set of equations obtained from calibration:

$$\theta_P = \phi_P(x, y),\ \theta_T = \phi_T(x, y),\ f = \phi_Z(x, y), \quad (3.6)$$

Assuming that during the period of time from t_k to $t_k+\Delta t$, the omnidirectional camera detects N_k objects, a sequence of $S_k=(s_1, s_2, \ldots, s_{N_k})$ is obtained as a permutation so that the corresponding $(\theta_{i,P}, \theta_{i,T}, f_i)$ are sent to the PTZ camera in the same order. That is $(\theta_{s_1,P}, \theta_{s_1,T}, f_{s_1})$ is sent to the PTZ camera first followed by $(\theta_{s_2,P}, \theta_{s_2,T}, f_{s_2})$with $(\theta_{s_{N_k},P}, \theta_{s_{N_k},T}, f_{s_{N_k}})$ at the end. There exist $\prod_{j=0}^{j=N_k-1}(N_k - j)$ choices of S_k. The second problem is to find the optimal S_k^* in real-time so that the maximum number of objects can be visited.

3.2 Dual camera mapping

In this section, we derive the equations that solve $(\theta_{i,P}, \theta_{i,T}, f_i)$ from (x_i, y_i). Two types of methods are discussed: (a) geometry and (b) homography calibrations. The geometry calibration requires the knowledge of the relative position between the omnidirectional and PTZ cameras, or equivalently the coordinates of their optical centers $(0, 0, Z_o)$ and (X_P, Y_P, Z_P). In comparison, the homography calibration relaxes this requirement at the cost of slightly degraded mapping accuracy. Due to the ambiguity of 2D to 3D mapping, the following derivation assumes planar motion, which is sufficient to represent the motion of pedestrian and traffic.

3.2.1 Geometry calibration

The key problem in the calibration of an omnidirectional camera is to find the appropriate approximation of the projection model. A camera calibration method with automatic model selection using the unified polynomial projection model is discussed in [Broaddus05, Orekhov07]. Statistical model selection is used to optimize the model parameters when several competing models can be used to explain an observation. In our applications, model selection optimizes the polynomial degree in Equation (3.5). The AIC criterion [Akaike74]

$$AIC = -2\log L(\omega;\mathbf{p}_i) + 2N_\omega, \tag{3.7}$$

is used, where $L(\omega;\mathbf{p}_i)$ is the likelihood of the model parameters ω. The model parameters ω include a total of N_ω camera intrinsic and extrinsic parameters, such as focal length f_o, polynomial approximation coefficients c_j, and the height of the camera Z_o. Assuming a Gaussian distribution of $\mathbf{p}_i$

$$\Pr(\mathbf{p}_i \mid \omega) = \prod_i \left(\frac{1}{\sqrt{2\pi}\sigma}\right) \exp\left\{e_i^2 / (2\sigma^2)\right\}, \tag{3.8}$$

where $e_i = \|\mathbf{p}_i - \breve{\mathbf{p}}_i\|$ and $\breve{\mathbf{p}}_i$ is the estimated projection based on ω, we have

$$\log L(\omega;\mathbf{p}_i) = \log \Pr(\mathbf{p}_i \mid \omega) \propto -\frac{1}{2\sigma^2}\sum_{i=1}^{N_p} e_i^2, \tag{3.9}$$

where N_p denotes the number of points used in calibration and the AIC criterion reduces to

$$AIC = \frac{1}{\sigma^2}\sum_{i=1}^{N_p} e_i^2 + 2N_\omega. \tag{3.10}$$

The optimal polynomial approximation is achieved by minimizing the AIC criterion.
The model selection algorithm proceeds as follows:
(1) Increase C.
(2) Perform camera calibration and obtain ω.
(3) Compute the corresponding model selection measure AIC.
(4) If the measure keeps decreasing, go to step (1). Otherwise, stop and output camera calibration results.

From camera calibration, the camera's intrinsic parameter K and projection model $r = \varphi(\theta)$ are known. Given a point in the image plane $\mathbf{p}(x,y)$, the projected point in a normalized image plane with f_o=1 is $\begin{bmatrix} x' & y' & 1 \end{bmatrix}^T = K^{-1}\begin{bmatrix} x & y & 1 \end{bmatrix}^T$. The incident angle can then be expressed as $\hat{\theta} = \varphi^{-1}(r')$ with

$r' = \sqrt{x'^2 + y'^2}$. With known relative position between the omnidirectional and PTZ cameras, a point restricted in the ground plane with Z=0 is estimated by:

$$\hat{X} = Z_o \tan\hat{\theta}\frac{x'}{r'},\ \hat{Y} = Z_o \tan\hat{\theta}\frac{y'}{r'}. \tag{3.11}$$

In consequence, the estimated pan, tilt, and zoom values are

$$\hat{\theta}_P = \tan^{-1}\left(\frac{\hat{X} - X_P}{\hat{Y} - Y_P}\right),\ \hat{\theta}_T = \tan^{-1}\left(\frac{\hat{R}}{Z_P}\right),\ \hat{f} = \lambda\sqrt{\hat{R}^2 + Z_P^2}, \tag{3.12}$$

where $\hat{R} = \sqrt{(\hat{X} - X_P)^2 + (\hat{Y} - Y_P)^2}$.

3.2.2 Homography calibration

The disadvantage of the geometry calibration method lies in the assumption of known relative position between the omnidirectional and PTZ cameras, the accuracy of which also affects the performance of spatial mapping. To automatically include the relative position into the calibration process, we propose to use a set of polynomials to directly relate (x_i, y_i) and ($\theta_{i,P}$, $\theta_{i,T}$, f_i).

We study the correlation between (x_i, y_i) and ($\theta_{i,P}$, $\theta_{i,T}$, f_i) in the search for the optimal modeling approach. Table 3.1 shows the mean correlation values averaged across a variety of dual camera system setups. Figure 3.2 shows one scatter plot matrix between (x_i, y_i) and ($\theta_{i,P}$, $\theta_{i,T}$, f_i) for one system setup. In Table 3.1, we can see that the correspondences between (x_i, y_i) and ($\theta_{i,P}$, $\theta_{i,T}$, f_i) are highly correlated, which reaches up to 74% between y_i and $\theta_{i,T}$, 67% between $\theta_{i,P}$ and y_i, and 76% between y_i and f_i. In Figure 3.2, taking the plot of f_i in the third row against the plot of y_i in the fifth column as an example, we can see strong linear tendency between f_i and y_i. Alternatively, by viewing the first column, $\theta_{i,P}$, we can compare the plots of $\theta_{i,T}$, f_i, x_i, y_i, and x_iy_i against $\theta_{i,P}$ and observe the linear tendency between $\theta_{i,P}$ and y_i, and also between $\theta_{i,T}$ and y_i. According to the correlation table in Table 3.1 and the scatter plot matrix in Figure 3.2, the multiple regression model [Neter04, Wackerly02] appears to be a good candidate considering both accuracy and computational complexity, in comparison with complicated non-linear fitting algorithms such as the Levenberg-Marquardt algorithm.

Table 3.1 Averaged correlation between (x_i, y_i) and ($\theta_{i,P}$, $\theta_{i,T}$, f_i)

	$\theta_{i,P}$	$\theta_{i,T}$	f_i	x_i	y_i	$x_i y_i$
$\theta_{i,P}$	1.0000	0.5577	-0.6094	-0.6467	0.6747	0.2375
θ_i	0.5577	1.0000	-0.7014	-0.0281	0.7434	0.7080
f_i	-0.6094	-0.7014	1.0000	0.0283	-0.7657	-0.6998
x_i	-0.6467	-0.0281	0.0283	1.0000	-0.1855	0.3311
y_i	0.6747	0.7434	-0.7657	-0.1855	1.0000	0.8353
$x_i y_i$	0.2375	0.7080	-0.6998	0.3311	0.8353	1.0000

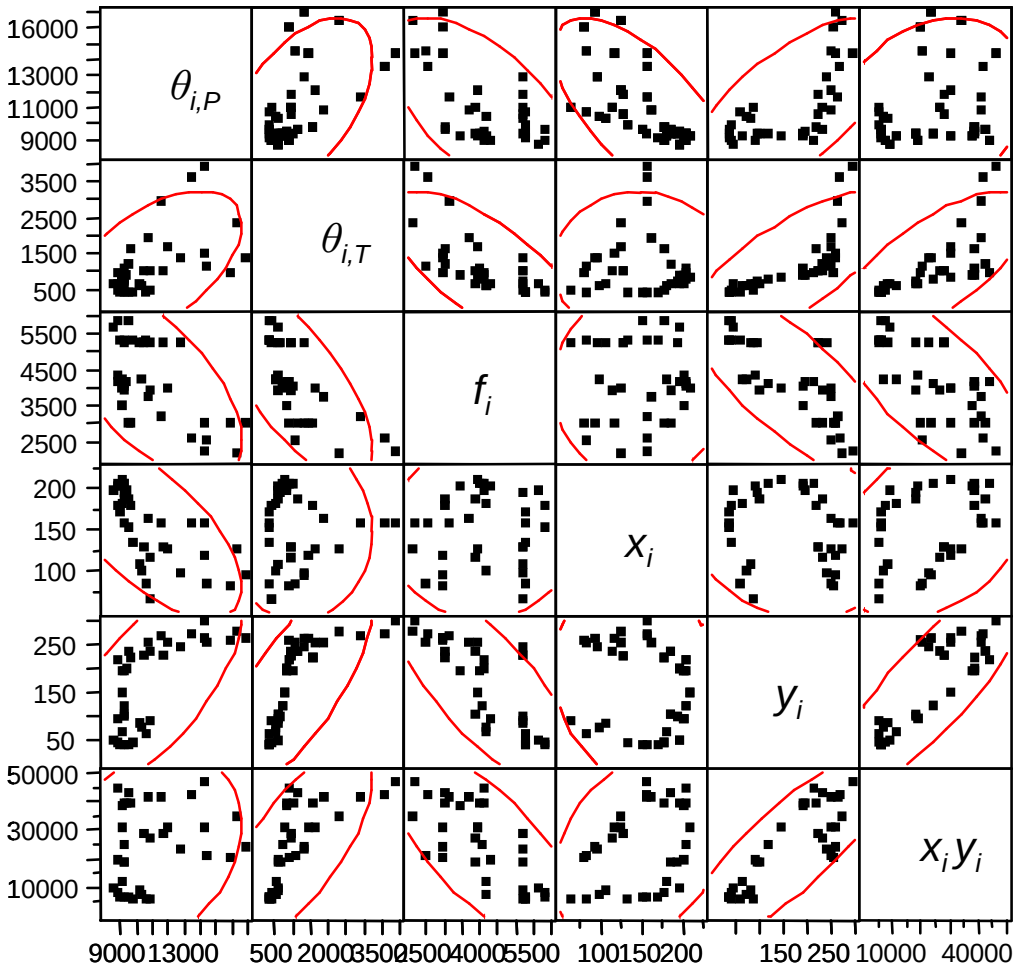


Figure 3.2 Illustration of one set of the scatter plot matrix between (x_i, y_i) and ($\theta_{i,P}$, $\theta_{i,T}$, f_i). The correspondences between (x_i, y_i) and ($\theta_{i,P}$, $\theta_{i,T}$, f_i) are highly correlated, which validates the use of the multiple regression model.

Since the derivations for pan, tilt, and zoom functions are similar, in the following discussion, we will take the pan angle, $\theta_P = \phi_P(x, y)$, as an example to save space. In general, we first fit a model with all possible predictor variables such as x, y, xy, x^2, xy, y^2, … , x^n, xy^{n-1}, … , and y^n. Therefore, let w_i, where $i=1,...k$, represent these k predictor variables. The pan angle in a complete model can then be expressed as

$$\hat{\theta}_{P,C} = \beta_0 + \beta_1 w_1 + \beta_2 w_2 + ... + \beta_k w_k + \varepsilon_C, \tag{3.13}$$

where $\beta_{i,}$ denotes the model fitting parameter and ε_C is a random error term with $E\{\varepsilon_C\} = 0$. Usually not all predictor variables are equally significant. A subset of these variables can be found for a reduced model:

$$\hat{\theta}_{P,R} = \beta_0 + \beta_1 w_1 + \beta_2 w_2 + ... + \beta_g w_g + \varepsilon_R, \tag{3.14}$$

where $g < k$ and ε_R is a random error term with $E\{\varepsilon_R\} = 0$. Let SSE_C and SSE_R denote the sum of squared error of the complete and reduced models.

$$\begin{aligned} SSE_C &= \Theta_{P,C}^T \Theta_{P,C} - \left[W_{P,C}^T W_{P,C} \right]^{-1} W_{P,C}^T \Theta_{P,C} \left] W_{P,C}^T \Theta_{P,C} \right. \\ SSE_R &= \Theta_{P,R}^T \Theta_{P,R} - \left[W_{P,R}^T W_{P,R} \right]^{-1} W_{P,R}^T \Theta_{P,R} \left] W_{P,R}^T \Theta_{P,R} \right. \end{aligned}, \tag{3.15}$$

where $\Theta_{P,C}$ / $\Theta_{P,R}$ is the vector of all response variables in a complete/reduced model and $W_{P,C}$ / $W_{P,R}$ is the vector of all predictor variables w_k / w_g in a complete/reduced model.

Intuitively, if w_1 , w_2 , …, and w_k are important information contributing variables, the complete model in Equation (3.13) should have a smaller prediction error than the reduced model in Equation (3.15): $SSE_C \le SSE_R$. The greater the difference $(SSE_R - SSE_C)$ is, the stronger is the evidence to support the alternative hypothesis that w_1, w_2, …, w_k are significant information contributing terms and to reject the null hypothesis:

$$H_0 : \beta_{g+1} = \beta_{g+2} = ... = \beta_k = 0 . \tag{3.16}$$

Define a testing variable:

$$F^* = \frac{\dfrac{SSE_R - SSE_C}{k-g}}{\dfrac{SSE_C}{n-(k+1)}}, \tag{3.17}$$

where n is number of data trails used in the model. If the null hypothesis H_0 is true, F^* should possess an F distribution with $(k-g)$ numerator degrees of freedom and $n-(k+1)$ denominator degrees of freedom. Large value of $(SSE_R - SSE_C)$ or equivalently large value of F^* leads to the

rejection of the null hypothesis. Let α denote the level of significance. If a test with a type I error probability α is used, $F^* > F_\alpha$ is the appropriate rejection region.

3.2.3 Algorithm comparison

Figure 1.2 compares different dual camera mapping methods. The reference approaches [Cui98, Scotti05] require the knowledge of both camera's projection models and relative positions. Based on the projection model, camera calibration is conducted to estimate the camera's intrinsic and projection parameters. These estimated parameters along with the known relative position are used to generate the transformation functions defined in Equation (3.6). The calibration results heavily depend on the accuracy of the projection model.

Our geometry calibration method relaxes the requirement of known camera's projection model. A polynomial approximation with automated model selection mechanism is incorporated into camera calibration. However, the requirement of known relative position is still needed. The proposed geometry calibration method has the following distinguishing advantages compared with the reference approaches [Cui98, Scotti05]. (1) The projection model is selected automatically based on the AIC criterion. Compared with the empirical model used in [Cui98], our algorithm has a solid theoretical derivation. Moreover, since the polynomial degree is determined based on a statistical measure of the system performance, the risk of choosing an erroneous projection model is reduced. (2) The remaining camera calibration process resembles the well-known algorithm proposed by Zhang [Zhang04]. Compared with the algorithm discussed in [Cui98], our method features significantly reduced complexity and straightforward implementation. In practice, with the abundance of available calibration software developed based on Zhang's algorithm, our approach can be readily applied with only moderate modifications. (3) The derived spatial mapping is application independent and capable of accommodating a large variety of omnidirectional cameras, most of which can be described with satisfactory accuracy by a polynomial of a degree no greater than five [Yao06].

The homograhpy calibration method proceeds without the knowledge of either the camera's projection model or the relative position. Polynomial approximation is used to directly model the relation between (x_i, y_i) and $(\theta_{i,P}, \theta_{i,T}, f_i)$. The homograhpy calibration method has the highest flexibility in comparison with the reference and our geometry calibration methods. In addition, the detected targets may not be of the same height, which violates the planar motion assumption and results in errors in the reference and geometry calibration methods. Since this type of errors also exists in the training set, the homography calibration method minimizes the errors simultaneously.

4 Homogeneous mapping theory

Surveillance systems [Bagdanov06, Angella07] with multiple PTZ cameras became popular in the past decade, because of their capacity to simultaneously cover wide area and maintain high object resolution imagery. Due to the time-varying relations among PTZ cameras, how to coordinate multiple PTZ cameras by means of changing their poses to achieve a better observation of the object of interest remains challenging. Even though there is a vast amount of literature on automatically calibrating larger camera networks [Svoboda05, Sugimure04], those works mainly deal with stationary perspective cameras.

Thus, the works of Chen and Wang [Chen06A, Chen07B] and Everts et al. [Everts07] proposed to use known intrinsic parameters of PTZ cameras to direct their poses, namely pan, tilt, and zoom values, whenever a change is needed. In other words, we have to individually calibrate each PTZ camera [Li07, Agapito01] to obtain their intrinsic parameters beforehand. This impedes their direct application to automated surveillance systems with changing configurations and a larger number of PTZ cameras. In particular, due to errors in the estimation of intrinsic parameters of PTZ camera, the works of Chen and Wang [Chen06A, Chen07B] need one more optimization process, sensitivity analysis, to obtain the pose relation between PTZ cameras. This increases the system's computational complexity in the calibration process. To overcome their limitations, we propose a novel mapping approach that directly derives a unified polynomial model between the pan, tilt, and zoom values of PTZ cameras with unknown intrinsic parameters and setups in the scene.

The contribution of this work is to derive the relation of pan, tilt, and zoom values between any pair of PTZ cameras without prior knowledge of their intrinsic parameters and relative positions. In comparison with the reference algorithm [Chen06A, Chen07B], our proposed approach not only reduces the dependence on the knowledge of intrinsic parameters of PTZ camera, but improves the degree of autonomy and reduces the system's computational complexity at the cost of slightly decreased pixel accuracy. In general, this slightly decreased pixel accuracy does not affect the overall performance for the application of automated surveillance systems, as long as the desired object can be seen within the field of view and can be compensated by consistent labeling approaches [Lowe04] without added cost.

The remainder of this chapter is organized as follows. The problem definition is given in Section 4.1. Section 4.2 describes homegenous mapping methods.

4.1 Problem definition

The setup of a pair of PTZ cameras is shown in Figure 4.1. We choose the coordinate of the zero position of a selected camera as the reference world coordinate, where pan and tilt angles

are both set to 0. A point $P_i = (X_i, Y_i, Z_i)^T$ in the reference world coordinate is projected onto the j^{th} PTZ camera's image coordinate $(x_{ij}, y_{ij}, \lambda_{ij})$ by

$$\begin{bmatrix} x_{ij} \\ y_{ij} \\ \lambda_{ij} \end{bmatrix} = \begin{bmatrix} f_{zoom,j} & s_{zoom,j} & x_{zoom,j} \\ 0 & \alpha_{zoom,j} f_{zoom,j} & y_{zoom,j} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta_{T,j} & 0 & -\sin\theta_{T,j} \\ 0 & 1 & 0 \\ \sin\theta_{T,j} & 0 & \cos\theta_{T,j} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_{P,j} & \sin\theta_{P,j} \\ 0 & -\sin\theta_{P,j} & \cos\theta_{P,j} \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix}, \quad (4.1)$$

where $\theta_{P,j}$ and $\theta_{T,j}$ represent the pan and tilt angles of the j^{th} PTZ camera, respectively. $(x_{zoom,j}, y_{zoom,j})$ represents the principal point in the j^{th} PTZ camera. $f_{zoom,j}$ denotes the focal length of the j^{th}. $\alpha_{zoom,j}$ and $s_{zoom,j}$ respectively represent the aspect ratio and skew of the j^{th} PTZ camera.

In essence, $(x_{zoom,j}, y_{zoom,j})$, $f_{zoom,j}$, $\alpha_{zoom,j}$, and $s_{zoom,j}$ are subject to the changes of zoom value z_j of the j^{th} camera. The same point is projected onto $\mathbf{p}_{ih} = (x_c, y_c, 1)^T$, the center of the image coordinates of the h^{th} PTZ, by proper pan, tilt, and zoom values:

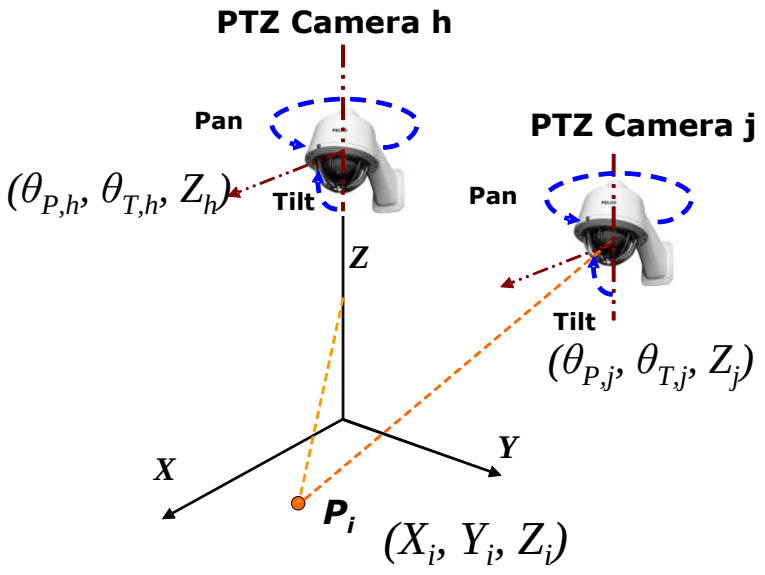


Figure 4. 1 Typical setup of a pair of PTZ cameras.

$$\begin{bmatrix} x_c \\ y_c \\ 1 \end{bmatrix} = \begin{bmatrix} f_{zoom,h} & s_{zoom,h} & x_{zoom,h} \\ 0 & \alpha_{zoom,h} f_{zoom,h} & y_{zoom,h} \\ 0 & 0 & 1 \end{bmatrix}$$
$$\begin{bmatrix} \cos\theta_{T,h} & 0 & -\sin\theta_{T,h} \\ 0 & 1 & 0 \\ \sin\theta_{T,h} & 0 & \cos\theta_{T,h} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_{P,h} & \sin\theta_{P,h} \\ 0 & -\sin\theta_{P,h} & \cos\theta_{P,h} \end{bmatrix} \left(\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} + t_{hj} \right), \tag{4.2}$$

where t_{hj} denotes the translation vector between the optical center of the h^{th} and j^{th} PTZ cameras.

Based on the point correspondences, two equations can be derived from Equation (4.3) so as to solve for $\hat{\theta}_{P,h}$, $\hat{\theta}_{T,h}$, and $\hat{Z}_h$. In essence, to avoid the needed knowledge of internal and external parameters of each PTZ camera in the scene, we propose to use a set of polynomials to directly relate $(x_{ih}, y_{ih}, \theta_{P,h}, \theta_{T,h}, Z_h)$ and $(x_{ij}, y_{ij}, \theta_{P,j}, \theta_{T,j}, Z_j)$ from a training set. The training set is collected from tracking the same object in two PTZ cameras where the centroid of the object stays at the image center of the h^{th} camera, but can be anywhere in the image of the j^{th} camera. This object in both images maintains a constant-sized pixel resolution for the future applications such as behavior understanding, face recognition, and so forth.

$$\begin{bmatrix} X_i \\ Y_i \\ Z_i \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_{P,h} & \sin\theta_{P,h} \\ 0 & -\sin\theta_{P,h} & \cos\theta_{P,h} \end{bmatrix} \begin{bmatrix} \cos\theta_{T,h} & 0 & -\sin\theta_{T,h} \\ 0 & 1 & 0 \\ \sin\theta_{T,h} & 0 & \cos\theta_{T,h} \end{bmatrix} \begin{bmatrix} \frac{1}{f_{zoom,h}} & \frac{-s_{zoom,h}}{f_{zoom,h}} & \frac{-x_{zoom,h}}{f_{zoom,h}} \\ 0 & \frac{1}{\alpha_{zoom,h} f_{zoom,h}} & \frac{y_{zoom,h}}{\alpha_{zoom,h} f_{zoom,h}} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ 1 \end{bmatrix} - t_{hj}$$
$$= \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_{P,j} & -\sin\theta_{P,j} \\ 0 & \sin\theta_{P,j} & \cos\theta_{P,j} \end{bmatrix} \begin{bmatrix} \cos\theta_{T,j} & 0 & \sin\theta_{T,j} \\ 0 & 1 & 0 \\ \sin\theta_{T,j} & 0 & \cos\theta_{T,j} \end{bmatrix} \begin{bmatrix} \frac{1}{f_{zoom,j}} & \frac{-s_{zoom,j}}{f_{zoom,j}} & \frac{-x_{zoom,j}}{f_{zoom,j}} \\ 0 & \frac{1}{\alpha_{zoom,j} f_{zoom,j}} & \frac{y_{zoom,j}}{\alpha_{zoom,j} f_{zoom,j}} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{ij} \\ y_{ij} \\ \lambda_{ij} \end{bmatrix} . \tag{4.3}$$

As a result, once Equation 4.4),

$$\begin{cases} \hat{\theta}_{P,h} = f_P(x_{ij}, y_{ij}, \theta_{P,j}, \theta_{T,j}, Z_j) \\ \hat{\theta}_{T,h} = f_T(x_{ij}, y_{ij}, \theta_{P,j}, \theta_{T,j}, Z_j) \\ \hat{Z}_h = f_Z(x_{ij}, y_{ij}, \theta_{P,j}, \theta_{T,j}, Z_j) \end{cases} \tag{4.4}$$

is derived, we can direct the h^{th} PTZ camera to the position where the i^{th} object is supposed to be placed at its image center with a desired pixel size, which is based on the pan, tilt, zoom values and the image coordinates of the i^{th} object in the j^{th} PTZ camera.

4.2 Cooperative mapping method

Our cooperative mapping methodology is inspired by the work of Chen et al [Chen08]. They pointed out that. Existing algorithms [Cui98, Scotti05] in the area of spatial mapping between the omnidirectional and PTZ cameras need to have prior knowledge of project models of cameras, namely internal and external parameters, and the environment geometry. This impedes their direct application to surveillance systems with changing configurations. This is similar to surveillance systems with multiple PTZ cameras. Thus, our proposed cooperative method can be divided into two phases, the data acquisition phase and the data fitting phase. Figure 4.2 illustrates the flow chart of these two phases. The purpose of data acquisition phase is to collect desired information to relate directly $(x_{ih}, y_{ih}, \theta_{P,h}, \theta_{T,h}, Z_h)$ and $(x_{ij}, y_{ij}, \theta_{P,j}, \theta_{T,j}, Z_j)$. The purpose of data fitting phase is to derive Equation (4.4) by the collected data set from data acquisition phase.

4.2.1 Data acquisition phase

At first, a single object moves around randomly in the overlapped field of views (FOVs) of the j^{th} and h^{th} PTZ cameras to collect its motion trajectory including $(x_{ih}, y_{ih}, \theta_{P,h}, \theta_{T,h}, Z_h)$ and $(x_{ij}, y_{ij}, \theta_{P,j}, \theta_{T,j}, Z_j)$. The centroid of the object stays at the image center of the h^{th} camera but can be anywhere in the image of the j^{th} camera. This object in both images maintains a constant-sized pixel resolution for the future applications such as behavior understanding, face recognition, and so forth. Since the focus of this paper is not developing a size preserving tracking approach, we utilize the algorithm proposed by Fayman at el. [Fayman01] in here. Once $(x_{ih}, y_{ih}, \theta_{P,h}, \theta_{T,h}, Z_h)$ and $(x_{ij}, y_{ij}, \theta_{P,j}, \theta_{T,j}, Z_j)$ are collected, we enter to data fitting phase to obtain Equation (4.4).

4.2.2 Data fitting phase

Since the derivations for pan, tilt, and zoom functions are similar, in the following discussion, we will take the pan angle, $\hat{\theta}_{P,h} = f_P(x_{ij}, y_{ij}, \theta_{P,j}, \theta_{T,j}, Z_j)$, as an example to save space. In general, we first fit a model with all possible predictor variables [Chen08] such as $\theta_{P,j}$, ..., $\theta_{P,j}^n$, $\theta_{T,j}$, ..., $\theta_{P,j}^n$, ... , Z_j, ..., Z_j^n, ... , x_{ij}, y_{ij}, x_{ij}^2, $Z_j x_{ij} y_{ij}$, y_{ij}^2, ..., x_{ij}^n, $x_{ij} y_{ij}^{n-1}$, ..., and $\theta_{T,j}^n y_{ij}^n$. Let w_i, with $i = 1,..k$, represent these k predictor variables. The pan angle in a complete model can then be expressed as

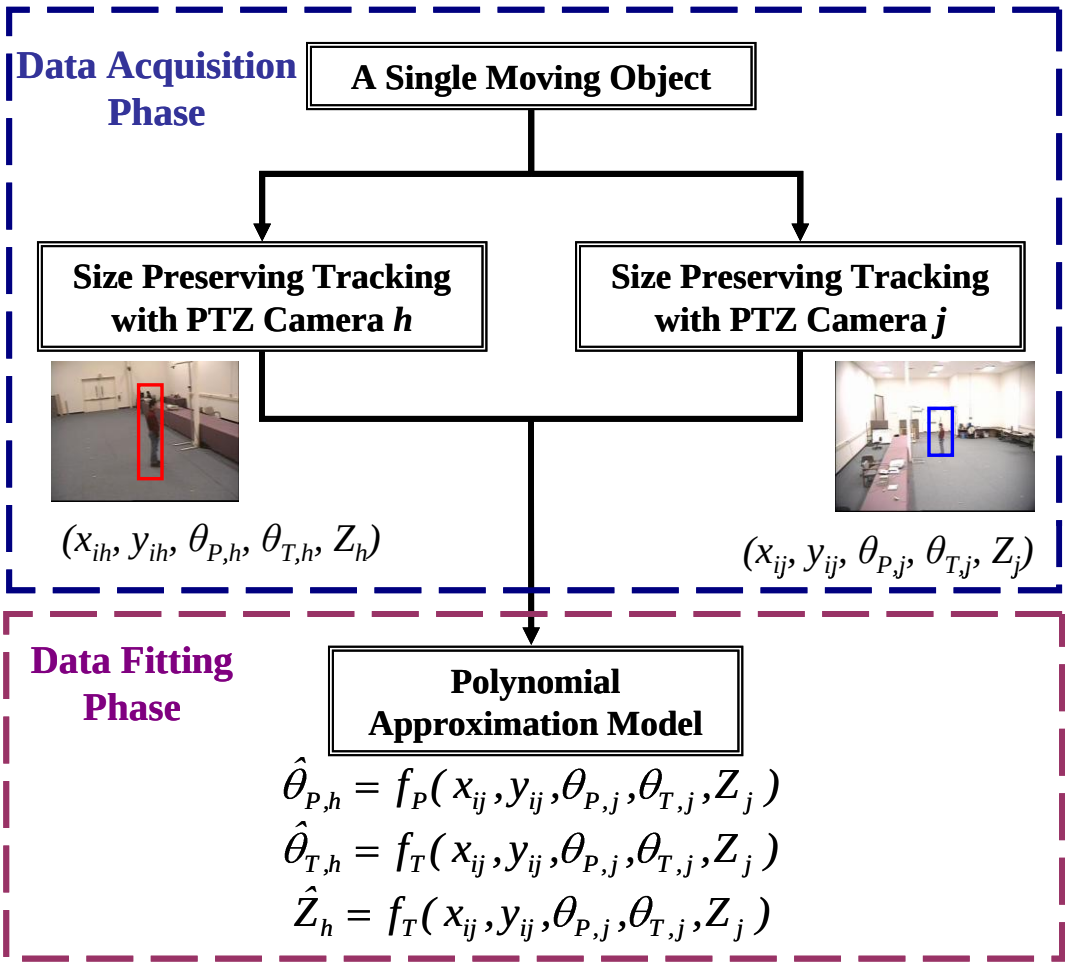


Figure 4.2 Illustration of our proposed cooperative mapping method.

$$\hat{\theta}_{P,h(C)} = \gamma_0 + \gamma_1 w_1 + \gamma_2 w_2 + \ldots + \gamma_k w_k + \varepsilon_C, \tag{4.5}$$

Where γ_i denotes the model fitting parameter and ε_C is a random error term with $E(\varepsilon_C) = 0$.

Usually not all predictor variables are equally significant. A subset of these variables can be found forming a reduced model:

$$\hat{\theta}_{P,h(R)} = \gamma_0 + \gamma_1 w_1 + \gamma_2 w_2 + \ldots + \gamma_g w_g + \varepsilon_R, \tag{4.6}$$

where $g < k$ and ε_R is a random error term with $E(\varepsilon_R) = 0$. Let SSE_C and SSE_R denote the sum of squared error of the complete and reduced models:

$$\begin{aligned} SSE_C &= \Theta_{P,C}^T \Theta_{P,C} - \Theta_{P,C}^T W_{P,C} (W_{P,C}^T W_{P,C})^{-1} W_{P,C}^T \Theta_{P,C}, \\ SSE_R &= \Theta_{P,R}^T \Theta_{P,R} - \Theta_{P,R}^T W_{P,R} (W_{P,R}^T W_{P,R})^{-1} W_{P,R}^T \Theta_{P,R} \quad , \end{aligned} \tag{4.7}$$

where $\Theta_{P,C}$ / $\Theta_{P,R}$ is the vector of all response variables in a complete/reduced model and $W_{P,C} / W_{P,R}$ is the vector of all predictor variables w_k / w_g in a complete/reduced model.

Intuitively, if w_1, w_2, …, and w_k are important information contributing variables, the complete model should have a smaller prediction error than the reduced model: $SSE_C \le SSE_R$. The greater the difference $(SSE_R - SSE_C)$ is, the stronger is the evidence to support the complete model that w_1, w_2, ..., w_k are significant information contributing terms and to reject the reduced model: $H_0 : \beta_{g+1} = \beta_{g+2} = \ldots = \beta_k = 0.$ Conversely, the acceptance of the reduced model suggests that the additional predictors in the complete model, $w_{g+1}, w_{g+2}, \ldots, w_k$, introduce no improvement to fitting accuracy. The predictors, w_1, $w_2, \ldots, w_g$ in the reduced model are sufficient and more significant information contributing terms than predictors, $w_{g+1}, w_{g+2}, \ldots, w_k$. In other words, this becomes a model selection problem. Thus, we use the recently proposed extension to Akaike's information criterion called information complexity (ICOMP) [Bozdogan00] as our fitness function, which has been proved more efficient than existing fitness functions such as F test used in [Chen08]. Other than its efficiency, another rationale for ICOMP as our fitness function is that it combines a badness-of-fit term with a measure of complexity of a model by taking into account the interdependencies of the parameter estimates, as well as the dependencies of the model residuals. This can increase the accuracy of estimation.

ICOMP is computed using Equations (4.8) and (4.9)

$$ICOMP = -2\log(L(\hat{w}_i)) + 2C_1(F^{-1}(\hat{w}_i)), \tag{4.8}$$

$$C_1(F^{-1}) = \frac{s}{2}\log\left[\frac{tr(F^{-1})}{s}\right] - \frac{1}{2}\log\left|F^{-1}\right|, \tag{4.9}$$

where $\hat{w}_i$ is the maximum likelihood estimator of w_i, L represents the likelihood function, F^{-1} is the inverse Fisher information matrix (IFIM), C_1 denotes the maximal information complexity of F^{-1}, s is the rank of F^{-1}, $|\,.\,|$ refers to the determinant, tr refers to the trace of the matrix. Equation (4.8) measures the lack of fit of the model, and Equation (4.9) measures the complexity of the estimated IFIM, which gives a scalar measure of the celebrated Cramér-Rao lower bound matrix. This takes into account the accuracy of the estimated parameters. The minimum value of ICOMP reveals the feature variable-subset is optimal in dimensionality and information content. More details behind the derivation of this formulation are available in [Bozdogan00]. In this paper, we use generic algorithm (GA) as a searching method along with the use of ICOMP criteria as the fitness function. How to use a GA-based procedure with informational complexity as the fitness function employed in this work is detailed in Bearse and Bozdogan [Bearse98].

5 Target hopping

The discrepancies in the FOV and resolution levels of the omnidirectional and PTZ cameras not only lead to difficulties in deriving an accurate spatial mapping, which is to be resolved by our geometry and homography calibration methods, but also bring in difficulties in multiple object tracking. Most existing algorithms are only able to handle cameras with similar FOV and resolution levels. Under this condition, the correspondences are of *1*-to-*1* mapping. In a dual camera system, the omnidirectional camera monitors and detects all latent targets in the environment while the PTZ camera only sees a small portion of the environment. Frequently, to achieve the required detail level, the PTZ camera can only monitor a single target at any given time. Therefore, there exists the problem of *N*-to-*1* mapping for a pair of dual cameras.

To upgrade the *N*-to-*1* mapping to a *1*-to-*1* mapping, one possible solution is to employ multiple PTZ cameras with the number of PTZ cameras N_c larger than or equal to the maximum number of targets, N_{max}, that may appear in the environment. However, this is usually not practical due to substantially increased cost and idle time for any individual PTZ camera. With limited number of PTZ cameras ($N_c<N_{max}$), an individual PTZ camera is responsible for observing multiple targets that may not fall into its FOV simultaneously. Therefore, a target-hopping scheme is exploited so that the PTZ camera cyclically visits multiple targets according to an automatically adjusted sequence. Each target is assigned a varying number of frames according to its priority. In so doing, all targets in the environment can be monitored with the required object resolution without necessitating more PTZ cameras. The remaining question, referred to as the next best target (NBT) problem, is how to generate an effective and dynamic visiting sequence for the PTZ camera.

The NBT problem is similar to the traveling salesman problem [Applegate06], but with two major differences. (1) The location of each tracked target changes frequently and (2) the computational and traveling time of generating and implementing the optimal visiting sequence affects the system performance significantly. For example, a difference of one or two seconds in the computational and traveling time is marginal in the traveling salesman problem. However, the same amount of time affects a surveillance system substantially to where the system may fail to appropriately detect abnormal events. In this dissertation, the NBT problem is solved with focus on the targets' dynamics and the system's computational complexity. An adaptive visiting scheme is developed, which alternates between the sequential visiting and nearest neighbor approaches depending on the distribution of the detected targets.

In a dual camera system, the omnidirectional camera detects multiple targets in the environment at low object resolution while the PTZ camera tracks one individual target at one time with high object resolution. It becomes necessary for the PTZ camera to distribute its observation time among multiple targets according to their priorities. The next best target (NBT)

problem arises and is addressed in this section. Figure 5.1 illustrates a scenario where multiple targets are detected by the omnidirectional camera and the PTZ camera selects a sequence to visit multiple targets one at a time.

As a similar study, the traveling salesman problem searches for the shortest route to visit a collection of cities [Applegate06]. The sequential visiting and nearest neighbor methods are the most commonly used solutions. In the sequential visiting method, the PTZ camera visits multiple targets according to a fixed sequence determined by the order serially assigned to each tracked target. When the nearest neighbor method is used, the PTZ camera moves to the next nearest target. The sequential visiting and nearest neighbor methods are effective for scenarios with clustered and scattered targets, respectively, in terms of computational complexity.

As mentioned before, the major difficulties in applying the solutions of traveling salesman problem to our NBT problem are the dynamics of the targets and constraints on computational and traveling time. To design our adaptive target-hopping method, we first examine the complexities of the sequential visiting and nearest neighbor methods. The following variables are defined. N_t is the number of tracked targets in the omnidirectional camera to be visited by the PTZ camera. T_m is the computational time for calculating the pan, tilt, zoom values. T_P, T_T, and T_R are the time needed for PTZ camera to move one degree of pan, tilt, and one unit of zoom respectively. The total computational and traveling time of the sequential visiting and nearest neighbor methods for the PTZ camera to finish a tour, TT_{SV} and TT_{NN}, is given by

$$TT_{SV} = N_t T_m + \sum_{i=1}^{N_t} (\hat{\theta}_{i,P}^{(SV)} T_P + \hat{\theta}_{i,T}^{(SV)} T_T + \hat{f}_i^{(SV)} T_Z) + \varepsilon_1, \tag{5.1}$$

$$TT_{NN} = N_t T_m + (N_t - 1)T_m + \ldots + 2T_m + T_m + \sum_{i=1}^{N_t} (\hat{\theta}_{i,P}^{(NN)} T_P + \hat{\theta}_{i,T}^{(NN)} T_T + \hat{f}_i^{(NN)} T_Z) + \varepsilon_2, \tag{5.2}$$

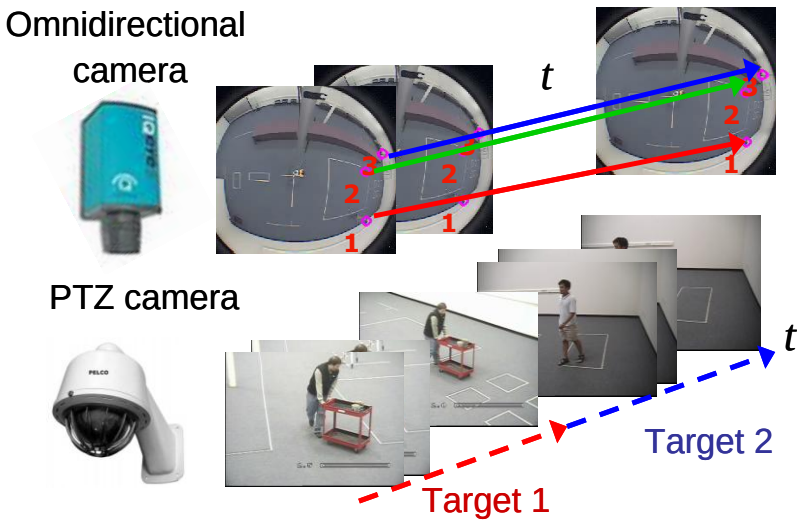


Figure 5. 1 Illustration of a scenario where multiple targets are detected in the omnidirectional camera and the PTZ camera selects the sequence to visit multiple targets cyclically.

where $\hat{\theta}_{i,P}^{(SV)}$, $\hat{\theta}_{i,T}^{(SV)}$, and $\hat{f}_i^{(SV)}$ respectively represent $\theta_{i,P}$, $\theta_{i,T}$, and f_i generated by the sequential visiting method and $\hat{\theta}_{i,P}^{(NN)}$, $\hat{\theta}_{i,T}^{(NN)}$, and $\hat{f}_i^{(NN)}$ respectively represent $\theta_{i,P}$, $\theta_{i,T}$, and f_i generated by the nearest neighbor method. ε_1 and ε_2 denote computational errors. Note that distance between any two objects can be assigned according to their priorities, instead of physical distance. In other words, if some objects (higher priorities) need to be watched more often than others, a shorter distance between it and any one can be assigned.

Given the complexities in Equation (5.1) and (5.2), we compare the sequential visiting and nearest neighbor methods to select the more suitable one with less computational and traveling time. We first hypothesize that the sequential visiting method requires less or equal amount of computational and traveling time $TT_{SV} \leq TT_{NN}$. This eliminates the need for an optimized solution such as the nearest neighbor method, since a random sequence can achieve similar or better performance. Plugging in Equation (5.1) and (5.2), we have

$$TT_{NN} - TT_{SV} = \left[\frac{(N_t+1)N_t T_m}{2} + \sum_{i=1}^{N_t} (\hat{\theta}_{i,P}^{(NN)} T_P + \hat{\theta}_{i,T}^{(NN)} T_T + \hat{f}_i^{(NN)} T_Z) + \varepsilon_2 \right] - \left[N_t T_m + \sum_{i=1}^{N_t} (\hat{\theta}_{i,P}^{(SV)} T_P + \hat{\theta}_{i,T}^{(SV)} T_T + \hat{f}_i^{(SV)} T_Z) + \varepsilon_1 \right] \geq 0 \quad , \tag{5.3}$$

$$\frac{(N_t-1)N_t T_m}{2} \geq \sum_{i=1}^{N_t} \left| (\hat{\theta}_{i,P}^{(SV)} T_p - \hat{\theta}_{i,P}^{(NN)} T_p) + (\hat{\theta}_{i,T}^{(SV)} T_t - \hat{\theta}_{i,T}^{(NN)} T_t) + (\hat{f}_i^{(SV)} T_Z - \hat{f}_i^{(NN)} T_p) \right| \quad , \tag{5.4}$$

where $\varepsilon_1 - \varepsilon_2 \cong 0$. To further simplify the derivation, we assume $\gamma = \frac{T_m}{\min\{T_P, T_T, T_Z\}}$ and arrive at

$$\frac{(N_t-1)N_t \gamma}{2} \geq \sum_{i=1}^{N_t} \left| (\hat{\theta}_{i,P}^{(SV)} - \hat{\theta}_{i,P}^{(NN)}) + (\hat{\theta}_{i,T}^{(SV)} - \hat{\theta}_{i,T}^{(NN)}) + (\hat{f}_i^{(SV)} - \hat{f}_i^{(NN)}) \right| \quad . \tag{5.5}$$

In Equation (5.5), we can see that when multiple targets are clustered in a small are, it is highly probable that $TT_{SV} \leq TT_{NN}$ while it is highly probable that $TT_{SV} > TT_{NN}$ when multiple targets are scatted in the environment. Based on this observation, we propose an adaptive method which alternates between the sequential visiting and nearest neighbor methods to save computational and traveling time. For the alternating scheme to work properly, a threshold is derived to govern the transition. Let $\hat{\theta}_{\max,P} = \max_i \left| \hat{\theta}_{i,P}^{(SV)} - \hat{\theta}_{i,P}^{(NN)} \right|$, $\hat{\theta}_{\max,T} = \max_i \left| \hat{\theta}_{i,T}^{(SV)} - \hat{\theta}_{i,T}^{(NN)} \right|$, and $\hat{f}_{\max} = \max_i \left| \hat{f}_i^{(SV)} - \hat{f}_i^{(NN)} \right|$, we have

$$\sum_{i=1}^{N_t} \left| (\hat{\theta}_{i,P}^{(SV)} - \hat{\theta}_{i,P}^{(NN)}) + (\hat{\theta}_{i,T}^{(SV)} - \hat{\theta}_{i,T}^{(NN)}) + (\hat{f}_i^{(SV)} - \hat{f}_i^{(NN)}) \right| \geq \hat{\theta}_{\max,P} + \hat{\theta}_{\max,T} + \hat{f}_{\max} \tag{5.6}$$

And finally the threshold is given by:

$$\frac{(N_t - 1)N_t\gamma}{2} \geq \hat{\theta}_{\max,P} + \hat{\theta}_{\max,T} + \hat{f}_{\max}. \tag{5.7}$$

The adaptive method, therefore, can proceed as follows: if Equation (5.7) holds, the sequential visiting method is selected; otherwise, the nearest neighbor method is used.

6 Camera handoff

To set up an automated surveillance system using multiple omnidirectional cameras, we encounter the same issues as systems based on multiple perspective cameras, including camera placement, camera handoff, and object tracking. Camera placement, as the first step to set up a surveillance system, determines the cameras' configuration including intrinsic and extrinsic parameters according to the geometry of the environment and the requirements of the system performance. Camera handoff, as the dynamic online coordination center, determines when and which camera to track and monitor the object of interest. Consistent labeling, as an important step in camera handoff, builds the connections of the same object in different camera's FOVs. Object tracking, as the fundamental online tracking function in a single camera, lays the foundation to keep the track of the object of interest and the understanding of their behaviors for an advanced application. With the proper functioning of these units, an automated surveillance system based on multiple omnidirectional cameras is able to fulfill tasks such as activity monitoring, behavior understanding, and threat awareness. Table 6.1 summarizes and compares the functions of these units. The difficulties caused by the use of omnidirectional cameras are listed as well. Since the focus of this dissertation is not developing a multi object tracking algorithm in the omnidirectional camera, we assume that reasonably correct object tracking result is available throughout whatever method is preferred by the user. In deed, we use the work of Cui et al. [Cui98] for target detection and tracking in an omnidirectional camera.

In general, Camera handoff regulates the collaboration among multiple cameras and answers the questions of When and Who: when a handoff request should be triggered to secure sufficient time for a successful consistent labeling and who is the most qualified camera to take over the

Table 6.1 Comparison of the functions of camera placement, camera handoff, and object tracking in a surveillance system

Unit	Range	Execution	Difficulties
Camera placement	Multiple cameras	Offline	Various projection models
Camera handoff	Adjacent cameras	Online	Nonuinform resolution Distorted appearance Various projection models
Object tracking	Single camera	Online	Nonuinform resolution Distorted appearance

object of interest before it falls out of the FOV of currently observing camera. In order works,camera handoff should comprise three fundamental components, time to trigger handoff process, the execution of consistent labeling, and the selection of the next camera. Nevertheless, most existing camera handoff algorithms discussed in systems based on multiple perspective cameras concentrate on the execution of consistent labeling and ignore the interrelation among those three fundamental components. As a result, there is no clear formulation to govern the transition between adjacent cameras. In addition, a complete camera handoff process including abovementioned three fundamental components needs a certain amount of time to be executed successfully, especially the time needed to execute consistent labeling. Thus, the size of overlapped FOV should be reserved enough for successfully carrying out consistent labeling, before the object falls out of the FOV of the observing camera.

In this chapter, we first present a novel solution to the consistent labeling problem in omnidirectional cameras. A spatial mapping procedure considering both the noise inherent to the tracking algorithms used by the system and the lens distortion introduced by omnidirectional cameras is proposed to automatically obtain the correspondences between the trajectories of the same object seen in different omnidirectional cameras without human interventions. We also propose to use the Wilcoxon Signed-Rank Test for trajectory associationsince. Afterwards, we propose a camera placement approach finding an optimized tradeoff between the overall coverage and the size of overlapped FOVs to maximize the performance of the automated surveillance system in terms of the continuity of the tracked object.

The remainder of this chapter is organized as follows. The camera handoff approach including consistent labeling method and observation measure is discussed in Section 6.1. Section 6.2 describes the camera placement approach.

6.1 Camera handoff system architecture

The flow chart of the proposed camera handoff algorithm is depicted in Figure 6.1, where operations are carried out at the handoff request and handoff response ends. Let the j^{th} camera be the handoff request end end and the i^{th} object be the one that needs a transfer. A handoff request of the i^{th} object is triggered and broadcasted by the the j^{th} camera to adjacent cameras if $S_{ij} \le S_T$ where S_{ij} is the observation measure of the i^{th} object in the j^{th} camera and S_T represents the trigger threshold. The trigger threshold is determined by the average moving speed of the object of interest and time needed to execute camera handoff successfully, which is given by

$$S_T = \mu V_m T_H , \tag{6.1}$$

Where S_T represents the trigger threshold, which is determined by the average moving speed of the object of interest and time needed to execute consistent labeling successfully. V_m represents the average moving speed of the object of interest, T_H denotes the average duration for a successful consistent labeling, and μ is a conversion scalar between the value of observation measure and world space. Afterwards, the j^{th} camera keeps tracking the i^{th} object and waiting for positive responses from adjacent cameras while it is still visible. In doing so, the necessary time

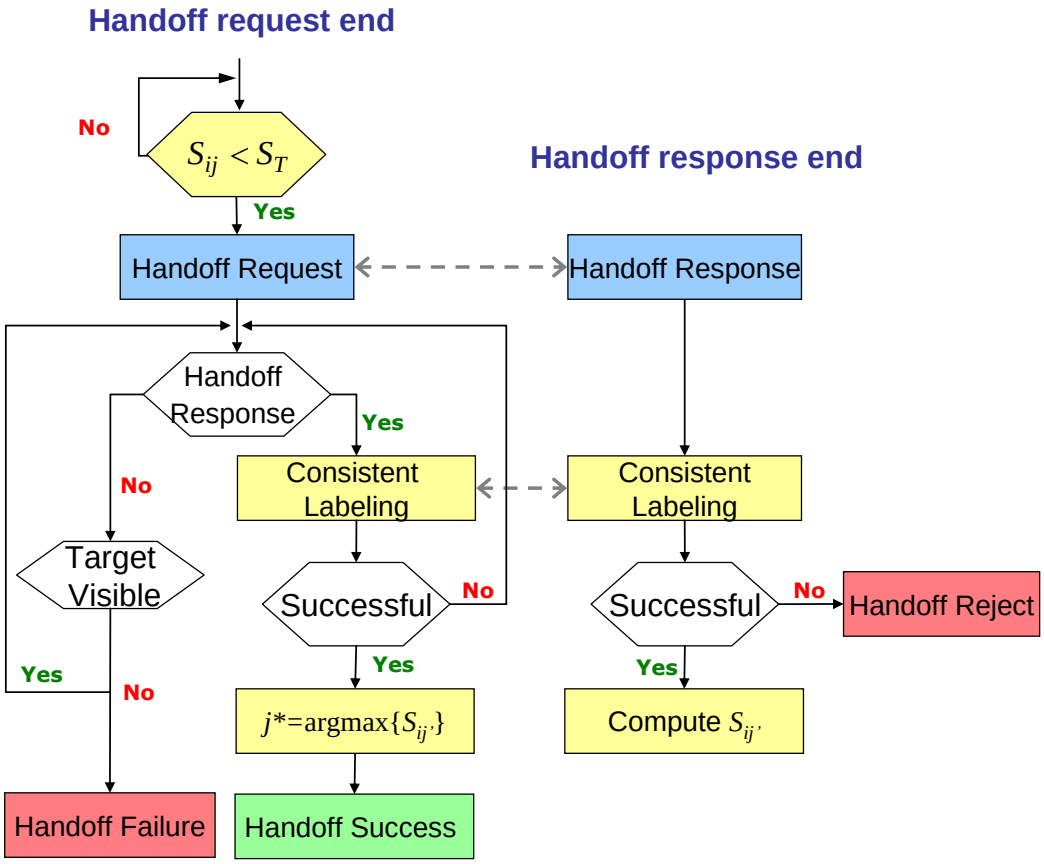


Figure 6. 1 Flow chart of the proposed camera handoff algorithm, where operations are carried out at the handoff request and handoff response ends**.**

margin for executing consistent labeling is converted to the thresholds that trigger camera handoff. Let the $(j')^{th}$ camera be the handoff response end. Once a positive response is received, the j^{th} and $(j')^{th}$ cameras perform consistent labeling to identify the i^{th} object. If the association of the i^{th} object is established successfully, the $(j')^{th}$ camera becomes a valid candidate. Otherwise, the handoff request is rejected. Figure 6.2 illustrates the concept of S_T. Figures 6.2(a) and (b) demonstrate the scenarios with $S_{ij} > S_T$, where the object of interest remains in the field of view of the observing camera and presents an acceptable object resolution. Figure 6.2(c) demonstrates scenario with $S_{ij} \le S_T$, where camera handoff is necessary due to the rapidly decreased object resolution, although the object of interest stays is still in the field of view of the observing camera. Back to the handoff request end, among all valid candidate cameras the $(j^*)^{th}$ camera with the highest observation measure $j^* = \arg\max_{j'}\{S_{ij'}\}$ is selected as the most appropriate camera to take over the i^{th} object in the pool of candidate cameras. If no positive response is received, the j^{th} camera continues tracking the i^{th} object and broadcasting the handoff request to adjacent cameras until the target falls out of its FOV or a positive handoff response is granted. A handoff failure is finally issued when the target becomes untraceable.

6.1.1 Observation measure

To maintain persistent and continuous object tracking, a handoff request is triggered before the object of interest is untraceable or unidentifiable. The object of interest may become untraceable or unidentifiable due to the following reasons: (1) the object is leaving the camera's FOV; and (1) the object's resolution is getting low. Accordingly, two criteria are defined in the observation measure to determine when to trigger a handoff request: resolution S_r and distance to the edge of the camera's FOV S_d. Both S_r and S_d are scaled to [0, 1] where zero means that the object is untraceable or unidentifiable and one means that the camera has the best effectiveness in tracking the object. The definition and derivation of observation measure are originally from the work of Yao [Yao08B]. In this dissertation, we use it as a quantified metric to direct camera handoff for continuous and automated tracking before the tracked object falls out of the FOV of currently observing camera.

The geometry of an omnidirectional camera is depicted in Figure 6.3. Given a point P(X, Y, Z), the pan θ_P and tilt θ_T angles are $\theta_P = \tan^{-1}(X/Y)$ and $\theta_T = \tan^{-1}(R/Z)$, respectively, with $R = \sqrt{X^2 + Y^2}$. The imaging process of an omnidirectional camera does not comply with the traditional perspective projection. Let r denote the distance between the projected point $\mathbf{p}(x, y)$ and the principle point and θ the angle between the incoming ray and the optical axis. The perspective projection is characterized by $r = f\tan\theta$. To realize a wider opening angle, this relation is changed. Various projection models exist in literature [Kannala04], such as the equidistance projection $r = f\theta$, the general polynomial model $r = f\sum_{k=1,odd}^{K}\lambda_k\theta^k$ where λ_k denote the approximation coefficients.

(a) (b) (c)

Figure 6. 2 Illustration of S_T. (a) and (b) demonstrate the scenarios with $S_{ij} > S_T$, where the object of interest remains in the field of view of the observing camera and presents an acceptable object resolution. (c) demonstrates scenario with $S_{ij} \leq S_T$, where camera handoff is necessary due to the rapidly decreased object resolution although the object of interest stays is still in the field of view of the observing camera.

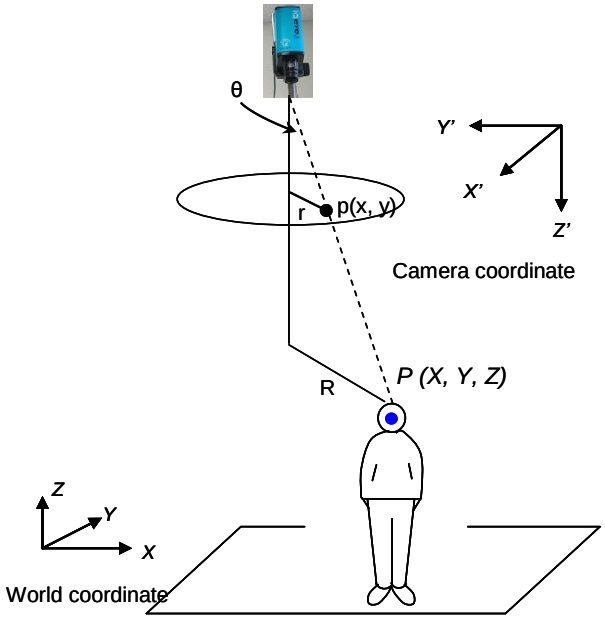


Figure 6. 3 Illustration of the geometry of an omnidirectional camera. In particular, The imaging process of an omnidirectional camera does not comply with the traditional perspective projection.

The use of a polynomial model provides us the flexibility of modeling and unifying omnidirectional cameras with various projection models, which is important for the integration of multiple omnidirectional cameras. For the purpose of camera handoff and camera placement, we need a unified basis on which omnidirectional cameras with various projection models can communicate. However, the difficulty lies in the selection of an appropriate polynomial power to avoid both over-estimated and under-estimated problems. To meet the above requirement, we used a statistical metric, the Akiake's information criterion (AIC) [Akaike74], to select the optimal polynomial power [Yao06].

Statistical model selection is used to optimize the model parameters when several competing models can be used to explain an observation. In our applications, model selection optimizes the polynomial degree, *K*. The AIC criterion, $-2\log L(\omega;\mathbf{p}_i)+2N$, is used by the work of Akiake [Akaike74], where $L(\omega;\mathbf{p}_i)$ is the likelihood of the model parameters ω (including a total of *N* camera intrinsic and extrinsic parameters). Assuming a Gaussian distribution of $\mathbf{p}_i$:

$$\Pr(\mathbf{p}_i \mid \omega) = \prod_i \left(\frac{1}{\sqrt{2\pi}\sigma}\right) \exp\left(-e_i^2 / (2\sigma^2)\right), \tag{6.2}$$

where $e_i = \|\mathbf{p}_i - \breve{\mathbf{p}}_i\|$ and $\breve{\mathbf{p}}_i$ is the estimated projection based on ω, we have

$$\log L(\omega;\mathbf{p}_i) = \arg\max_{\omega} \log \Pr(\mathbf{p}_i \mid \omega) \propto -\frac{1}{2\sigma^2}\sum_i e_i^2, \tag{6.3}$$

and the AIC criterion reduces to $\frac{1}{\sigma^2}\sum_i e_i^2 + 2N$.

The model selection algorithm proceeds as follows:

(1) Increase *K*.

(2) Perform camera calibration and obtain ω.

(3) Compute the corresponding model selection measure AIC.

(4) If the measure keeps decreasing, go to step (1). Otherwise, stop and output camera calibration results.

As the polynomial degree K increases, the corresponding model varies from an under-fitting to an over-fitting one. As a result, the AIC criterion decreases and then increases. Since our model selection algorithm starts from a smaller *K* (usually the initial *K* is set to 1), it is sufficient to stop the process once the AIC criterion begins to increase and assume that the model obtained in the previous iteration is the optimal solution. A detailed discussion regarding the performance of the aforementioned polynomial approximation can be found in [Yao06, Orekhov07], where an accuracy of 94.8% is reported. Another concern regarding the model selection algorithm that may arise is attributed to the assumption of Gaussian distribution. An extension of the AIC criterion, information complexity (ICOM) is proposed [Bozdogan00], where the assumption of Gaussian distribution is relaxed. However, in our experiments, the use of ICOM does not introduce noticeable performance improvement while the computational complexity increases

significantly. Therefore, considering both accuracy and computational cost, the AIC with the assumption of Gaussian distribution is exploited

Based on the polynomial approximation with automated model selection, we are able to define our observation measure for omnidirectional cameras. The image resolution of the i^{th} object in the j^{th} camera is actually the partial derivative of r with respective to R:

$$S_{r,ij} = \alpha \frac{\partial r}{\partial R} = \frac{\alpha Z}{Z^2 + R^2} \sum_{k=1,odd} \lambda_k k\theta^{k-1}, \tag{6.4}$$

where α is a normalization coefficient.

The distance to EFOV of currently observing camera for the i^{th} object in the j^{th} camera is given by

$$S_{d,ij} = \beta(-r / r_o)^2, \tag{6.5}$$

where r_o represents the image size of the omnidirectional camera and β is a normalization coefficient. The observation measure is given by:

$$S_{ij} = w_r S_{r,ij} + w_d S_{d,ij}, \tag{6.6}$$

where w_d, and w_r are importance weights. Their sum is one.

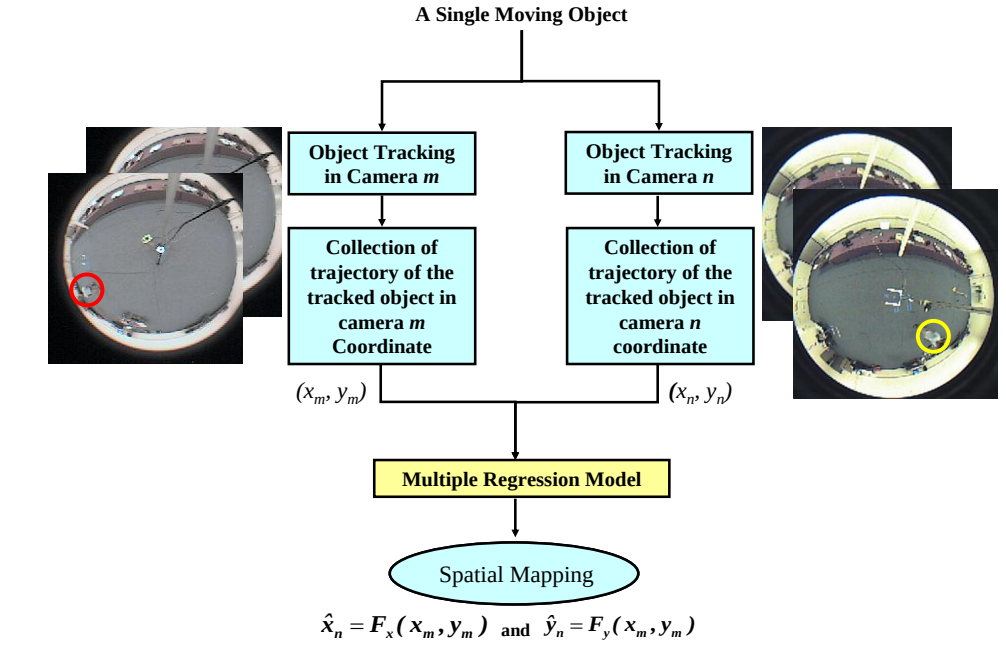
6.1.2 Consistent labeling

Our consistent labeling algorithm can be divided into two phases, the spatial mapping phase and the pair matching phase. Figure 6.4 illustrates the flow chart of these two phases. In Figure 6.4, the purpose of the spatial mapping phase is to automatically obtain the homography functions,

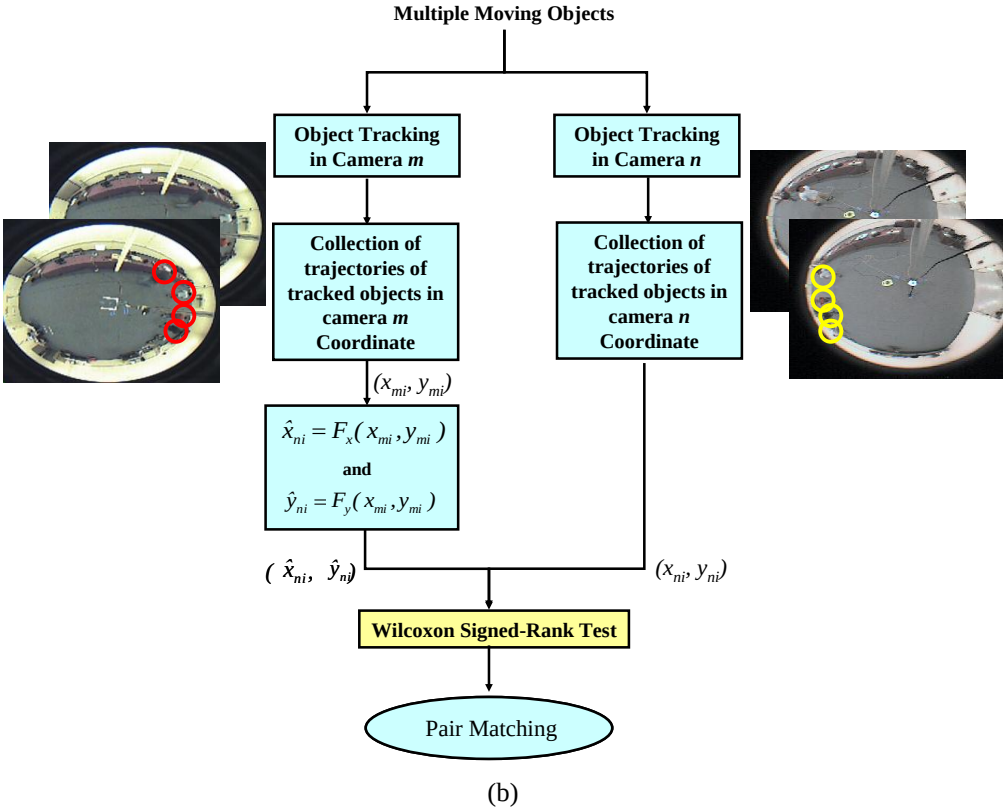
$$\hat{x}_n = F_x(x_m, y_m) \text{ and } \hat{y}_n = F_y(x_m, y_m), \tag{6.7}$$

between any two omnidirectional cameras with a joint view, where (x_m, y_m) represents the image coordinates of a single object seen in the m^{th} camera. ($\hat{x}_n$, $\hat{y}_n$) represent the estimated image coordinates of the same single object in the n^{th} camera. Once the homography functions are obtained, they are deployed in the system until the configuration of the system is changed. The purpose of the pair matching phase is to utilize the derived homography functions to mach any pair of objects in the n^{th} camera such as (x_{ni}, y_{ni}) and ($\hat{x}_{ni}$, $\hat{y}_{ni}$) where (x_{ni}, y_{ni}) represents the image coordinates of the i^{th} object seen in the n^{th} camera and ($\hat{x}_{ni}$, $\hat{y}_{ni}$) represents the estimated image coordinates of the i^{th} object in the n^{th} camera, which is derived from the m^{th} camera by the homography functions shown in Equation (6.7). In essence, the Wilcoxon Signed-Rank Test is incorporated into the pair matching phase to increase the accuracy of matching pairs of objects.

Our spatial mapping method proceeds without the knowledge of either the camera's projection model or their relative position. Polynomial approximation is used to directly model



(a)



(b)

Figure 6. 4 Illustration of our consistent labeling algorithm including the spatial mapping phase and the pair matching phase. (a) The spatial mapping phase. (b) The pair matching phase.

the relation between (x_m, y_m) and (x_n, y_n), and then obtain Equation (6.7). On the other hand, our calibration approach does not only have high flexibility and autonomy, as compared to the geometry-based approach based on the camera calibration, but also add the noise inherent to the tracking algorithms used by the system and the lens distortion introduced by omnidirectional cameras. Because of the imperfection inherent in the fitting model used by consistent labeling methods in all categories, the least square error or similar approaches is inefficient in matching pairs of objects, which is detailed in the following Section. According to our experiments, the Wilcoxon Signed-Rank Test, which studies if any pairs have the same distribution versus the alternative that distributions differ in location, is proved robust and efficient.

6.1.2.1 Spatial mapping phase

Our spatial mapping method proceeds without the prior knowledge of either the camera's projection model or their relative positions. Polynomial approximation is used to directly model the relation between (x_m, y_m) and (x_n, y_n). On the other hand, our spatial mapping approach does not only have high flexibility and autonomy, as compared to the conventional geometry-based approach requiring fully calibrated cameras, but also add the noise inherent to the tracking algorithms used by the system and the lens distortion introduced by omnidirectional cameras. Since the imperfection inherent in the fitting model used by consistent labeling methods in all categories, the least square error or similar approaches is inefficient on matching pairs of objects. According to our experiments, the Wilcoxon Signed-Rank Test, which studies if any pairs have the same distribution versus the alternative that distributions differ in location, is proved robust

At first, a single object moves around randomly in the overlapped FOVs of the m^{th} and n^{th} omnidirectional camera to collect its motion trajectory tracked by the two cameras, (x_m, y_m) and (x_n, y_n). Since the focus of this paper is not developing a multi object tracking approaches, we simply utilize the algorithm discussed by Cui at el. [Cui98]. Once (x_m, y_m) and (x_n, y_n) are collected, we want to find a suitable fitting method to derive the homography functions. Thus we first study the correlation between (x_m, y_m) and (x_n, y_n). Table 6.2 shows the mean correlation values averaged across a variety of omnidirectional camera system setups where omnidirectional cameras are placed overhead with various relative distances and heights. Rotations are not considered since an omnidirectional has a 360^0 FOV. This configuration is commonly used in most surveillance systems. Figure 6.5 shows one scatter plot matrix between (x_m, y_m) and (x_n, y_n) for one system setup.

In Table 6.2, we can see that the correspondences between (x_m, y_m) and (x_n, y_n) are highly correlated, which reaches up to 69% between x_m and x_n, and 71% between y_m and y_n. In Figure 6.5, taking the plot of x_m in the first row against the plot of x_n in the third column as an example, we can see a strong linear tendency between x_m and x_n. Alternatively, by viewing the fourth column y_n against y_m in the second row, we observe a linear tendency between y_m and y_n. According to the linear tendency shown in Table 6.2 and the scatter plot in Figure 6.5, the multiple regression model [Wackerly02, Neter04] appears to be a good candidate considering both accuracy and computational complexity, in comparison with complicated non-linear fitting algorithms such as the Levenberg-Marquardt algorithm.

Table 6.2 Averaged correlation between (x_m, y_m) and (x_n, y_n)

	x_m	y_m
x_n	0.6961	0.1027
y_n	-0.1374	0.7126

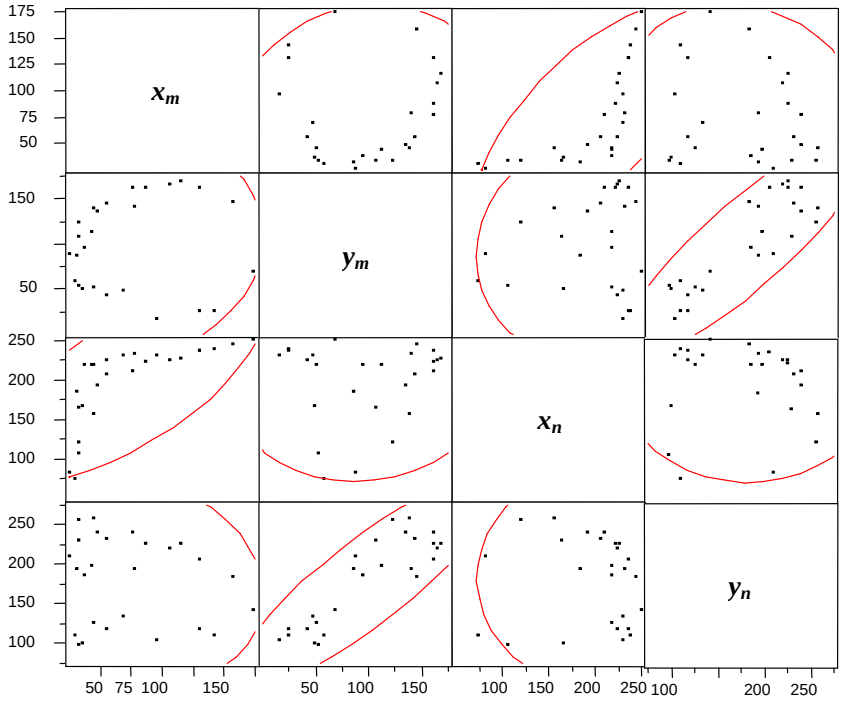


Figure 6.5 Illustration of one set of the scatter plot matrix between (x_m, y_m) and (x_n, y_n). The correspondences between (x_m, y_m) and (x_n, y_n) are highly correlated, which validates the use of the multiple regression model.

6.1.2.2 Pair matching phase

The least square error or similar approaches have been widely used to match pairs of objects in the geometry-based or homography-based approach. Figure 6.6 illustrates the problem of using the least square error or similar approaches in the homography-based approach. In Figure 6.6, $\mathbf{p}_{1m}$ and $\mathbf{p}_{1n}$ denote the pixel locations of object 1 in the m^{th} and n^{th} camera respectively. $\mathbf{p}_{2m}$ and $\mathbf{p}_{2n}$ denote the pixel locations of object 2 in the m^{th} and n^{th} camera respectively. $\hat{\mathbf{p}}_{1n}$ and $\hat{\mathbf{p}}_{2n}$ respectively denotes the estimated pixel locations of object 1 and 2 in the m^{th} camera, which are respectively derived from $\mathbf{p}_{1m}$ and $\mathbf{p}_{2m}$ by correspondence functions shown in Equation (6.7). Both estimated pixel locations, $\hat{\mathbf{p}}_{1n}$ and $\hat{\mathbf{p}}_{2n}$, suffer degradation caused by image noise and distortions, while the precision of calibration method cannot be guaranteed. As a result, the distances between $\mathbf{p}_{1n}$ and $\hat{\mathbf{p}}_{1n}$, and $\mathbf{p}_{1n}$ and $\hat{\mathbf{p}}_{2n}$ ($\mathbf{p}_{2n}$ and $\hat{\mathbf{p}}_{1n}$, and $\mathbf{p}_{2n}$ and $\hat{\mathbf{p}}_{2n}$) are the same based on the least square error method, therefore, the least square error or similar approaches unable to match the pairs appropriately.

In order to overcome the problem that least square error or similar approaches face, the Wilcoxon Signed-Rank Test is used to solve the problem [Wackerly02], because it can test if any pairs have the same distribution versus the alternative that distributions differ in location. In addition, since the distribution of each pair is unknown, a nonparametric statistical approach should be used in this case, instead of the parametric approach (such as the small-sample hypothesis testing based on a normal distribution presumption [Wackerly02]). To carry out the Wilcoxon Signed-Rank Test, we calculate the differences for each pair in the pool of collected objects' motion trajectories. Then we rank the absolute values of the differences, assigning a 1 to the smallest, a 2 to the second smallest, and so on. If two or more absolute differences are tied for the same rank, the average of the ranks that would be assigned to these differences is assigned to each member of the tied group. We use T [Wackerly02] as a test statistic to test the null hypothesis that the two population relative frequency histograms are identical. The smaller the value of T, the greater will be the evidence favoring the rejection of null hypothesis. Hence we will reject the null hypothesis if T is less than or equal to value, T_0, based on the assigned significance level, α. Since each object in our case has 2D position in the image coordinates, positions along x-axis and y-axis are respectively calculated and tested twice, only the two of that show the same evidence will be paired.

For clear presentation, an example of how the Wilcoxon Signed-Rand Test is incorporated into the pair matching phase is illustrated. Since the calculation for positions along x-axis and y-axis are similar, only the computation along the x-axis is presented to save space. Test the hypothesis that there is no difference in population distributions of positions along the x-axis for a matched pairs experiment involving six acquired positions over six frames, one for object A and the other for object B in each pair in images with 320×240 resolutions. Table 6.3 illustrates paired data and the calculation for the Wilcoxon Signed-Rank Test.

The null hypothesis to be tested is that the two population distributions of positions along x-axis are identical. The alternative hypothesis is that the distributions differ in location. We conduct our test with $\alpha = 0.1$. According to the work of Wackerly et al. [Wackerly02], T_0 is equal to 2. Hence, the null hypothesis is rejected if $T \leq 2$. Because there is only positive difference that has rank 3, $T^+ = 3$ and $T^- = 18$ (5 + 1.5 + 4 + 1.5 + 6 = 18), we have $T = 3$ by the rule of $T = min\ (T^+, T^-)$. Since the observed value of T exceeds T_0, there is not sufficient evidence to indicate a difference in the two population distributions of positions along the x-axis.

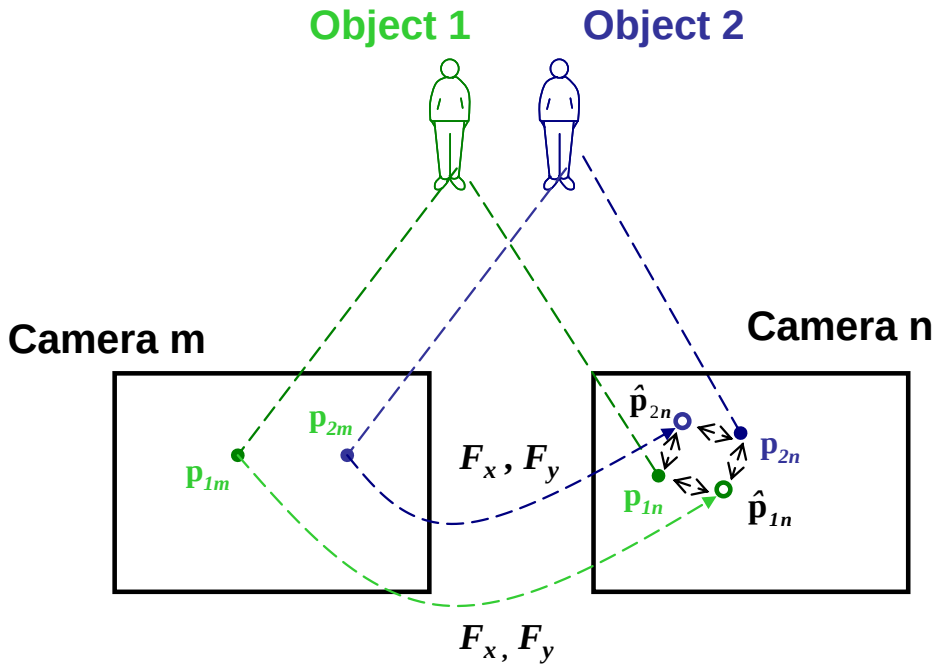


Figure 6. 6 Illustration of the problem caused by the least square error or similar approaches for matching pairs. Because the distances between p_{1n} and $\hat{p}_{1n}$, and p_{1n} and $\hat{p}_{2n}$ (p_{2n} and $\hat{p}_{1n}$, and p_{2n} and $\hat{p}_{2n}$) are similar due to image noise, the least square error or similar approaches cannot match the pairs appropriately.

Table 6.3 Paired data and the calculation for the Wilcoxon Signed-Rank Test

Object A	***Object B***	***Difference (A-B)***	***Absolute Difference***	***Rank of Absolute Difference***
135	129	6	6	3
102	120	-18	18	5
108	112	-4	4	1.5
141	152	-11	11	4
131	135	-4	4	1.5
144	163	-19	19	6

If the same result with the two population distributions of positions along the y-axis, we could claim that these two objects, A and B, are the same and consistent labeling is established

6.2 Determination of size of overlapped views

As we discussed before, a complete camera handoff process including abovementioned three components needs a certain amount of time to be executed successfully, especially the time needed to execute consistent labeling. This verifies the need of sufficient overlapped FOVs. Most existing camera placement algorithms [Erdem06, Mittal04] do not provide a solution to optimally determine the size of overlapped FOVs for carrying out consistent labeling successfully.

Assume that a polygonal floor plan is represented as an occupancy grid. Let A_1 represent the grid coverage with $a_{1,ij}=1$ if $S_{ij} > S_F$ and $a_{1,ij}=0$ otherwise. Two additional matrices are constructed A_2 and A_3. The matrix A_2 has $a_{2,ij}=1$ if $S_F < S_{ij} \le S_T$ and $a_{2,ij}=0$ otherwise, where S_F and S_T denote the failure and triggering threshold. S_F is the failure threshold that can simply be interpreted as invisible areas. In so doing, the necessary time margin for executing camera handoff is converted to the thresholds that trigger camera handoff. The matrix A_3 has $a_{3,ij}=1$ if $S_{ij} \ge S_T$ and $a_{3,ij}=0$ otherwise. Matrices A_2 and A_3 represent the handoff safety margin and visible area, respectively. Let $\mathbf{c}'_k = A_k x$, $k=1,2,3$. The solution vector $\mathbf{x}$ specifies a set of chosen camera configurations with the corresponding element x_j=1 if the configuration is chosen and x_j=0 otherwise. The objective function is formulated as:

$$c_i = w_1(c'_{1,i}>0) + w_2(c'_{2,i}=2) - w_3(c'_{3,i}>1), \tag{6.8}$$

where w_1, w_2, and w_3 are predefined importance weights. The operation $(c'_{1,i}>0)$ means $(c'_{1,i}>0)=\begin{cases}1 & c'_{1,i}>0\\ 0 & otherwize\end{cases}$. The first term in the objective function considers coverage, the second term produces sufficient overlapped handoff safety margins, and the third term penalizes excessive overlapped visible areas. Let the cost associated with the j^{th} camera configuration be ω_j. Given the maximum cost $C_{\max}$, the Max-Coverage problem can be described by:

$$\max \sum_i c_i, \text{ subject to } \sum_j \omega_j x_j \le C_{\max} \quad . \tag{6.9}$$

Given a specified coverage vector $\mathbf{b}_{C,o}$ or a minimum overall coverage $C_{\min}$, the Min-Cost problem can be modeled as:

$$\min \sum_j \omega_j x_j \text{ then } \max \sum_i c_i, \text{ subject to } A_1\mathbf{x} \ge \mathbf{b}_{C,o} \text{ or } \sum_i b_i \ge C_{\min}. \tag{6.10}$$

Since different consistent labeling approaches need different amount of time to execute consistent labeling successfully, the optimal size of overlapped FOVs O_{FOV}, can be formulated as follows:

$$O_{FOV} = \mu(S_T - S_F), \tag{6.11}$$

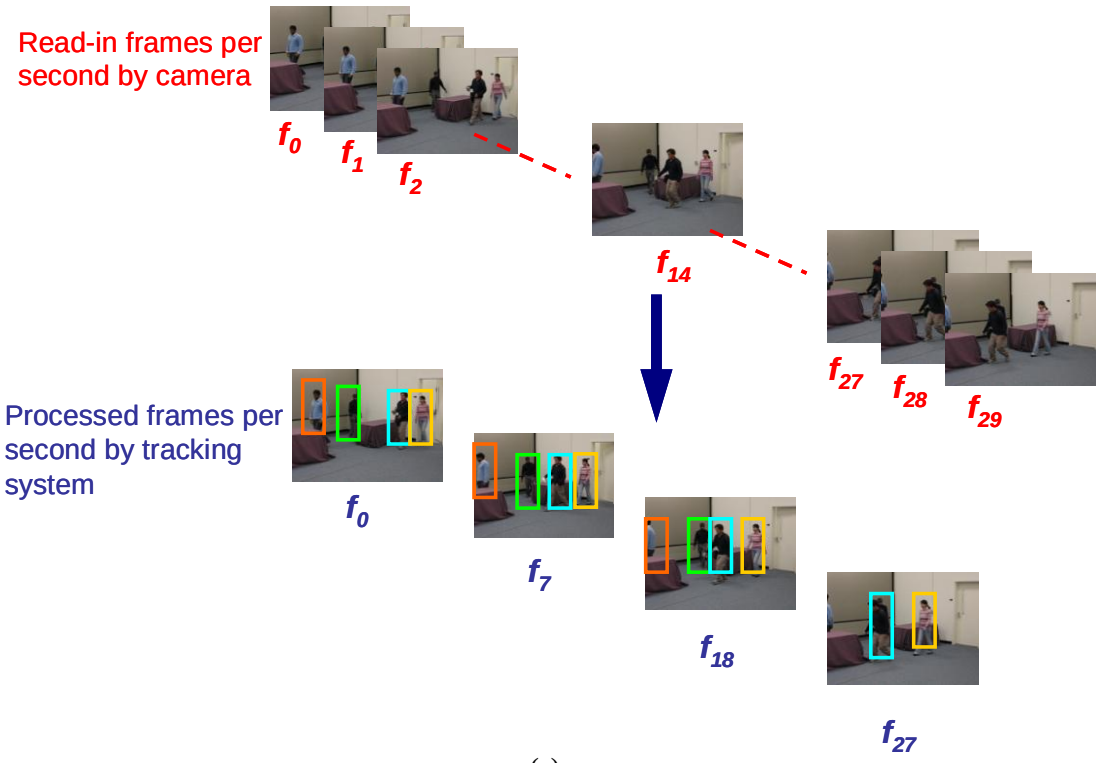
Where μ is a conversion scalar between the value of observation measure and world space. In essence, we want to minimize the size of overlapped FOVs to obtain the maximal coverage of the area to be monitored. However, the time needed to successfully execute consistent labeling is a crucial factor to determine whether the goal of continuous object tracking can be accomplished. Therefore, sufficient overlapped FOVs defined by Equation (6.11) must be reserved. The details about how to incorporate Equation (6.11) into camera placement problem is shown in the work of Yao [Yao08B]. In this dissertation, our contributions in the camera placement problem are to find out the importance of size of overlapped FOVs for the overall performance of automated surveillance system in terms of the continuity of the tracked object and derive Equation (6.11).

7 Adaptive resource management theory

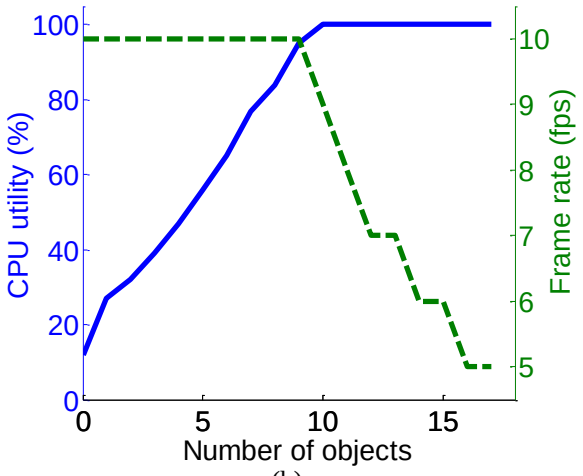
Most multiple objects tracking systems [Beleznai06, Luo05, Yao06] find it difficult to maintain a constant frame rate given limited resources. Note that frame rates in this paper represent the number of processed frames per second by the tracking system for executing functions such as tracking, crowd segmentation, and behavioral understanding, instead of the number of read-in frames by cameras themselves. This difference occurs due to the tracking system incapable of processing each read-in frame for accommodating the execution of all functions in real-time given limited resources, even though cameras themselves are capable to acquire more frames. Herewith, resources include (I) CPU capacity for executing object tracking, crowd segmentation, and behavior understanding in a automated manner [Hu04], and (II) network bandwidth for exchanging camera handoff information. The computational complexity of most existing tracking systems [Beleznai06, Luo05, Yao06] is of the order from $N_pO(n)$ to $N_pO(n^3)$ [Sebe05], where N_p is the number of tracked objects and n represents the number of steps to execute the algorithm. There inherently exists an upper bound on the number of objects that can be tracked simultaneously without deteriorating the system's frame rate.

Those unprocessed read-in frames may be dropped immediately or reserved for future reference. Regardless of those unprocessed read-in frames being dropped immediately or reserved for future reference, it is crucial for a tracking system to be able to maintain a reasonable frame rate in real-time. A lower frame rate may result in the following problems: (I) the surveillance system's real-time ability to automatically detect a threatening event degrades, causing possible observation leaks. This dangerous loophole impedes the practical application of these real-time multi-camera multi-object tracking systems [Shah03]; (II) the decreased frame rate also affects the performance of consistent labeling and consequently camera handoff, because a successful execution of consistent labeling requires accumulated information of the object of interest over a period of time [Khan03, Fluret08, Guler03]. The reduced frame rate leads to a decreased number of available frames/information for carrying out consistent labeling successfully. Figure 7.1(a) illustrates the difference between read-in frames per second by camera and processed frames per second by system in a real-life case. Figure 7.1(b) demonstrates one example of a decreased frame rate when the system discussed in [Yao06] is used to track an increased number of objects with limited CPU capacity.

The remainder of this chapter is organized as follows. Section 7.1 illustrates the overall system architecture of our proposed camera handoff algorithm with the adaptive resource management mechanism. Section 7.2 presents the trackability measure. Section 7.2 presents the adaptive resource management mechanism.



(a)



(b)

Figure 7.1 Illustration of system overload for a multiple object tracking system [Yao06]. (a) Illustration of the difference between read-in frames per second by camera and processed frames per second by tracking system in a real-life case. (b) The solid curve illustrates that the CPU utility increases and saturates as the number of objects increases. The dashed curve shows that the frame rate decreases as the number of objects increases after the CPU utility reaches 100%.

7.1 System architecture

The flow chart of the proposed camera handoff algorithm is shown in Figure 7.2, where operations are carried out at the handoff request and handoff response sides. Let the j^{th} camera be the handoff request side and the i^{th} object be the one that needs a transfer. To maintain persistent and continuous object tracking, a handoff request is triggered before the object of interest is untraceable or unidentifiable in the currently observing camera. The object of interest may become untraceable or unidentifiable due to the following reasons: (1) the object is being occluded by other objects; (2) the object is leaving the camera's FOV; and (3) the object's resolution is getting low. Accordingly, three criteria are defined in the trackability measure to determine when to trigger a handoff request: occlusion (M_O), distance to the edge of the camera's FOV (M_D), and resolution (M_S). Let $M_{O,ij}$, $M_{D,ij}$, and $M_{S,ij}$ be the M_O, M_D, and M_S values of the i^{th} object observed by the j^{th} camera, respectively. These three components $M_{O,ij}$, $M_{D,ij}$, and $M_{S,ij}$, to be discussed in details in Section 7.2, are scaled to [0, 1] where zero means that the object is untraceable or unidentifiable and one means that the camera has the best effectiveness in tracking the object.

Define the trigger criterion $C_{T,ij}$ as:

$$C_{T,ij} = \left[M_{O,ij} < T_O \wedge \left(\frac{dM_{O,ij}}{dt} < 0 \right) \right] \vee \left[M_{D,ij} < T_D \wedge \left(\frac{dM_{D,ij}}{dt} < 0 \right) \right] \vee \left[M_{S,ij} < T_S \wedge \left(\frac{dM_{S,ij}}{dt} < 0 \right) \right], \tag{7.1}$$

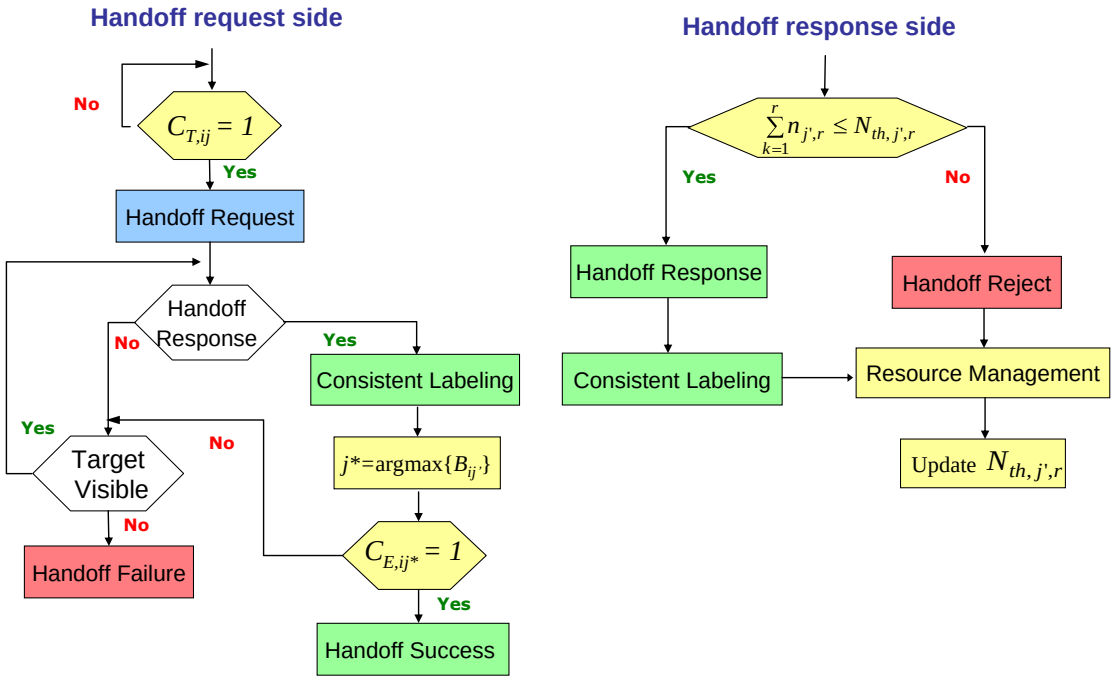


Figure 7.2 Flow chart of the proposed camera handoff algorithm.

where $\wedge$ and $\vee$, both logical symbols, represent 'and' and 'or' operations, respectively. T_O, T_D, and T_S, associated with M_O, M_D, and M_S, represent the predefined thresholds for triggering handoff and are mainly determined by the time needed for handoff execution and the objects' maximal moving speed. A handoff request, therefore, is triggered and broadcasted, if $C_{T,ij} = 1$, which suggests that at least one of the three components is below the predefined threshold and is decreasing. Note that in some applications such as the work of Lien and Huang [Lien06], each object of interest is tracked by multiple cameras to obtain more or better monitoring results. Our proposed handoff algorithm can still be applied to these applications, because each camera still needs to handoff the object of interest to another camera that is not tracking the object when either occlusion, low resolution, or falling out of its FOV occurs. In addition, we use Kalman filter to smooth the jitter effect in Equation (7.1), which is caused by multiple handoff requests being generated at the boundary due to noises.

Afterwards, the j^{th} camera keeps tracking the i^{th} object and waits for confirmation responses from adjacent cameras while the object is still visible. At the handoff response side, the $(j')^{th}$ camera examines its current load. Let $N_{th,j',r}$ denote the maximum number of objects with a priority rank smaller than or equal to r that can be tracked simultaneously and $n_{j',r}$ the number of objects with a priority rank r that have been tracked by the $(j')^{th}$ camera. A positive handoff response for the i^{th} object is granted, if $\sum_{k=1}^{r} n_{j',k} < N_{th,j',r}$, which represents the total number of tracked objects from different priority ranks has to be less than the maximum number of tracked objects in the system.

To achieve a higher acceptance rate or equivalently a higher handoff success rate, the thresholds $N_{th,j',r}$ should be adaptively adjusted according to the system's current load. Given limited capacity, more resources should be allocated to objects with higher priorities at the cost of dropping out objects with lower priorities. Such a system provides a higher threat awareness level compared with systems where objects have the same priority ranks. Sometimes additional requirements on the overload probabilities of objects with different priority ranks are given. To meet these requirements, we need an online learning process to automatically adjust the distribution of the capacities according to estimated system load.

Thus, for a more efficient allocation of limited resources, an adaptive resource management algorithm is proposed and implemented at the response side. With the adaptively adjusted resource allocation, a smaller number of objects will be dropped and hence a higher handoff success rate can be achieved. Furthermore, with the capability of actively selecting the objects with higher priorities to track, a constant frame rate can be maintained at the cost of dropping out objects with lower priorities if necessary. The derivation of the overload probability and the algorithm for adaptively adjusting the resources among multiple objects are introduced in Section 7.3.

Back to the handoff request side, if no positive handoff response is received before the j^{th} camera loses the track of the i^{th} object, a handoff failure is issued. Otherwise, consistent labeling is carried out between the handoff request side and all available candidate cameras. In order to select the most appropriate candidate camera to take over the object of interest in the pool of candidate cameras, the one with the lowest system load $P_{O,ij'}$ and the highest trackability measure

$Q_{ij'}$ is chosen:

$$B_{ij'} = (1 - P_{O,ij'})Q_{ij'}, \tag{7.2}$$

where $P_{O,ij'}$ is the overload probability of the i^{th} object in the $(j')^{th}$ camera and $Q_{ij'}$ denotes the trackability measure of the i^{th} object in the $(j')^{th}$ camera. The detailed definition of $Q_{ij'}$ and $P_{O,ij'}$ are given in Section 7.2 and 7.3, respectively. The term (1-$P_{O,ij'}$) is included to reduce the chances of choosing a camera with high system load, which ensures an evenly distributed system load across all cameras.

The execution criterion $C_{E,jj*}$ is defined as:

$$C_{E,ij*} = \left[\left(M_{O,ij} < M_{O,ij*} \right) \vee \left(\frac{dM_{O,ij*}}{d} > 0 \right) \right] \wedge \left[\left(M_{D,ij} < M_{D,ij*} \right) \vee \left(\frac{dM_{D,ij*}}{d} > 0 \right) \right] \wedge \left[\left(M_{S,ij} < M_{S,ij*} \right) \vee \left(\frac{dM_{S,ij*}}{d} > 0 \right) \right], \tag{7.3}$$

Since an efficient tracking system should be able to direct camera handoff for continuous and automated tracking before the tracked object is occluded or falls out of the FOV of currently observing camera. In the meanwhile, system load can be evenly distributed without deteriorating the frame rate of each camera. Thus, the i^{th} object is transferred to the $(j*)^{th}$ camera if $C_{E,ij*} = 1$.

7.2 Trackability measure

In the following discussion, formulas are derived for a single target observed by a single camera. For clear representation, the subscripts *i* and *j* are omitted. Assume that from object tracking the target image's relative scale ρ and center of mass $\mathbf{g} = [g_x \quad g_y]^T$ are estimated. The resolution component M_S is defined as:

$$M_S = \alpha_S \frac{f}{Z_r} = \alpha_S \rho, \tag{7.4}$$

where *f* represents the camera's focal length, Z_r is the average target depth, and α_S denotes the normalization coefficient.

To reserve enough computation time for the execution of the handoff between cameras, the object should remain at a distance from the boundaries of the camera's FOV. This margin distance is also affected by the object's depth. When the object is at a closer distance to the observing camera, its projected image undergoes a larger displacement in the image plane. Therefore, a larger margin should be reserved. In our definition, a varying polynomial power is used to achieve different decreasing/increasing rates and in turn different margin distances. The M_D term is defined as:

$$M_D = \alpha_D \left[\left(\frac{N_x}{2} - |g_x - \frac{N_x}{2}| \right)^2 + \left(\frac{N_y}{2} - |g_y - \frac{N_y}{2}| \right)^2 \right]^{\beta_1 \rho + \beta_0}, \quad (7.5)$$

where α_D is a normalization factor and N_x (N_y) denotes the width (height) of the image. The coefficients ρ, β_1, and β_0 are used to adjust the polynomial power and depend on the depth of field.

In order to continuously track multiple objects, the system should be able to transfer the tracked object with latent occlusion to another camera with a clear view. Therefore, the occlusion caused by objects' motion is also considered. The M_O term is defined as:

$$M_O = \alpha_O \left[\min_{i \neq j} \left\{ (g_{x,i} - g_{x,j})^2 + (g_{y,i} - g_{y,j})^2 \right\} \right]^{\beta_1 \rho + \beta_0}, \quad (7.6)$$

where α_O is a normalization weight. $[g_{x,i} \quad g_{y,i}]^T$ and $[g_{x,j} \quad g_{y,j}]^T$ denote the centers of mass of any pair of objects in the field of view of currently tracking camera. Occlusion can be caused by stationary obstacles, such as tables and cabinets, or other moving pedestrians in the environment. Thus, those objects do not only represent mobile objects, but stationary. Nevertheless, how to differentiate which one is in the front or in the back is not the scope of this paper. Interesting readers can refer the work of Hoiem et. al [Hoiem07]. In conclusion, the trackability measure is given by:

$$Q = M_O (w_S M_S + w_D M_D), \quad (7.7)$$

where w_S, and w_D are importance weights for the resolution and distance components, respectively. The sum of these importance weights is one.

7.3 Adaptive resource management

In this section, we first derive the overload probabilities of objects at different priority ranks and then introduce our resource management algorithm. In the following discussion, formulas are derived for any single camera. For clear representation, the subscript *j* is omitted.

7.3.1 Probability of camera overload

Assume that the arrival of objects with priority rank *r* follows a Poisson distribution with a rate λ_r. The amount of time that an object remains within the camera's FOV, is independent and follows an exponential distribution with mean of $\frac{1}{\mu}$. Let $N_{th,r}$ be the maximum number of objects with a priority rank smaller than or equal to *r* that can be tracked simultaneously. We

deliberately add $N_{th,0}$=0 to simplify the formulation. Let the maximum number of objects that can be tracked simultaneously be N_{max} and the total number of priority ranks be N_{pr}.

To derive the overload probability of objects at different priority ranks, a multi-object tracking system is modeled as an *M/M/N*$_{max}$*/N*$_{max}$*/FCFS* queuing system, where *M* represents arrival or departure distribution as a Poisson distribution, and the serving rule is first come first serve (*FCFS*) [Klennrock75, Huang04]. Such a system constitutes a Markov process of the birth-death type, as shown in Figure 7.3. We examine the queuing system when it is at equilibrium. Under proper conditions, such equilibrium will be reached after the system has been operating for a period of time. It also implies that the probability of *n* objects being tracked, *P*(*n*), eventually becomes stable, where *n* ranges from 0 to N_{max}. Therefore, the probability of the n^{th} state can be computed given the probability of the $(n\text{-}1)^{th}$ state:

$$P(n) = \frac{\sum_{k=r}^{N_{pr}} \lambda_k}{n\mu} P(n-1), \tag{7.8}$$

where $N_{th,r-1} < n \le N_{th,r}$. This relation leads to:

$$P(n) = \frac{P(0)}{n!} \prod_{k=1}^{r-1} \left(\sum_{l=k}^{N_{pr}} \frac{\lambda_l}{\mu} \right)^{N_{th,k} - N_{th,k-1}} \left(\sum_{k=r}^{N_{pr}} \frac{\lambda_k}{\mu} \right)^{n - N_{th,r-1}}, \tag{7.9}$$

where

$$P(0) = \left\{ \sum_{n=0}^{N_{\max}} \left[\frac{1}{n!} \prod_{g=1}^{r-1} \left(\sum_{h=g}^{N_{pr}} \frac{\lambda_h}{\mu} \right)^{N_{th,g} - N_{th,g-1}} \left(\sum_{g=r}^{N_{pr}} \frac{\lambda_g}{\mu} \right)^{n - N_{th,r-1}} \right] \right\}^{-1}, \tag{7.10}$$

According to Equations (7.9) and (7.10), the overload probability for the object with a priority rank of *r* is given by

$$P_{O,r} = \sum_{n=N_{th,r}}^{N_{\max,}} P(n), \tag{7.11}$$

The overload probability is one important criterion to evaluate the performance of a multi-camera system fulfilling multiple object tracking. It determines the number of objects that may be dropped due to limited resources. Therefore, in practice, it is desirable to distribute the resources dynamically according to the system's current load and the object's priority rank.

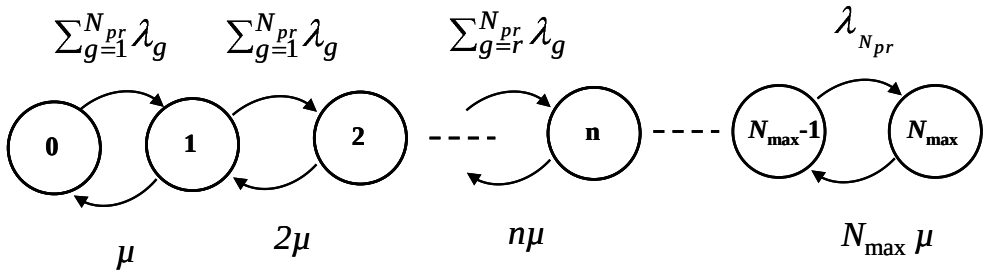


Figure 7..3 Illustration of the state transition of an *M/M/* N_{max} */* N_{max} */*∞*/FCFS* queuing system, which is used to model a multi-object tracking system.

From the above derivations, we learn that $N_{th,r}$ determines the overload probabilities. Given the overload probabilities for objects at different priority ranks, we could adjust these thresholds to achieve the requirements. If the real-time estimated overload probability, $\hat{P}_{O,r}$, for the object with a priority rank r exceeds the desired overload probability, $P_{th,r}$, we need to decrease the thresholds $N_{th,\,k}$ with $1 \le k < r$ or increase the thresholds $N_{th,\,k}$ with $r \le k \le N_{pr}$. Based on this key concept, we develop our adaptive resource management algorithm.

7.3.2 Algorithm description

The flow chart of our resource management algorithm is illustrated in Figure 7.4. If given the known arrival rate λ_r with $1 \le r \le N_{pr}$, the initial thresholds $N_{th,r}$ can be computed as $N_{th,r} = \frac{\sum_{k=1}^{r} \lambda_k}{\sum_{k=1}^{N_{pr}} \lambda_k} N_{\max}$. If not, the initial values can be set to $N_{th,r} = \frac{r N_{\max}}{N_{pr}}$. Let n_r be the number of tracked objects with priority rank r. As we mentioned before, if $\sum_{k=1}^{r} n_k < N_{th,r}$, the handoff request is accepted. Otherwise the handoff request is rejected. Afterwards, the real-time arrival rates of objects with different ranks $\hat{\lambda}_r$ are estimated during the time frame $\frac{1}{\mu}$. Note that even in scenarios with known average arrival rates, it is also necessary to estimate the real-time arrival rates so as to adjust resource allocation among objects with different ranks according to current system load. Given the estimated $\hat{\lambda}_r$, the real-time overload probability, $\hat{P}_{O,r}$, for objects with rank r can be computed according to Equation (7.11). The estimated overload probability $\hat{P}_{O,r}$ is then compared with the predefined or desired overloa
d probability $P_{th,r}$. If $\hat{P}_{O,r} > P_{th,r}$, the thresholds $N_{th,r-1}$ and $N_{th,r}$ should be adjusted. Ideally, we want to increase $N_{th,r}$ and decrease $N_{th,r-1}$.

However, varying $N_{th,r-1}$ and $N_{th,r}$ also affects the overload probability of objects from other ranks. In addition, the estimated overload probability $\hat{P}_{O,r}$ may fluctuate, which in turn induces unnecessary adjustment of the thresholds. Therefore, to smooth the decisions over a period of time and incorporate the requirements from objects of other ranks, a flag is set up for the

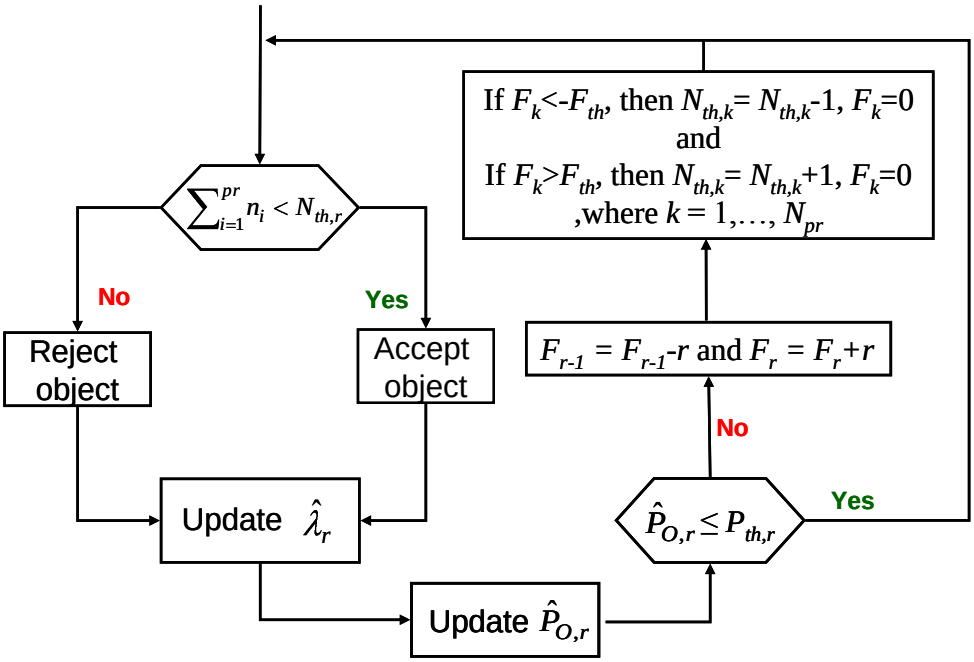


Figure 7.4 Flow chart of the proposed adaptive resource management scheme. In general, if the real-time estimated overload probability, $\hat{P}_{O,r}$, for the object with a priority rank r exceeds the predefined or desired overload probability $P_{th,r}$, we need to decrease the thresholds $N_{th,k}$ with $1 \leq k < r$ or increase the thresholds $N_{th,k}$ with $r \leq k \leq N_{pr}$.

thresholds at each priority rank, which is defined as F_r. If $\hat{P}_{O,r} > P_{th,r}$, decrease F_{r-1} by r suggesting that a decrease in $N_{th,r-1}$ is requested and increase F_r by r suggesting that an increase in $N_{th,r}$ is preferred. Since it is cumulative, F_r takes the previous decision into consideration as well. If multiple handoff requests are received, the same procedure repeats for each object and the decisions from multiple objects are combined in F_k with k=1, …, N_{pr}. The contribution in F_k from each object is associated with its priority rank. In so doing, more importance is assigned to the decisions from objects with higher priorities and the following adjustment of the thresholds favors a smaller overload probability for objects with higher priorities. In addition, a more prompt response is also achieved for objects with higher priorities. In addition, the priority rank is included to improve the system's level of threat awareness. Priority ranks can be assigned to tracked objects according to their behaviors. For example, in the surveillance of an airport, passengers moving along the indicated direction (from the gates to exit) in the hall way are assigned with a lower priority while passengers moving in the opposite direction are assigned with a higher priority.

After all the objects have been processed, the thresholds are updated. If $F_k > F_{th}$, $N_{th,k}$ is increased by one, where F_{th} is a predefined threshold. If $F_k < -F_{th}$, $N_{th,k}$ is reduced by one. After the adjustment of $N_{th,k}$, the corresponding F_k is reset to zero. $N_{th,k}$ remains the same if $|F_k| \leq F_{th}$. The complexity of computing $\hat{P}_{o,r}$ and F_k is of the order $O\left(\sum_{k=1}^{N_{pr}} n_k\right)$. The adjustment of thresholds $N_{th,k}$ has a computational complexity of $O(N_{pr})$. As a result, the proposed resource adjustment is able to dynamically relocate the available resources with marginally increased computational cost in comparison with the complexity of multiple object tracking and consistent labeling.

7.3.3 Example System

To further study the effect of adjusting $N_{th,r}$ for adaptive resource management, we consider the asset monitoring system as an example. In this application, people who are close to or carry the valued asset should be adaptively allocated more resource when the system's load is high. This is because the tracking system needs to continuously track the people to immediately detect any threats to the valued asset. A system with N_{pr}=2, therefore, represents a system with only two types of objects, high and low priorities. Let λ_H and λ_L be the arrival rate of objects with high and low priorities. The probability of n tracked objects is given by

$$P(n) = \begin{cases} \frac{P(0)}{n!}\left(\frac{\lambda_H + \lambda_L}{\mu}\right)^n & 0 \leq n \leq N_{th} \\ \frac{P(0)}{n!}\left(\frac{\lambda_H + \lambda_L}{\mu}\right)^{N_{th}}\left(\frac{\lambda_H}{\mu}\right)^{n-N_{th}} & N_{th} < n \leq N_{\max} \end{cases}, \tag{7.12}$$

With

$$P(0) = \left(\sum_{n=0}^{N_{th}} \frac{1}{n!} (\frac{\lambda_H + \lambda_L}{u})^n + \sum_{n=N_{th}+1}^{N_{\max}} \frac{1}{n!} (\frac{\lambda_H + \lambda_L}{u})^{N_{th}} (\frac{\lambda_H}{u})^{n-N_{th}} \right)^{-1}, \quad (7.13)$$

The overload probabilities for the object of high and low priorities are $P_{O,H} = P(N_{\max})$ and $P_{O,L} = \sum_{n=N_{th}}^{N_{\max}} P(n)$. These two probabilities are monotonously increasing and decreasing functions of the threshold N_{th} as shown in Figure 7.5 Suppose we have $\frac{\lambda_H}{\mu} = 2$, $\frac{\lambda_L}{\mu} = 1$, and N_{max}=6. The initial N_{th} is initialized by $\frac{\sum_{g=1}^{r} \lambda_g}{\sum_{g=1}^{N_{pr}} \lambda_g} N_{\max} = 2$. The corresponding $P_{O,H}$ and $P_{O,L}$ are 0.015 and 0.710, respectively. The $P_{O,L}$ is much higher than the probability $P_{th,L} = 0.2$. Our resource management algorithm is able to increase N_{th} by one at one time so as to decrease $P_{O,L}$. At equilibrium, we arrive at N_{th}=5 resulting in $P_{O,H}$=0.035 and $P_{O,L}$=0.142. Figure 7.5 also depicts the adjustment.

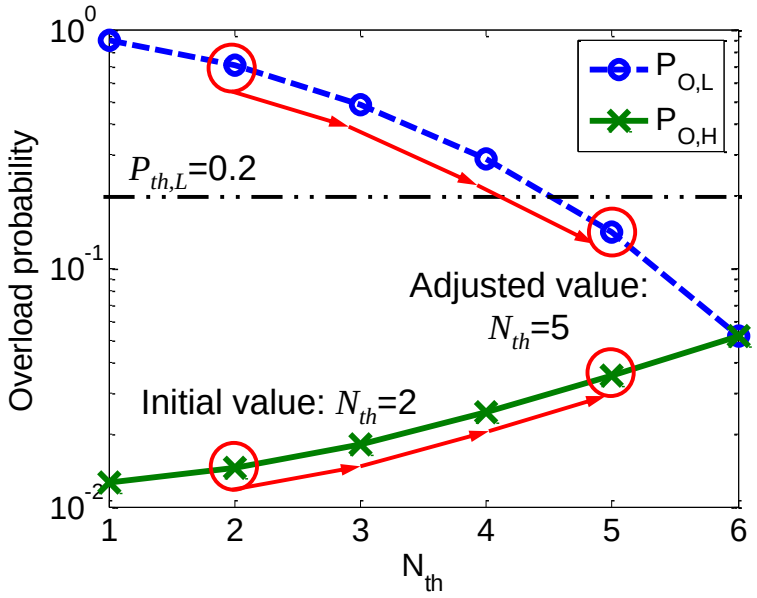


Figure 7.5 Illustration of the overload probabilities $P_{O,H}$ and $P_{O,L}$ as functions of N_{th}. $\frac{\lambda_H}{\mu} = 2$, $\frac{\lambda_L}{\mu} = 1$, N_{max}=6. The corresponding $P_{O,H}$ and $P_{O,L}$ are 0.015 and 0.710, respectively. In the beginning, the $P_{O,L}$ is much higher than the probability $P_{th,L} = 0.2$. Our resource management algorithm is able to increase N_{th} by one at one time so as to decrease $P_{O,L}$. At equilibrium, we arrive at N_{th}=5 resulting in $P_{O,H}$=0.035 and $P_{O,L}$=0.142.

8 Mobile tracking platform

Video tracking, surveillance systems, and robotic platforms are fields that have been well studied in the past decade. However, in the majority of surveillance and video tracking systems, the cameras are stationary. The stationary cameras systems require the desired object to stay within the surveillance range of the system. If the object goes beyond this range, it no longer becomes tractable. One solution to this problem is to design a mobile system that uses a laser range sensor, and a visual-spectrum camera, to track the moving object and avoid obstacles simultaneously so that each object of interest can still be monitored. This research topic has been partially studied in several different areas. Studies made by the automotive industry in this area develop systems that assist a human driver for safety or comfort [Perez04, Philomin00]. NASA has applied this to help astronauts carry more equipment while walking on the moon [Graham03]. These systems are primarily concerned with object tracking, and are not concerned with the obstacle avoidance problem. One popular approach used for obstacle avoidance is the Potential Fields method. In the Potential Fields method, an artificial field is generated where a goal position produces an attractive force to the robot, and obstacles generate a repulsive force on the robot. The resultant force will decide the moving direction of the robot. Since traditional applications of Potential Fields methodologies do not allow tracking a moving object, the development of a modified version of a potential field method that can be used for a mobile system is necessary.

Figure 8.1 represents the whole system architecture. In general, the overall system consists of six main phases: image input, object tracking, robot control, obstacle detection, obstacle avoidance, and robot mobility phases. If no obstacles are detected, the system skips the obstacle avoidance phase, and only uses five phases. The following sections explain how each phase works individually, and how the various phases work in conjunction with each other.

The Logitech Web Camera has a fixed view and is attached to the robotic platform. It is used to acquire color-based 320x240 images. The camera is tasked to follow the tracked object. Lucas and Kanade's algorithm [Lucas81] does not use an iterative method to compute the optical flow and it is less affected by illuminant changes, which makes this method a more appropriate choice for real-time object tracking operations. We use this method to compute motion vectors for the tracked object in two consecutive images and then perform four different directions of the tracked object which include moving forward, moving backward, moving toward right, and moving toward left. Because there are so many motion vectors for the tracked object in the two consecutive images, we use Equation (8.1) to obtain the mass motion vector of the tracked object

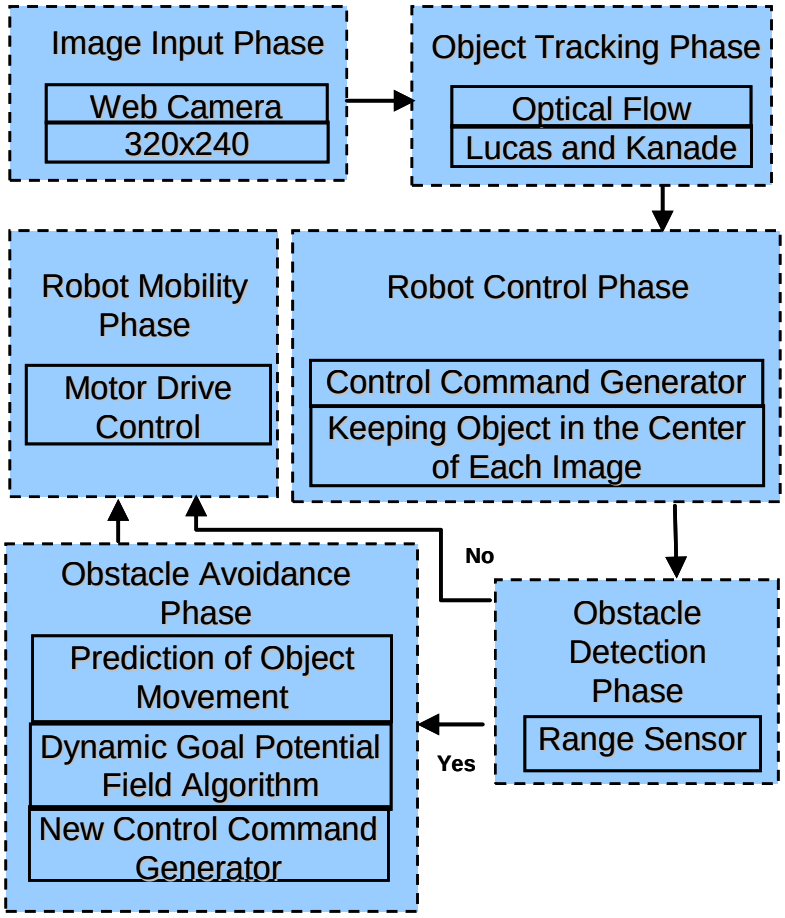


Figure 8.1 The illustration of the overall system for the mobile tracking platform, which it includes one visual camera for single object tracking and one range sensor for obstacle avoidance.

$$M = \frac{\sum_{i=1}^{N} X_i}{N}, \tag{8.1}$$

where M represents the mass motion vector of the tracked object (in 1×2 matrix form). X_i represents each motion vector of the tracked object (in 1×2 matrix form). N represents amount of total motion vector.

Once the object tracking phase is in control of monitoring the direction of the moving object, it will continue to send the robot control phase the current mass motion vector, M, of the moving object. It uses this to compute the difference between it and the origin, M_0, which represents the motion vector of the tracked object that stays in the center of the image. These operations facilitate the calculation of the robot's control commands. Equation (8.2) represents the method used in the robot control phase that eventually generates the difference vector between the world coordinate system and the image coordinate system.

$$(M - M_o) \times \begin{bmatrix} X_i & Y_i \\ X_j & Y_j \end{bmatrix} = W, \tag{8.2}$$

where $\begin{bmatrix} X_i & Y_i \\ X_j & Y_j \end{bmatrix}$ represents the conversion matrix which converts the image coordinates into the 2D world coordinate system. W represents the difference vector in the world coordinate system. Figure 8.2 shows the concept of the conversion method.

In Figure 8.2, W_0 represents the origin vector, [0, 0], in the 2D world coordinate system. R represents the robot vector, [R_x, R_y], in the 2D world coordinate system. θ describes the angular measure of turn required for the robot in order to keep the tracked object in the origin of the 2D world coordinate. θ can be computed by Equation (8.3),

$$\theta = \cos^{-1} \frac{(\overrightarrow{W_0 - R}) \cdot (\overrightarrow{W - R})}{\left\| (\overrightarrow{W_0 - R}) \right\| \left\| (\overrightarrow{W - R}) \right\|}. \tag{8.3}$$

After making a turn, the robot then determines the distance to move, either forward or backward, for keeping it in a fixed distance from origin vector, W_0.

Before the robot sends a command to move forward (or backward), it uses a laser scanner (SICK - LMS 200) to sense if there is any obstacle in its projected path. If no obstacle is detected, the robot mobility phase is activated. Subsequently, the control of the system returns back to the image input phase. Otherwise, the system uses the obstacle avoidance phase for generating another robot control command in order to avoid the obstacle. The obstacle avoidance phase uses the modified Potential Fields methodology mentioned earlier. The advantage of Potential Fields methods is that they are simple and fast. The most significant disadvantage concerns local minima problems which plagues Potential Fields methods. Essentially, this means that these methods cannot guarantee to find a path between the source and destination configuration even though a possible path does exist.

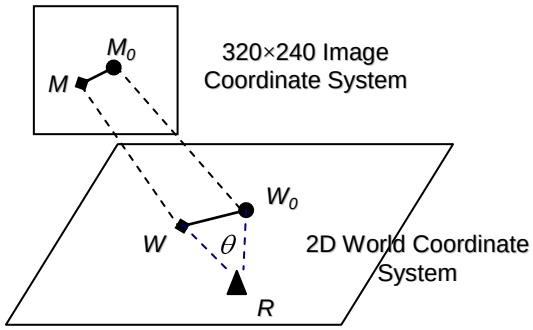


Figure 8.2 Conversion from image to 2D world coordinate system.

For this application, the robot needs to keep tracking the object while avoiding obstacles. Since the object might move to a new position when the robot is avoiding obstacles, traditional Potential Fields methods can not be directly applied because these methods assume the goal position is static (while the mobile robot is avoiding obstacles).

To deal with this dynamic goal position problem, the obstacle avoidance algorithm needs to adjust its path corresponding to the change of its destination during obstacle avoidance. We propose a new algorithm called Dynamic Goal Potential Fields which is based on the traditional Potential Fields methods to solve this type of problems. The Dynamic Goal Potential Fields algorithm is based on the following:

1. Using the current configuration, goal configuration and sensor data, it runs a basic Potential Fields algorithm to predict a path;
2. If the goal configuration does not change too much, then the robot follows this path to avoid any obstacle;
3. If the goal configuration moves to a new position which has a big change from the old position, the algorithm randomly chooses some points in the predicted path and runs the basic Potential Fields method to compute several paths starting from these points based on current sensor data;
4. The path with the lowest cost is selected (based on Euclidian distance). The robot is now using the new path to move to the new goal configuration.
5. Repeat it until reaching the goal.

Figure 8.3 shows the concept of Dynamic Potential Fields method. If the goal configuration does not change too much during the obstacle avoidance procedure, the Dynamic Potential Fields method is similar to the traditional Potential Fields method except that a predicted path is retained in every step. This predicted path might not be the same as the exact path taken by the robot because it is based only on the current sense data. If the goal configuration has a big change, the Dynamic Goal Potential Fields method has the capability to quickly adjust its path to move to the new position with low cost.

Like the Potential Fields method, Dynamic Goal Potential Field method is a local path planning method. It cannot guarantee to find the optimal path to move towards the goal. What it does is to ‘guess’ the best path moving towards goal based on current information about the environment. Furthermore, to achieve high computation efficiency, when the goal position changes to a new position, this algorithm does not compute every possible adjusted path starting

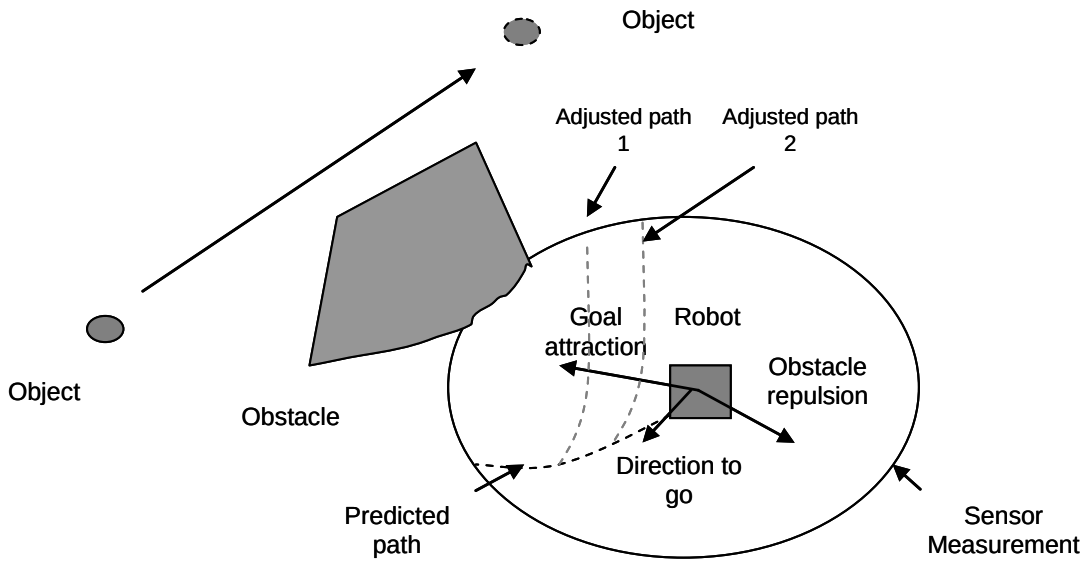


Figure 8.3 The concept of dynamic goal potential field method.

along the predicted path. Instead, it randomly samples several steps along the predicted path. Basically the more samples it takes along the predicted path, the better adjusted-path the algorithm can get. However the computational cost will be too high for achieving real-time reaction with high samples.

To achieve more computational efficiency, we do not take local minimal into account while computing adjusted paths. If an adjusted path is into a local minimal, we abandon this path by simply set the cost of this path to be infinite. The cost of each adjusted path can be calculated as follows:

$$C_i = C_{si} + C_{mi} + C_{li}, \tag{8.4}$$

Where C_i is the cost of i-th adjusted path. C_{si} is the Euclidian distance from the current position to the starting point on the predicted path. C_{mi} is the Euclidian distance from the starting point on the predicted path to the end of the i-th adjusted path, if this path is in a local minimal, then C_{mi} would be set to be infinite. C_{li} is the Euclidian distance from the end point of the i-th adjusted path to the goal position.

If the goal configuration does not change too much during the obstacle avoidance procedure, the Dynamic Goal Potential Fields method is similar to the traditional Potential Fields method except that a predicted path is retained in every step. This predicted path might not be the same as the exact path taken by the robot because it is based only on the current sense data. If the goal configuration has a big change, the Dynamic Goal Potential Fields method has the capability to quickly adjust its path to move to the new position with low cost. Here we only take robot's moving distance as account. A more reasonable measurement may need to consider the cost of robot motion as well. For example, the time for a robot to make a turn sometimes is longer than

to move a certain distance. And a small orientation error will cause big distance error after a long distance moving. So we probably should weight orientation with high value.

9 Experimental Results

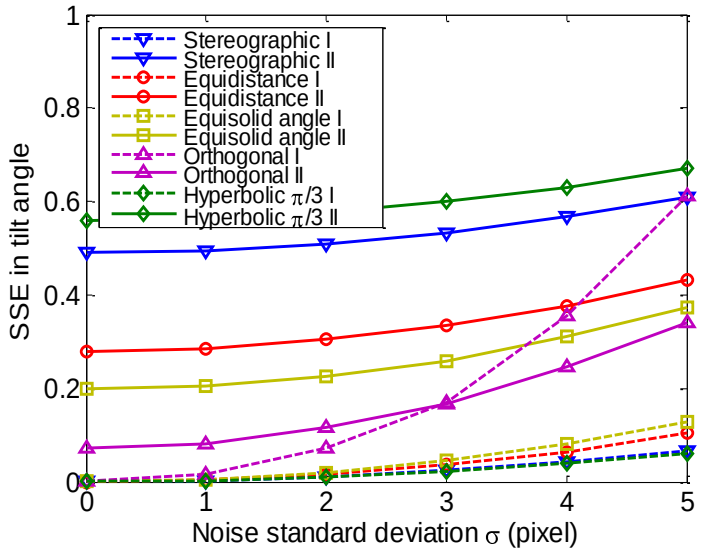
The chapter illustrates experimental results for the heterogeneous mapping of omnidirectional and PTZ cameras, homogeneous mapping of PTZ cameras, target hopping application, camera handoff and determination of size of overlapped views for multiple omnidirectional cameras, camera handoff with adaptive resource management, and the mobile tracking platform.

9.1 Heterogeneous mapping

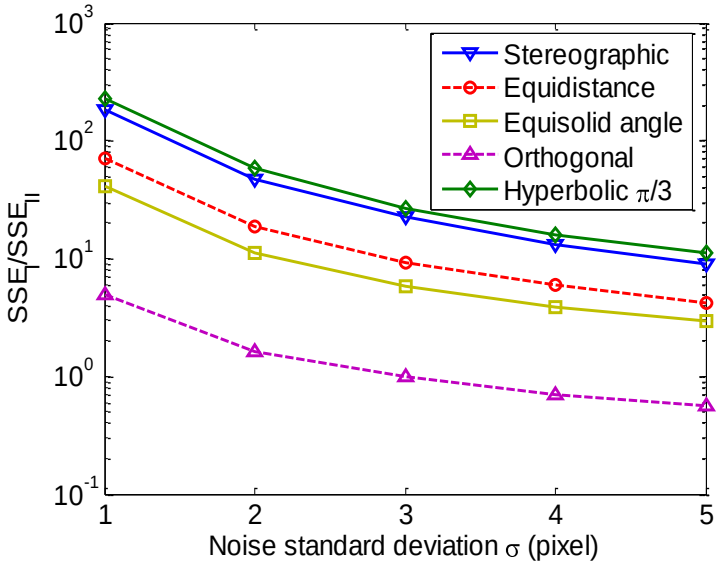
In this section, we first compare the performance of our geometry and homography calibration methods with the reference methods [Cui98, Scotti05] using synthetic/real datasets and real-time tracking sequences. Comparison among the sequential visiting, nearest neighbor, and our adaptive methods for the NBT problem (Target hopping application) is then illustrated.

9.1.1 Synthetic calibration data

The synthetic dataset includes points uniformly sampled on a cube having a dimension of X and Y: (-5m, 5m) and Z:(0, 2m). The omnidirectional camera is placed at the proximity of (0, 0, 2m). The pan and tilt angles are estimated using the optical center of the omnidirectional camera as the coordinate origin. The pan angle is then independent of model selection and the zoom value has no effect on the projection of points. Therefore, only the estimation accuracy, expressed in the sum of squared error (SSE), of the tilt angle is studied, as shown in Figure 9.1. Generally speaking, the geometry calibration method outperforms the homography calibration method except for the orthogonal projection model with higher noise levels $\sigma \geq 3$. For a clear performance comparison, we define the following metric $\rho = \frac{SSE_I}{SSE_{II}}$ where SSE_I and SSE_{II} denote the sum of squared errors in the estimated tilt angles using the geometry and homography calibration methods, respectively. Figure 9.1(b) shows that ρ decreases as the noise level increases. When the noise standard deviation reaches 5 pixels, ρ is maintained within the range of (1, 10) indicating that the performance of both methods is comparable and of the same magnitude. The homography calibration method is a further generalization of the geometry calibration method, where a polynomial is used to directly model the spatial relation between the omnidirectional and PTZ cameras. Therefore, this amount of degradation in accuracy is expected.



(a)



(b)

Figure 9.1 Performance comparison of the geometry (1) and homography (2) methods for various projection models. (a) Estimation errors in the tilt angle. The geometry calibration method outperforms the homograhpy calibration method. (b) ρ: the ratio between SSE_I and SSE_{II}. As the noise level increases, the performance difference between the geometry and homography calibration methods decreases.

9.1.2 Real calibration data

Our indoor surveillance system includes an IQeye3 omnidirectional camera and a Pelco PTZ (Spectra III SE dome) camera. To calibrate the IQeye3 camera, a total of 8 images (resolution: 800×720 pixels) of a planar checkerboard are collected, with 49 control points per image. From our automated model selection, a polynomial of degree one was the best choice [Yao06].
The resulting camera intrinsic parameters are listed as follows:
(1) $r = 1.342\theta$;

(2) $K = \begin{bmatrix} 204.05 & 0.64 & 417.04 \\ 0 & 204.96 & 377.68 \\ 0 & 0 & 1 \end{bmatrix}$.

It is shown in [Yao06] that the influence from skew and aspect ratio is relatively trivial and thus is neglected without inducing increased error in estimating the pan and tilt angles for the PTZ camera. With the optical center of the omnidirectional camera as the coordinate origin, the pan and tilt angles are estimated by:

$$\begin{cases} \hat{\theta}_P = \tan^{-1}\left(\dfrac{y-377.68}{x-417.04}\right) \\ \hat{\theta}_T = \dfrac{\sqrt{(x-417.04)^2+(y-377.68)^2}}{204.05\times 1.342} \end{cases}. \tag{9.1}$$

A total of 96 samples uniformly distributed in the ground plane (Z=0) with an interval of 0.3m in both *X* and *Y* directions are collected as the training set for the homography calibration method. The resulting transformation functions are

$$\begin{cases} \hat{\theta}_P = 1.84 - 0.00416x + 0.00318y + 2.44\times 10^{-5}(y-377.68)^2 \\ \hat{\theta}_T = -0.37 + 0.0010x + 0.0017y + 1.57\times 10^{-5}(y-377.68)^2 \\ \quad\quad + 1.05\times 10^{-5}(x-417.04)(y-377.68) \end{cases}. \tag{9.2}$$

To compare the accuracy of spatial mapping, we conducted the following experiments. The 360 degrees' FOV of the omnidirectional camera is divided into evenly distributed 12 sectors. In each sector, 9 points are sampled at *R*= 0.25m, 0.5m, 0.75m, 1m, 1.5m, 2m, 3m, 4m, and 5m. The tested points cover 85% of the image plane and achieve a maximum tilt angle of 74 degrees for a camera at a height of *Z*=1.5m above the ground plane. Like the experiments with synthetic data, the pan angle is independent of model selection and the zoom value has no effect on the projection of points. Therefore, only the estimation accuracy of the tilt angle is studied.

Figure 9.2 depicts the averaged relative estimation error $\frac{\sum_i |\hat{\theta}_{T,i} - \theta_T|}{N_p \theta_T} \times 100\%$ with respect to the various θ_T tested. In the range of 20 to 75 degrees, less than 10% relative errors are observed. When fitted by a polynomial of degree three, the error rate increases substantially, especially for smaller tilt angles. Since the optimal model is indeed linear, a polynomial of a fixed degree three suffers from the over-fitting problem, where the resulting model is drawn towards the noise in

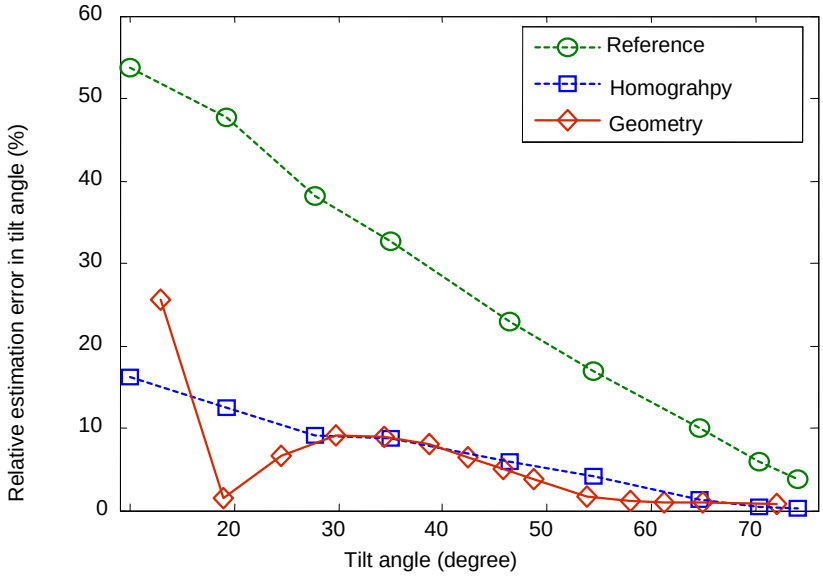


Figure 9.2 Performance comparison between our algorithm and the reference algorithm [Cui98] based on relative errors in the estimated $\hat{\theta}_T$. The reference algorithm uses a fixed model and suffers from over-fitting problem due to noisy data.

the data instead of the actual characteristics of the true data. The algorithm discussed in [Cui98] can produce similar performance only when a polynomial of degree three is the best fit according to the AIC criterion. Therefore, in general, our algorithms achieve a consistent accuracy independently of the actual projection model.

Different from the results based on synthetic data, similar estimation accuracy is achieved for both geometry and homography methods. The accuracy degradation from measuring the relative position of the two cameras deteriorates the performance of the geometry method. The advantage of using the homography calibration becomes evident due to its independence of the knowledge of the camera's projection model and relative position. We also compare the estimation accuracy of our algorithms with the algorithm proposed in [Scotti05] at the points listed in Table 9.1. Although different types of omnidirectional cameras are used, where a dioptric (fisheye) and a catadioptric camera are used in our system and the reference system [Scotti05], our system presents a better performance at all tested angles in terms of estimation accuracy. Based on the performance comparisons, we are able to conclude that our algorithms achieve improved accuracy in describing the spatial mapping between the omnidirectional and PTZ cameras. In addition, our algorithms are application independent and fully automated. The geometry calibration process follows the widely used Zhang's algorithm [Zhang00] and utilizes a simple planar checkerboard to learn the internal parameters of the omnidirectional camera. The homography method further simplifies the calibration process and reduces the dependency on the camera's characteristics and placement. From our experimental results, the homography method is able to achieve comparable accuracy as the geometry method with significantly improved flexibility to varying system configuration and reduced computational complexity.

Table 9.1 Performance comparison based on relative errors ($\hat{\theta}_T$)

Tilt angle (degree)	40	60	75
Reference algorithm [Scotti05] (%)	9.50	2.84	0.88
Geometry calibration (%)	7.50	2.63	0.38
Homography calibration(%)	7.52	1.03	0.83

9.1.3 Real-time tracking data

In our indoor real-time surveillance system, the omnidirectional camera (IQeye3) fulfills target detection, pan/tilt/zoom estimation, and object tracking. Background differencing and radial profile analysis [Cui98] are used for target detection and tracking. The mapping algorithms described in sections III are implemented for pan/tilt/zoom estimation. Omnidirectional images, with a resolution of 320×240, are obtained from the IQeye3 camera via an intranet connection. The resolution of the Pelco PTZ camera is 640×480. The PTZ camera, with the capacity to pan from 0^0-360^0, tilt from 0^0-90^0, and zoom from 1×-22×, receives the estimated pan, tilt, and zoom values from the omnidirectional camera and turns towards the detected target. The efficiency of the proposed geometry and homography calibration methods is validated via real-time sequences, as shown in Figure 9.3. The red circle highlights the motion detected by the omnidirectional camera. Ideally, the PTZ camera should be able to place the target at its image center by panning and tilting. The distance between the centroid of the target's image and the image center describes the accuracy of the mapping. The average deviations are 30.2% and 41.8% for the geometry and homography calibration methods, respectively, when normalized with respect to the image width. The difference between the desired (50×170 pixels) and actual target's image sizes describes the accuracy of the camera's zoom control. The average deviations are 15.4% and 19.1% for the geometry and homography calibration methods, respectively, when normalized with respect to the desired object size. For all tested positions, the target is maintained within the PTZ camera's FOV, which verifies the effectiveness of the collaboration between the omnidirectional and PTZ cameras.

9.2 Homogeneous mapping

In our experiments, we compare our proposed cooperative mapping approach with the reference algorithm [Chen07A] in an indoor surveillance system including two Pelco PTZ cameras(Spectra III SE dome with 640×480 pixels, 0^o ~ 360^o pang angle, 0^o ~ 90^o tilt angle, and 1 ~ 184 zoom position). To compare the accuracy between our and the reference algorithms [Chen07A], we conduct the following experiment. In our cooperative mapping approach, a total of 825 samples uniformly distributed in the scene are collected by a single moving person as the training set for the correspondence functions, which are shown in Equation (9.3). Figure 9.4 shows the estimation error in pan values, where Figure 9.4(a) and (b) indicate the estimation error in comparison with the original sample set (825 sample) and relative pan angles (0^o ~ 360^o),

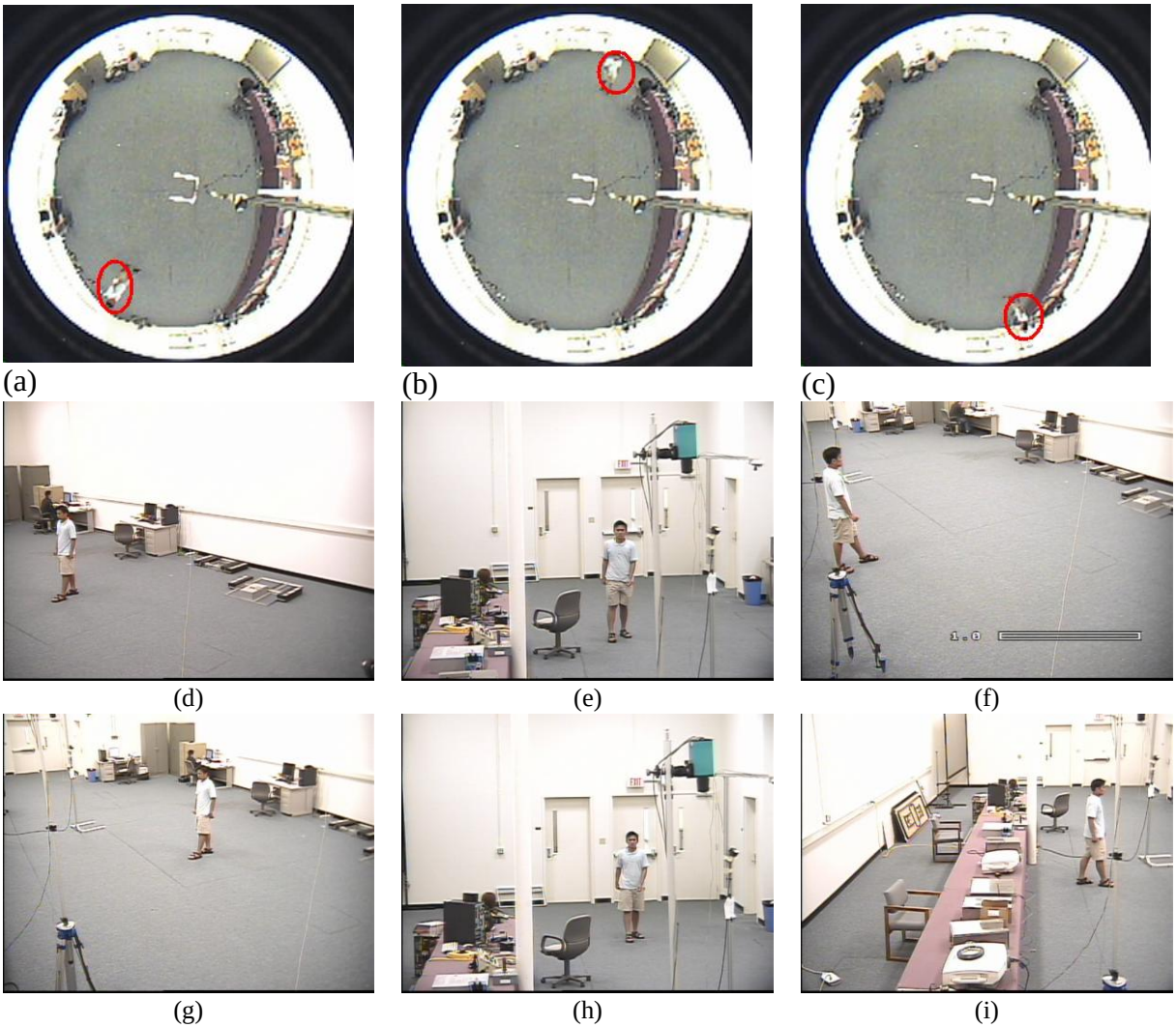
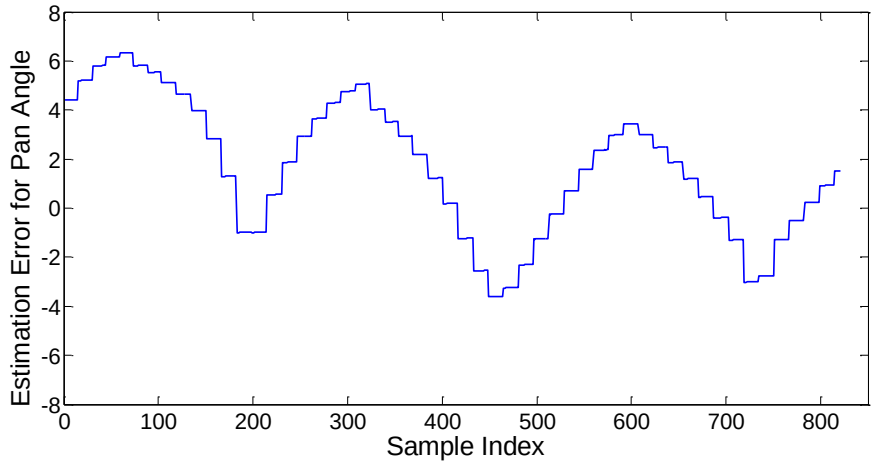
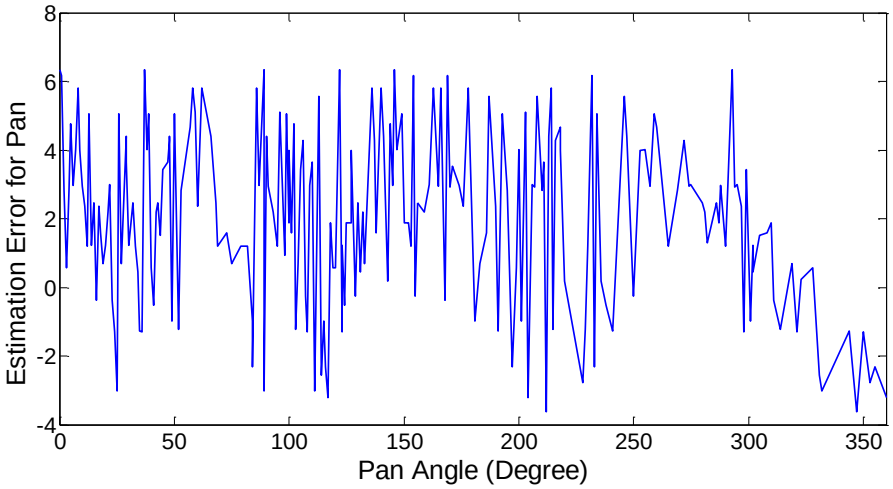


Figure 9.3 Performance of the geometry and homography calibration methods for a real-time dual camera system. (a)-(c) Frames from the omnidirectional camera. (d)-(f) Frames from the PTZ camera, when the geometry calibration method is used. (g)-(i) Frames from the PTZ camera, when the homography calibration method is used.

.



(a)



(b)

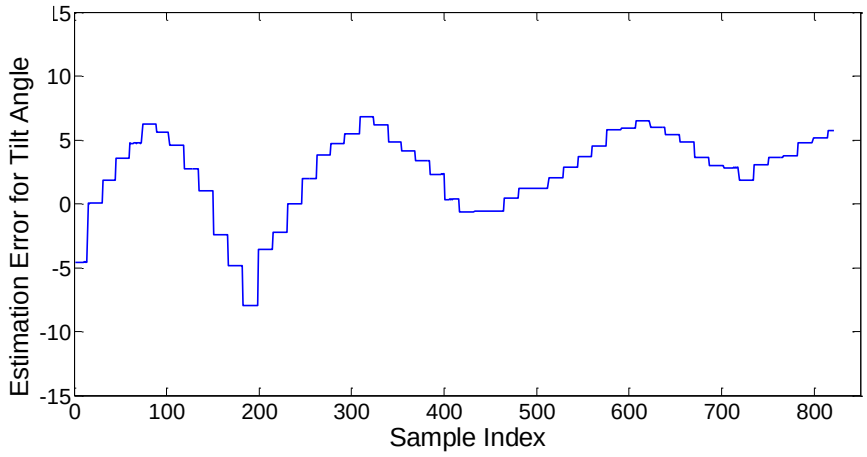
Figure 9. 4 Estimation errors in pan values: (a) comparison to the original sample set (825 samples), (b) relative pan angle (0^o ~ 360^o).

Respectively. Figure 9.5 shows the estimation error in tilt values, where Figure 9.5(a) and (b) indicate the estimation error in comparison with the original sample set (825 sample) and relative tilt angles (0^o ~ 90^o), respectively. Figure 9.6 shows the estimation error in zoom values, where Figure 9.6(a) and (b) indicate the estimation error in comparison with the original sample set (825 sample) and relative zoom positions (1 ~ 184). In average, the estimation error in pan angle is less than ± 6.3°, which depicts the averaged relative estimation error with respect to complete pan angle 360^o is 1.7% (it is calculated by $\frac{6.3}{360}$). The estimation error in tilt angle is less than ± 8.5°, which depicts the averaged relative estimation error with respect to complete tilt angle 90^o is 9.4% (it is calculated by $\frac{8.5}{90}$). The estimation error in zoom value is less than ± 19.5°, which depicts the averaged relative estimation error with respect to complete zoom value 184 is 10.5% (it is calculated by $\frac{19.5}{184}$).

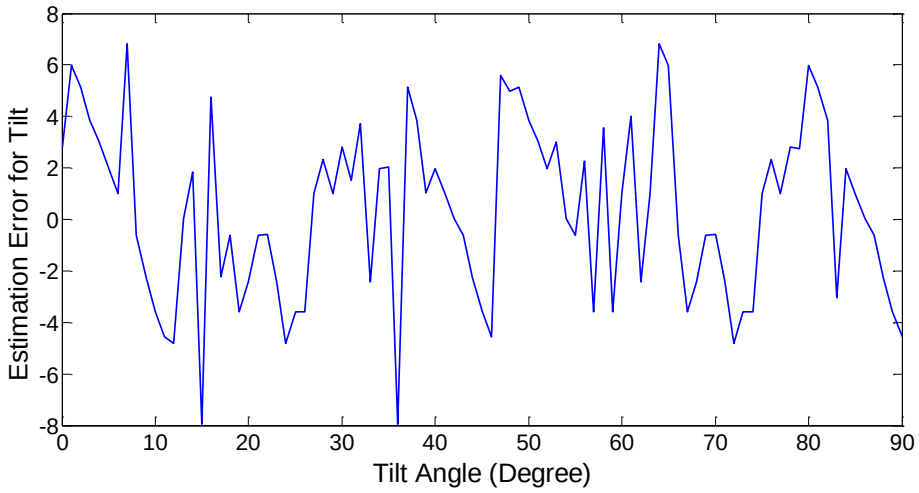
$$\begin{aligned}
\hat{\theta}_{P,h} &= 155.376 - 16.612\theta_{P,j} + 37.412\theta_{T,j} - 10.290x_j + 2.977y_i + 5.469\theta_{P,j}^2 - 23.364\theta_{T,j}^2 + 2.067x_j^2 - 0.804\theta_{P,j}\theta_{T,j} \\
&+ 6.764\theta_{P,j}x_j - 1.940\theta_{T,j}y_j - 0.658x_jy_j \\
\hat{\theta}_{T,h} &= -7.964 - 29.955\theta_{T,j} - 6.465y_j - 0.900\theta_{P,j}^2 + 24.060\theta_{T,j}^2 - 0.558y_j^2 - 1.386\theta_{P,j}x_j + 0.7291\theta_{P,j}y_j - 1.940\theta_{T,j}y_j \\
\hat{Z}_h &= -0.439 + 0.7324\theta_{P,j} - 0.6218Z_j + 0.1221y_j + 0.0817\theta_{P,j}^2 + 0.086\theta_{T,j}^2 + 0.5934Z_j^2 + 0.0218x_j^2 - 0.0141y_j^2 \\
&+ 0.0153\theta_{P,j}\theta_{T,j} + 0.0723\theta_{T,j}y_j + 0.0596Z_jx_j + 0.0596Z_jy_j + 0.0125x_jy_j
\end{aligned} \quad (9.3)$$

For the reference algorithm, we manually calibrate two PTZ cameras to learn their intrinsic parameters fist. This manual intervention impedes their direct application to surveillance systems with changing setups and larger number of PTZ cameras in the scene. Then we have 20 points forming a rectangular pattern in a 1 meter high table to estimate pose relationship based on back projections. Afterwards, we compare their accuracy to infer pixel correspondences between two PTZ cameras, where a single moving person is tested in the scene. Table 9.2 illustrates the comparison between our and reference algorithms. In Table 9.2, the averaged pixel distance deviation indicates the distance between the centroid of the object in the image and image center (320×240), when normalized with respect to the half of image width (320). The averaged pixel size deviation indicates the difference between the derived pixel size of the object and the desired pixel size (50×170=7500 pixels), when normalized with respect to the desired pixel size (7500). We can see that our proposed approach reduces the dependence on the knowledge of intrinsic parameters of the PTZ camera and improves the degree of autonomy at the cost of slightly decreased pixel accuracy, as compared to Chen and Wang' method.

Figures 9.7 and 9.8 show real-time video sequences for our proposed, and Chen and Wang's approaches. In Figure 9.7, the j^{th} PTZ camera uses Equation (9.3) to obtain $\hat{\theta}_{P,h}$, $\hat{\theta}_{T,h}$, and $\hat{Z}_h$ to direct the h^{th} PTZ camera to place the object in the center of the image with desired pixel size (7500) ideally. Figure 9.7shows the example where the single object is far away (18 meters) from the h^{th} PTZ camera (The tilt angle of the h^{th} PTZ camera is about 17^o). Figure 9.8 shows the example where the single object is close to (3 meters) the h^{th} PTZ camera (The tilt angle of the h^{th} PTZ camera is about 75^o). In Figures 9.7 and 9.8, both (a) and (b) show the first row shows five different locations in images of the j^{th} PTZ camera, the second row shows their respective pixel locations and sizes, derived by our approach, in images of the h^{th} PTZ camera, and the third row shows their respective pixel locations and sizes, derived by Chen and Wang's



(a)



(b)

Figure 9.5 Estimation errors in tilt values: (a) comparison to the original sample set (825 samples), (b) relative tilt angle (0^o ~ 90^o).

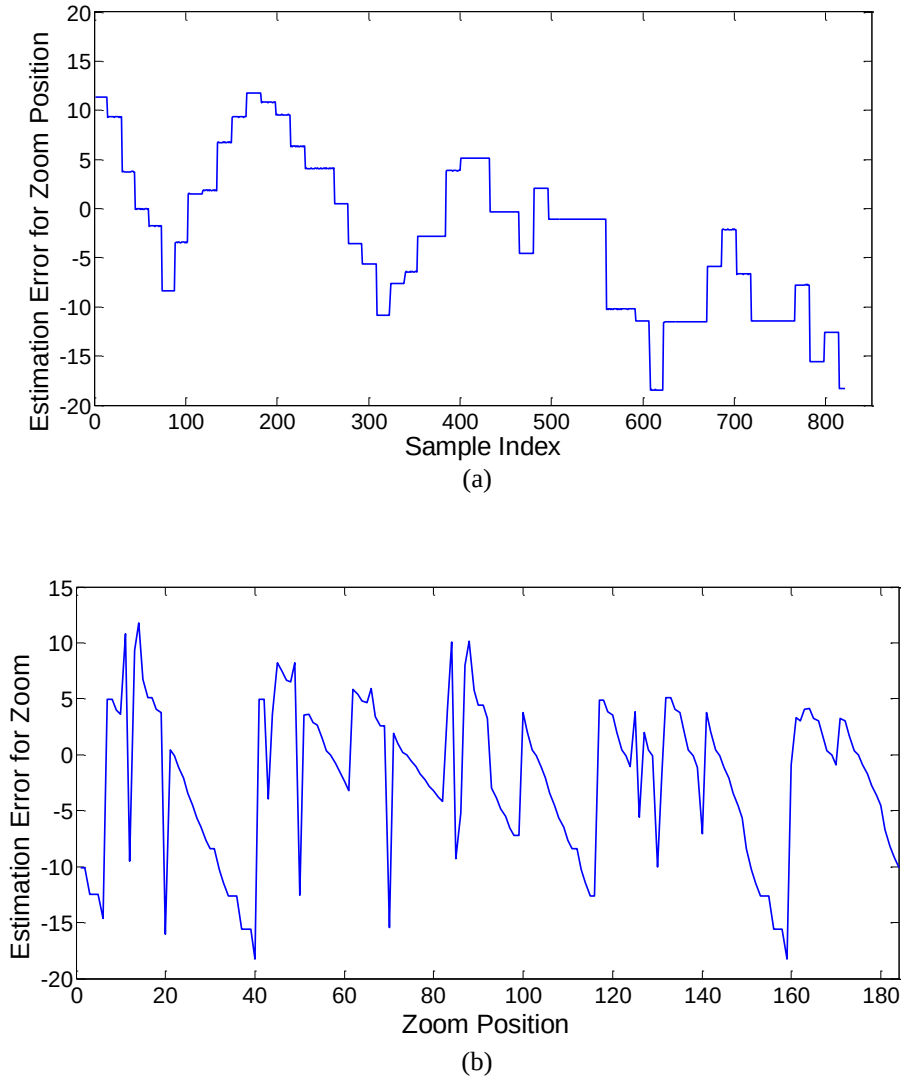
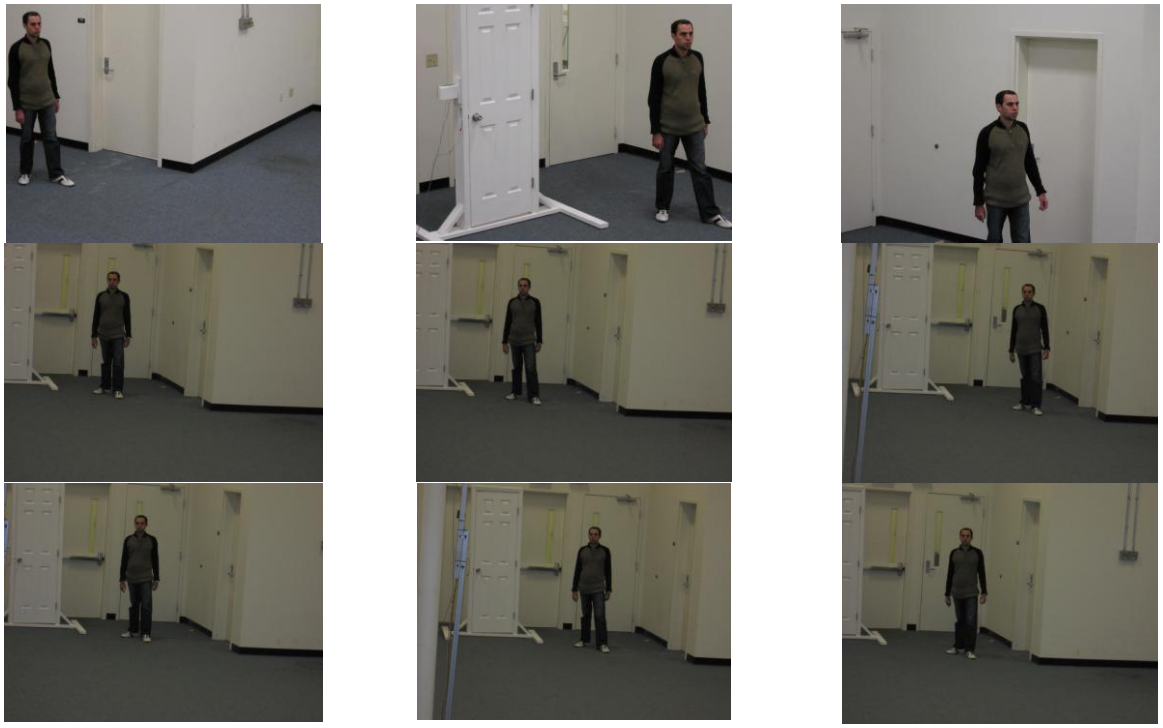


Figure 9.6 Estimation errors in zoom values: (a) comparison to the original sample set (825 samples), (b) relative zoom position (1 ~ 184).

Table 9.2 Comparison between our and reference algorithms

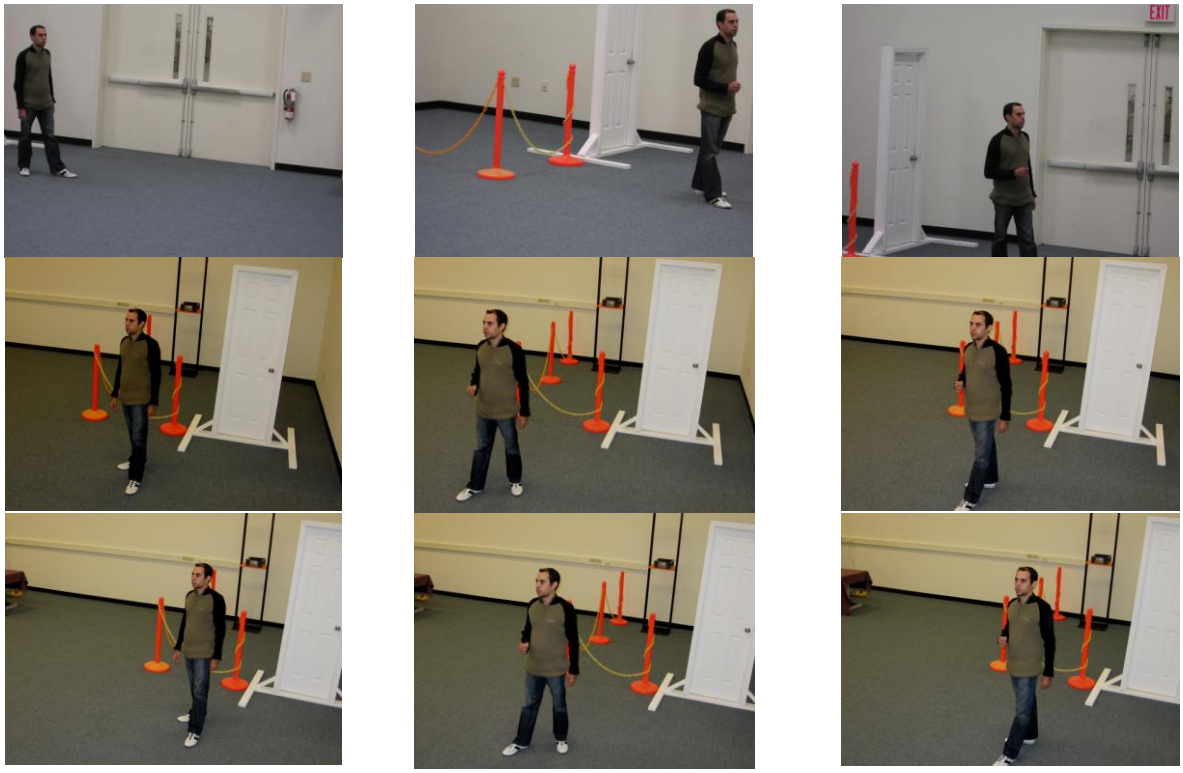
	Averaged Pixel Distance Deviation	***Averaged Pixel Size Deviation***
Our Method	11.1%	16.7%
Chen and Wang [Chen07A]	9.2%	15.2%



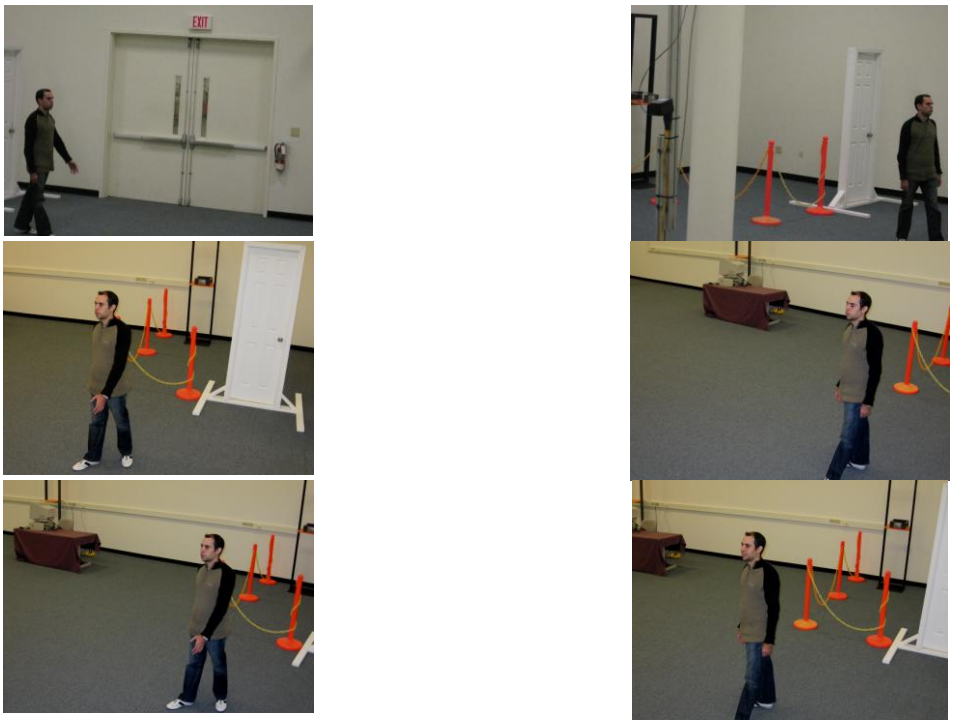
(a)

(b)

Figure 9.7 Performance of our proposed methods for a real-time multiple PTZ cameras system, when the single object is far away (18 meters) from the h^{th} PTZ camera (The tilt angle of the h^{th} PTZ camera is about 17^{o}).



(a)



(b)

Figure 9.8 Performance of our proposed methods for a real-time multiple PTZ cameras system, when the single object is close to (3 meters) the h^{th} PTZ camera (The tilt angle of the h^{th} PTZ camera is about 75^{o}).

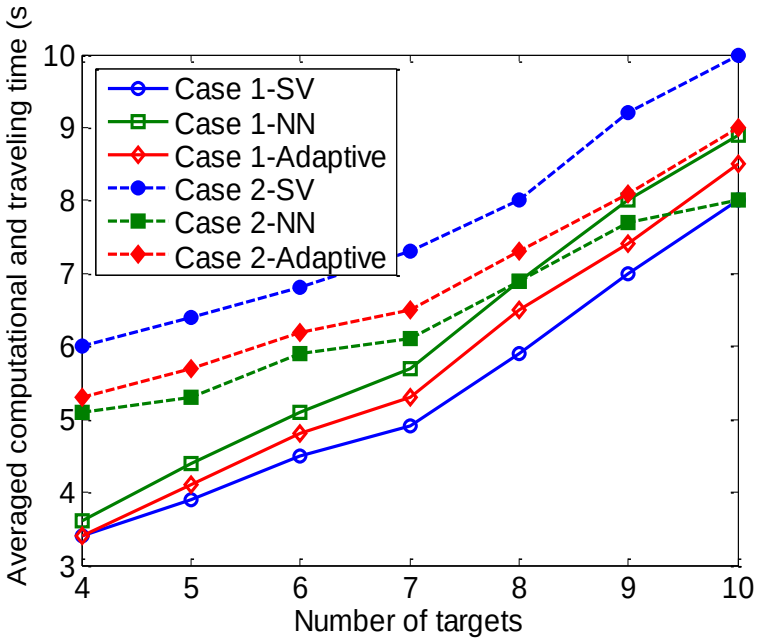
approach, in images of the h^{th} PTZ camera. In both examples, the averaged pixel distance deviations are 12.6% and 10.3% for our proposed, and Chen and Wang's methods, respectively. The averaged pixel size deviations are 14.6% and 12.7% for our proposed, and Chen and Wang's methods, respectively. Regardless of our proposed or Chen and Wang's methods, a consistent labeling approach is needed to identify the object of interest in both PTZ cameras after the occurrence of changing pose. Since this object of interest is maintained within the field of view of the h^{th} PTZ camera by both methods and maximal estimation errors for pan and tilt angles are 6.3^o and 8.5^o for our proposed method. Consistent labeling approaches can be carried out without added cost in here, because existing consistent labeling approaches such as scale-invariant feature transform (SIFT) [Lowe04] had been proved efficient when viewing angle is less than 50 degree. In other words, this slightly decreased pixel accuracy in our proposed approach has comparable result for the application of automated surveillance systems, as compared with Cheng and Wang's method. However, we reduce the dependence on the knowledge of intrinsic parameters of PTZ camera, thus holding the direct application to automated surveillance systems with changing configurations and a larger number of PTZ cameras.

9.3 Target hopping application

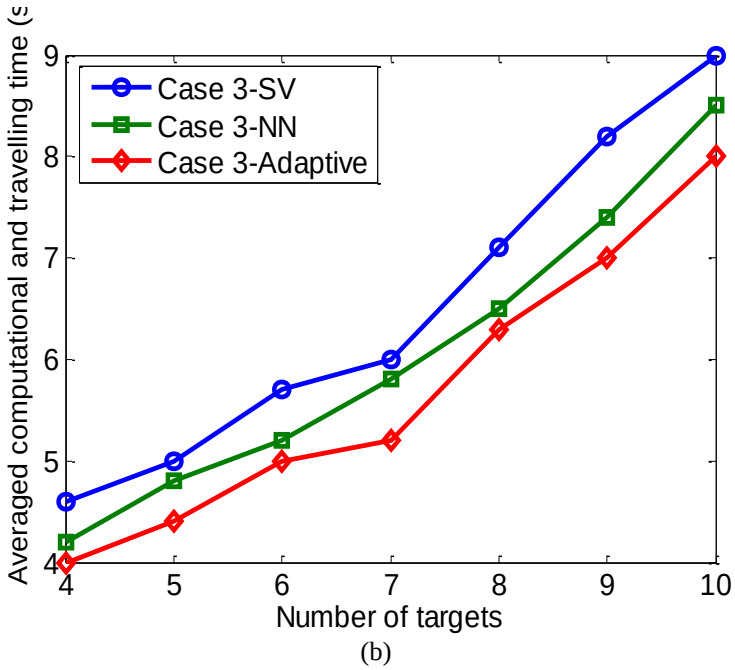
The performance of the sequential visiting, nearest neighbor, and our adaptive methods is compared based on the computational and traveling time. Three scenarios are tested:

- Case 1: targets only cluster within a small area of 5m×6m.
- Case 2: targets scatter all over an environment of 15m×6m.
- Case 3: targets are free to move in a 15m×6m environment.

Cases 1 and 2 are intentionally introduced to show the different performance of the sequential visiting and nearest neighbor methods, which validates our motivation to develop the adaptive method. In Figure 9.9 illustrates our experimental results. In Figure 9.9(a), we can see that the sequential visiting method requires less computational and traveling time for case 1 while the nearest neighbor method produces a better performance for case 2. Our adaptive algorithm is able to choose a scheme with less computational and traveling time according to the targets' dynamics. As expected, it yields a performance close to the better one for both cases. As for case 3 shown in Figure 9.9(b), the sequential visiting and nearest neighbor methods need more computational and traveling time. In comparison, our adaptive algorithm produces the best performance, which verifies the effective of the threshold derived in Equation (5.7). On average, our adaptive algorithm is able to save 10% of the averaged computational and traveling time. This is because our adaptive algorithm can choose the best approaches between the sequential visiting and nearest neighbor methods according to the current distribution of objects in the monitored scene. Figure 9.10 illustrates a real-time tracking sequence with multiple targets moving in the environment. Based on the detected motions in the omnidirectional camera, our adaptive algorithm chooses to visit multi40ple targets in the sequence of {4, 5, 1, 3, 2}. The adaptive algorithm favors the nearest neighbor method and saves 8.3% computational and traveling time to finish one visiting cycle compared with the sequential visiting method.

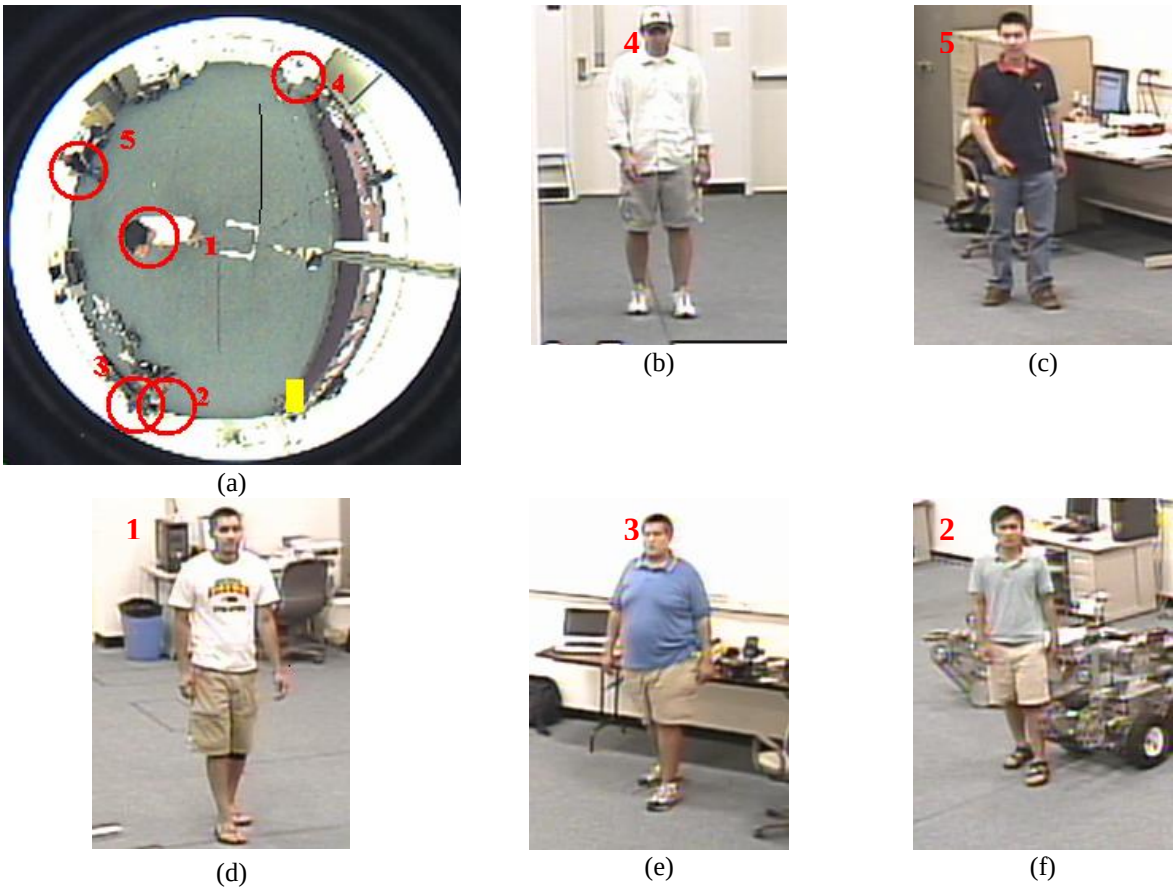


(a)



(b)

Figure 9.9 Averaged computational and traveling time for sequential visiting (SV), nearest neighbor (NN), and adaptive methods under different scenarios: (a) cases 1 and 2 and (b) case 3. In both case 1 and 2, our adaptive algorithm produces a performance closer to the one that requires less computational and traveling time. Case 3 resembles the practical surveillance application. Our adaptive algorithm requires the least computational and traveling time.



(a) (b) (c) (d) (e) (f)

Figure 9.10 A real-time sequence with the adaptive algorithm applied to generate the visiting sequence for the PTZ camera. (a) Detected multiple motions highlighted by red circles in the omnidirectional camera. The yellow square specifies the position of the PTZ camera. (b)-(f) Close-up observation of individual targets sorted in the sequence that the PTZ camera automatically selects and visits.

9.4 Camera handoff and determination of size of overlapped views

In our experiments, we first verify the effectiveness of our proposed camera handoff algorithm. For consistent labeling, our spatial mapping method is compared with the geometry-based and homography-based approaches in the spatial mapping stage, while our Wilcoxon Signed-Rank Test method is compared with the least square error approach in the pair matching stage. In themeantime, we show the individual effects of the two components, $S_{r,ij}$ and $S_{d,ij}$ defined in the observation measure by a real-time video sequence. To show the effectiveness of our proposed camera placement method, the algorithm presented by Erdem and Sclaroff [Erdem06] is implemented and used as a comparison reference. The performance of these two algorithms is compared in terms of coverage and handoff success rate.

$$\text{Handoff Sucess Rate} = \frac{\text{Number of Sucessfully Carried Out Handoff Requests}}{\text{Number of Handoff Requests}}. \tag{9.4}$$

9.4.1 Experiments on observation measure

In the following experiments, we study the behavior of the newly defined observation measure, as shown in Figures 9.11 and 9.12. , Figure 9.11 shows the plot of the corresponding S_{ij} values. In Figure 9.12, an omnidirectional camera is placed at $T_c = [\,0 \quad 3m\,]^T$. The image size is conventional 640×640. Points are uniformly sampled in the ground plane (Z=0) with X: -6m~6m and Y: 6m~6m. The normalization coefficient for the resolution component is given by $\alpha = \frac{6}{640} = 9.4\times10^{-3}$. Other parameters used are listed as follows: $\beta = 1$, $w_r = 0.25$, and $w_d = 0.75$. The best observation area is in the vicinity of [0, 0, 0]. As the object moves away from this area, the S_{ij} value decreases. In Figure 9.12, we illustrate the resolution $S_{r,ij}$ and distance to the boundary of camera's FOV $S_{d,ij}$ by a video sequence. As expected, $S_{r,ij}$ increases as the target moves toward one of the omnidirectional camera along the optical axis and $S_{d,ij}$ increases as the target moves toward the image center. In order to provide a clearer view of S_{ij} in a generic caseThe proposed observation measure gives a quantified measure of the tracking or observation suitability, which also agrees with our intuition and visual inspection.

9.4.2 Experiments on consistent labeling

In our indoor real-time surveillance environment with a dimension of 30m×15m×3m, two omnidirectional cameras (IQeye3) are placed 3m apart and used to fulfill target detection and object tracking. Background differencing and radial profile analysis [Cui98] are used for target detection and tracking. Omnidirectional images, with a resolution of 320×320, are obtained via an intranet connection with 4 frames per second. The performance of the proposed consistent

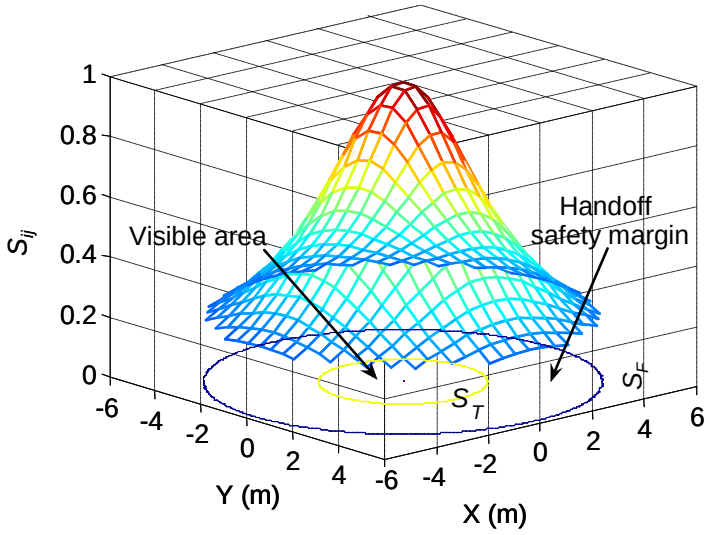


Figure 9.11 Graphical illustration of the observation measure and handoff safety margin for the omnidirectional camera.

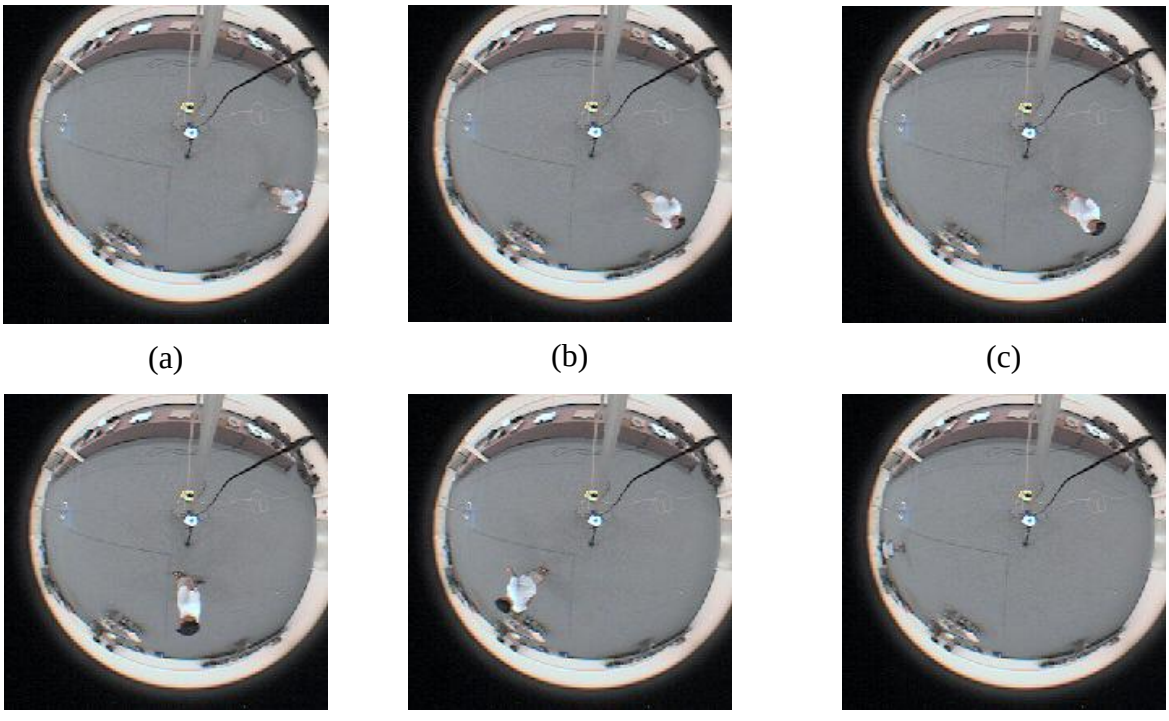


Figure 9.12 Illustration of the resolution and distance to the boundary of camera's FOV. (a) $S_{r,ij}$ = 0.35 and $S_{d,ij}$ = 0.36, (b) $S_{r,ij}$ = 0.54 and $S_{d,ij}$ = 0.59, (c) $S_{r,ij}$ = 0.74 and $S_{d,ij}$ = 0.79m, (d) $S_{r,ij}$ = 0.89 and $S_{d,ij}$ =0.88, (e) $S_{r,ij}$ = 0.59 and $S_{d,ij}$ = 0.6, and (f) $S_{r,ij}$ = 0.18 and $S_{d,ij}$ = 0.15.

labeling algorithm is compared with two reference algorithms: geometry-based method and Calderara's homography-based method [Calderara05]. For the geometry-based method, Zhang's calibration algorithm [Zhang00] is implemented to recover the 3D information of tracked objects by learning its intrinsic and extrinsic parameters and distortion model based on a total of eight images (800×800 pixels) of a planar checkerboard with forty-nine control points per image. During the spatial mapping phase, one single object moves around randomly in the environment when its relative coordinates are simultaneously collected by two IQeye3 cameras to derive the homography functions defined in Equation (5.7). From our experimental results, the homography functions are listed in Equation (5.13).

$$x_2 = -755.23 + 2.88x_1 + 0.03(x_1 - 225.7)^2 + 0.02(y_1 - 174.42)^2$$
$$\text{and} \qquad (9.5)$$
$$y_2 = -281.1 + 0.77x_1 + 1.2y_1 + 0.03(y_1 - 174.42)^2 + 0.02(y_1 - 174.42)(x_1 - 225.7)\,.$$

The polynomial model used in this paper is utilized to estimate model parameters in situations that error terms, ε_R and ε_C, are normally distributed and that variance of the error terms does not depend on the value of any independent variables. Generally, assessments of the validity of our polynomial model assumptions are based on analyses of residuals, the differences between the observed and predicted values of the response variable. Data points with unusually large residuals may be outliers that indicate that something went wrong when the derived model was made. In other words, error terms, ε_R and ε_C, are not normally distributed or variance of the error terms depend on the value of any independent variables. This can be caused by the nature of collected data. The root mean squared error (RMSE) between the observed and predicted response variables along *x* axis is in between 10 and 17 pixels in our data set. It is calculated by $\sqrt{(\hat{x}_{2i} - x_{2i})^2}$ where $\hat{x}_{2i}$ and x_{2i} represent the i^{th} predicted and observed response variables, corresponding to a maximally relative error of 5.3% ($\frac{17}{320} = 5.3\%$) when normalized with respect to the image width. Similarly, the RMSE between the observed and predicted response variables along *y* axis is in between 8 and 14 pixels, corresponding to a maximally relative error of 5.7% ($\frac{14}{320} = 5.7\%$) when normalized with respect to the image height. It does not appear unusually large residuals according to the work of D. Wackerly et al [Wackerly02]. Thus, it validates the accuracy of this selected model.

In order to understand how the Wilcoxon Signed-Rank Test improves the accuracy of pair matching as compared with the least square error approach in conjunction with different calibration methods, we conduct experiments where five people walk randomly in the tested indoor environment at a speed less than 4km/hour. Table 9.3 specifies the methods used in the calibration and pair matching stages for each tested case. For example, case C employs the geometry-based approach and the least square error approach for calibration and pair matching, respectively.

Figure 9.13 shows the success rate of consistent labeling with respect to the number of frames used for pair matching. In general, we can see that the success rate of consistent labeling increases as the number of frames used for pair matching increases for all tested cases. Case C

Table 9.3 Specification of tested cases for the study of the performance of various consistent labeling methods

	Tested cases	A	B	C	D	E
Spatial Mapping	Our method	×			×	
	Geometry-based approach		×	×		
	Calderara's method [Calderara05]					×
Pair matching	Our method	×	×			
	Least square error			×	×	×

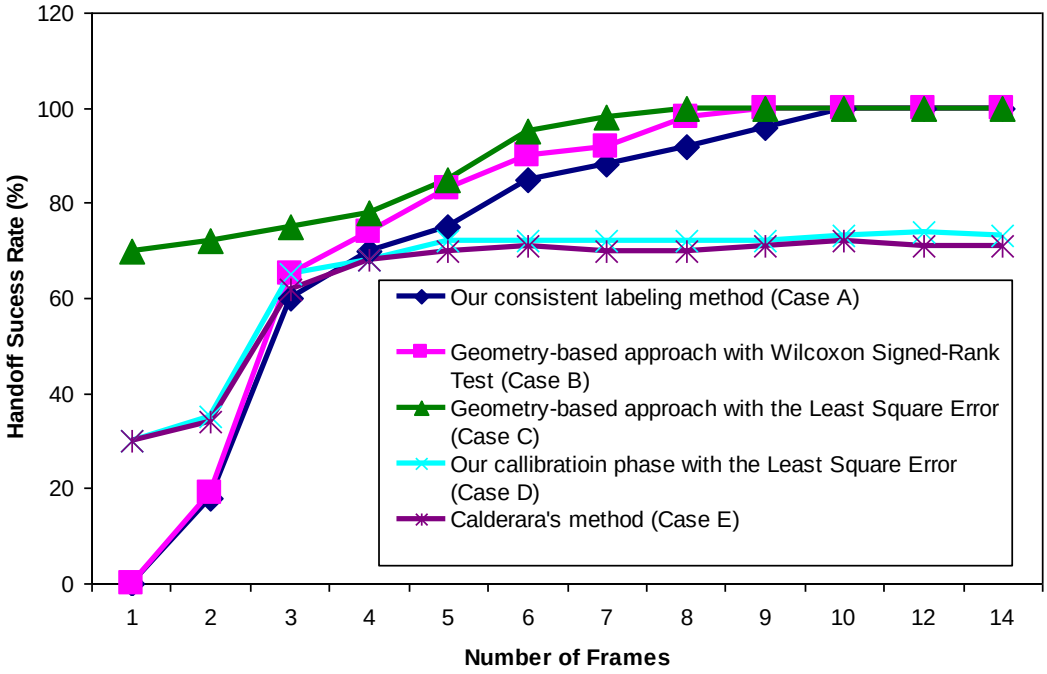


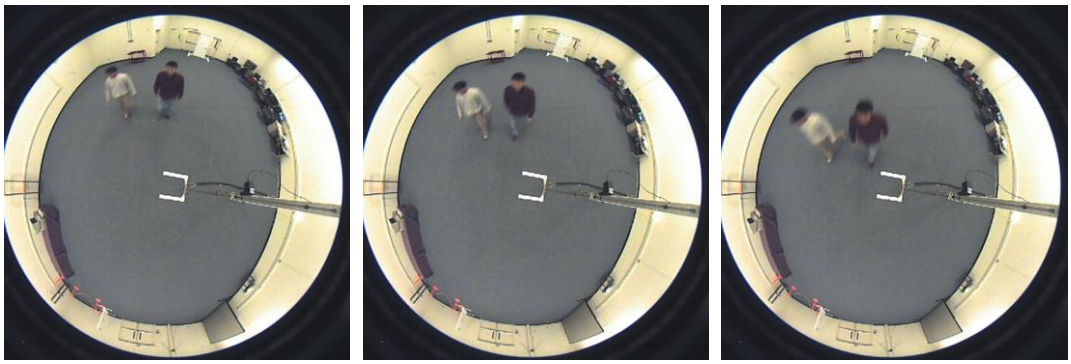
Figure 9.13 Performance of various consistent labeling methods based on the success rate of consistent labeling versus the number of frames used.

case 3, two objects are walking in the same direction within three joint t omnidirectional views.The threshold S_T is 0.3 to comply with the time needed for executing camera handoff has the highest success rate of consistent labeling (71%) when only one frame is used, because it can recover the 3D position of the tracked object with a better accuracy. Our method (case A) can achieve a higher success rate of consistent labeling than 90% if at least eight frames are used and yields a similar performance as case B. Since Calderara's method uses the least square error method to match pairs and is inefficient in finding pixel-to-pixel correspondence between two omnidirectional camera images, our method (case A) outperforms it when more than three frames are used to match pairs. Moreover, the Wilcoxon Signed-Rank Test used in the case A improves the success rate of consistent labeling in the case D where the least square error approach is used, which validates the effectiveness of the Wilcoxon Signed-Rank Test. However, when only one frame/trial datum is collected, the Wilcoxon Signed-Rank Test cannot rank the absolute value of no difference. As a result, its success rate of consistent labeling is approximately zero. However, the more trail data are collected, the higher the success rate of consistent labeling will turn to. In conclusion, even though geometry-base approach (case C) can reach a 90% success rate of consistent labeling with five frames, which is only three frames less than what our proposed consistent labeling method needs (case A), it needs an expensive procedure to calibrate each camera, which is almost impractical in real-time surveillance.

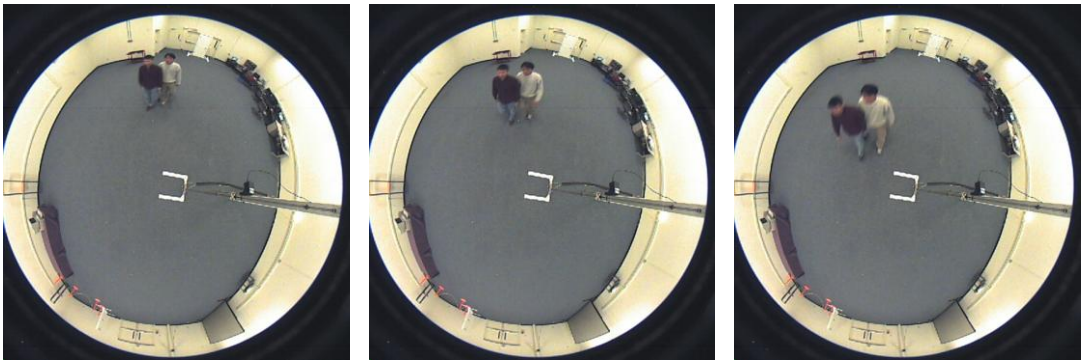
The performance of our proposed consistent labeling method relies on the Wilcoxon Signed-Rank Test to compensate for the imperfection of homography functions derived by multiple regression models. This method is not only computationally efficient ($O(n)$ where n is the number of observations/frames), but robust to varied proximity and trajectory difference between objects. The number of observations/frames used in the test is mainly related to the proximity and trajectory difference between objects. In general, the bigger the proximity and trajectory difference between objects are, the less frames are needed. Figure 9.14 illustrates two example levels of the proximity and trajectory difference between objects at which the Wilcoxon Signed-rank test can and cannot tolerate. In Figure 9.14(a), even though two objects are close to each other and their trajectories are identical, the Wilcoxon Signed-Rank Test is still capable of distinguishing those two objects across cameras. However, in Figure 9.14(b), when two objects are within 30cm in ground space, causing partial occlusions, and their trajectories are identical, the Wilcoxon Signed-Rank Test is incapable of distinguishing those two objects before they fall out of the FOV of the observing camera. This is cased by the tie of T, which is a statistic to test whether or not the two populations are identical. In our current setting, we continue to collect more frames to make a final decision. Consistent labeling may fail when the tie of T is not resolved before the objects fall out of the FOV of the observing camera. In the future work, a compensation method will be investigated.

9.4.3 Experiments on camera handoff

To clearly exhibit how a camera handoff is triggered, three cases are illustrated in omnidirectional cameras environment where two omnidirectional cameras are used in the first two case and three omnidirectional cameras are used in the third case. In case 1, two objects are walking in an opposite direction. In case 2, four objects are walking in he same direction. In



(a)



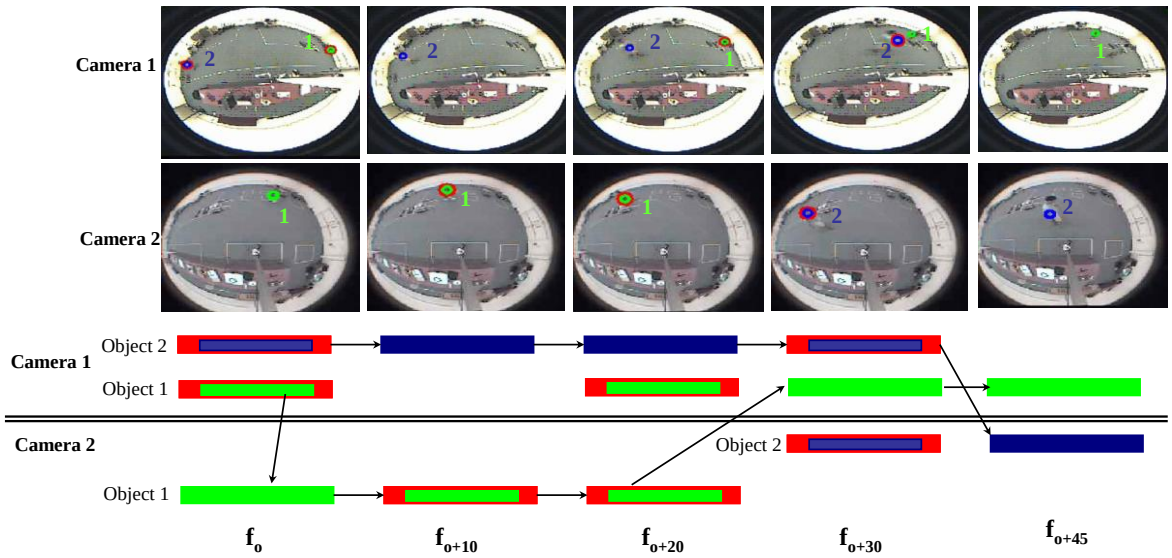
(b)

Figure 9.14 Performance illustration of Wilcoxon Signed-Rank Test. (a) two objects are close to each other and their trajectories are identical. (b) two objects are within 30cm in the ground space, causing partly occluded, and their trajectories are identical.

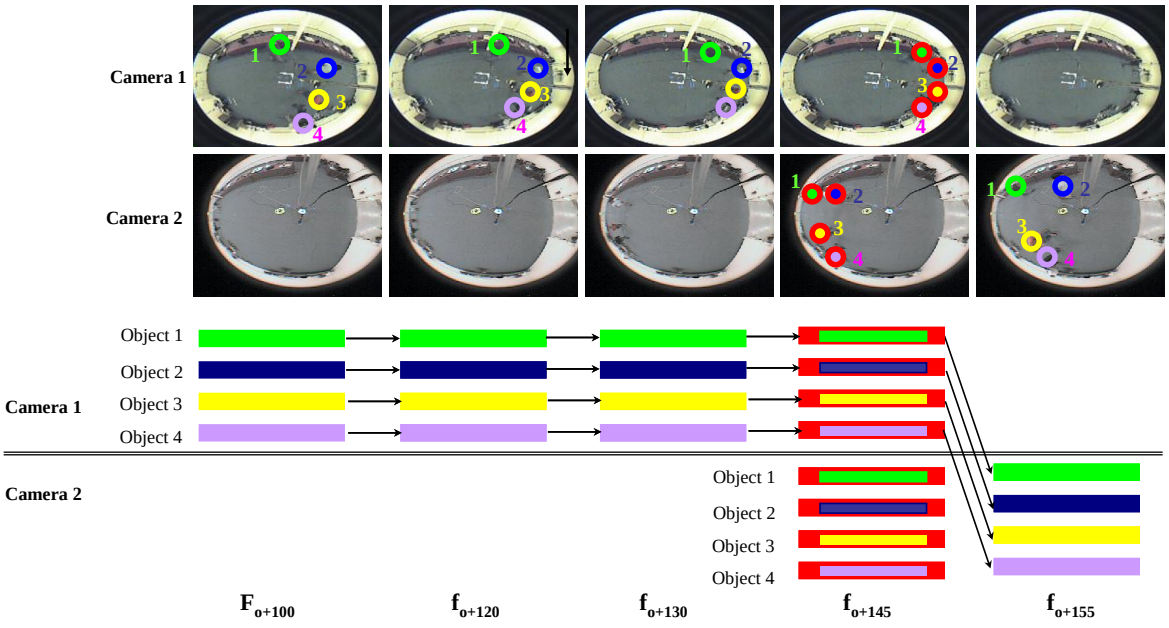
(10 frames) and the maximal moving speed of the objects (0.6 meters per second). Figure 9.15(a) and (b) show the sampled frames for the first two cases, respectively. In both cases, solid green, blue, yellow, and purple circles/rectangles represent object 1, 2, 3, and 4 respectively. Solid green, blue, yellow, or purple circles/rectangles with red circles or rectangles outside indicate thisbject is under camera handoff. Figure 9.16, shows the sampled frames for the third case. In this case, solid green and blue circles/rectangles represent object 1 and 2. Solid green and blue circles/rectangles with red circles or rectangles outside indicate this object is under camera handoff.

In Figure 9.15(a), since both objects are close to the EFOV of camera 1 (S_{11} = 0.2 and S_{21} =0.2) in frame f_0, they are under camera handoff process and camera two is only capable of tracking object 1. However, since object 2 is not seen by camera 2, it is tracked by camera 1 u40 ntil it becomes untraceable. From frame f_{0+10} to f_{0+20}, object 2 is no long under camera handoff process and object 1 (S_{12} = 0.22) is under camera handoff process since it is close to EFOV of camera 2. In frame f_{0+30}, object 1 is handed over to camera 1 and object 2 (S_{21} = 0.29) is under camera handoff process. In frame f_{0+45}, object 1 and object 2 are tracked by camera 1 and 2 respectively. In Figure 9.15(b), four objects are tracked by camera 1 from frame f_{0+100} to f_{0+130}. In frame f_{0+145}, because four objects are close to the EFOV of camera 1 and their resolutions are deteriorating (S_{11} = 0.27, S_{21} =0.2, S_{31} = 0.2, and S_{41} =0.6), they are under handoff process. In frame f_{0+155}, the handoff process is carried out successfully for each object and the four objects are tracked by camera 2.

In Figure 9.16, both objects are tracked by camera 1 from frame f_0 to frame f_{0+30}, but they are moving toward the EFOV of camera 1. In frame f_{0+40}, since they are close to the EFOV of camera 1 (S_{11} = 0.19 and S_{21} =0.21), they are under camera handoff process and both camera 2 and 3 are capable of tracking both objects. However, according to our criterion indicated in Section 3, the highest observation measure $j^* = \arg\max_{j'}\{S_{ij'}\}$ is selected as the most appropriate camera to take over the i^{th} object in the pool of candidate cameras j'. Thus, object 1 is assigned to camera 2, because it has the highest observation measure in camera 2 (S_{12} = 0.68 and S_{13} =0.28). Object 2 is assigned to camera 3, because it has the highest observation measure in camera 2 (S_{22} = 0.29 and S_{23} =0.8). In conclusion, our proposed consistent labeling algorithm can perform as accurately as the geometry-based approach without tedious calibration processes and outperform Calderara's homography-based approach. In the meantime, our observation measure can quantitatively formulate the effectiveness of a camera in observing the tracked object, so that camera handoff can smoothly transfer objects for automated and persistent object tracking. In addition, our system design follows the distributed approach, where cameras only exchange information with adjacent cameras. Usually, one camera communicates with 2 to 4 other cameras, which is optimized by camera placement. As the scale of the camera network increases, it is always doable to divide the whole network into several subnets, where one camera communicates with limited number of adjacent cameras. Therefore, due to the distributed nature of our system, computations are carried out in each subnetwork independently. In this sense, our system readily adopts parallel computations



(a)



(b)

Figure 9.15 The illustration of camera handoff procedure in a real time system with two cases. Solid green, blue, yellow, and purple circles/rectangles represent tracked object 1, 2, 3, and 4 respectively. Solid green, blue, yellow, or purple circles/rectangles with red circles or rectangles outside indicate this object is under camera handoff. (a) Case 1: two objects are walling in an opposite direction. (b) Case 2: four objects are walking in the same direction.

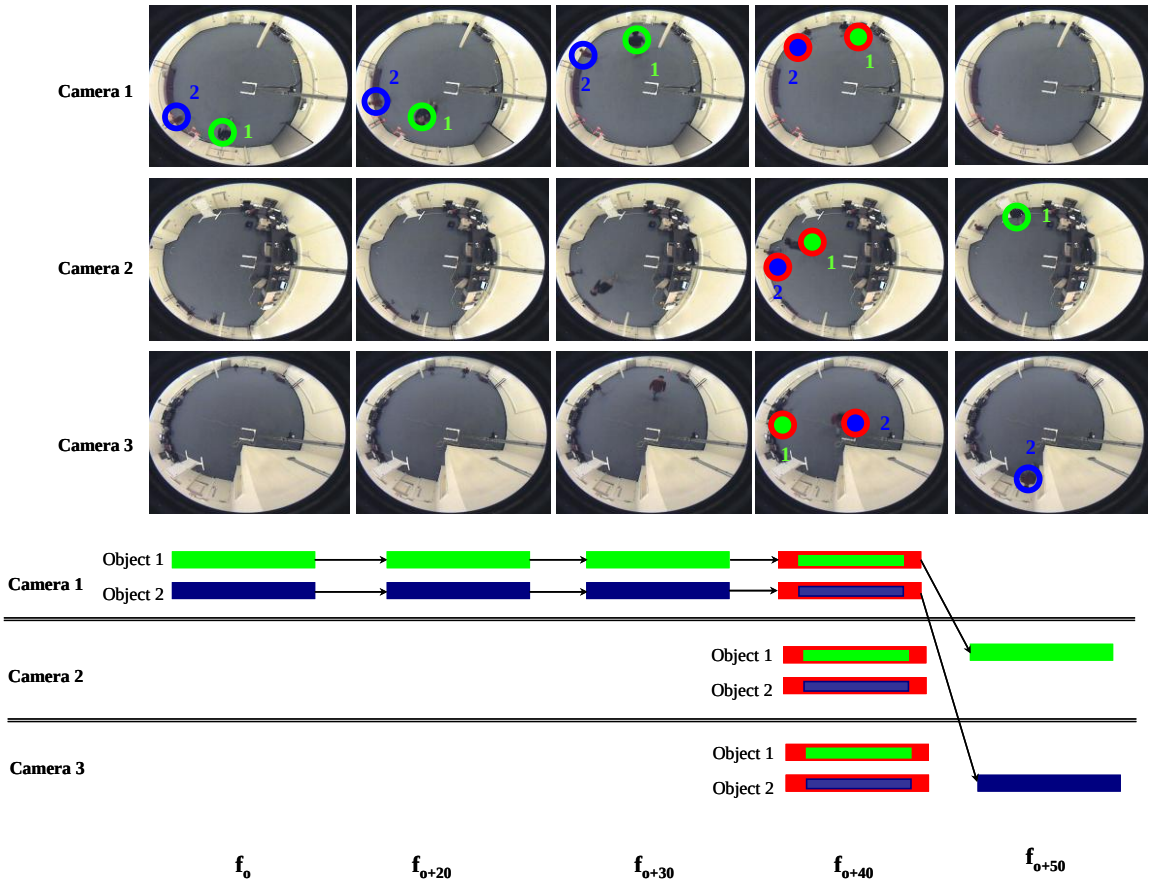


Figure 9.16 The illustration of camera handoff procedure in a real time system with two objects walking in the same direction within three joint ominidirectional views. Solid green and blue circles/rectangles represent tracked object 1, and 2 respectively. Solid green and blue circles/rectangles with red circles or rectangles outside indicate this object is under camera handoff.

9.4.4 Experiments on determination of size of overlapped views

In this section, we conduct camera placement experiments in two indoor floor plans. Our proposed camera placement method is compared with the reference algorithm proposed by Erdem and Sclaroff [Erdem06]. The floor plans under the test are shown in Figures 9.17 (a) and (b), which are originally from [Yao08] The floor plan in Figure 9.17(a) represents two types of indoor area encountered in practical surveillance: space with obstacles (region A illustrated in yellow) and open space where pedestrian can move freely (region B illustrated in green). Region B is deliberately included since it imposes more challenges on camera placement when considering the handoff success rate. Figure 9.17(b) illustrates an environment with a predefined path where workers proceed in a predefined sequence.

To obtain a statistically valid estimation of handoff success rate, simulations are carried out to enable a large number of tests under various conditions. The work of Antonini et al. [Antonini06] for pedestrian behavior simulator is implemented so that we could have a close resemblance to the experiments in real environments and in turn an accurate estimation of the handoff success rate. In our experiments, the arrival of the pedestrian follows a Poisson distribution with an average arrival rate of 0.05 (person/second). The average walking speed is 0.5 (meters/second). 300 pedestrian traces are randomly generated for our simulation. Several points of interest are generated randomly to form a pedestrian trace.

Figures 9.18 and 9.19 show optimal camera arrangements for two indoor floor plans shown in Figure 9.17(a) and 9.17 (b) respectively. In Figure 9.18, at the cost of slight decrease in coverage, the handoff success rate significantly increases from 52.8% to 79.0%. In Figure 9.19, the similar result exists. The handoff success rate increases from 50% to 92.6% at the cost of slight decrease in coverage from 92.1% to 81.5%. Our experiment validates the importance of reserving sufficient cameras' overlapped FOVs for improving the overall performance of the automated surveillance system in terms of the handoff success rate. Our proposed camera placement method exhibits a significant increase in the camera handoff success rate at the cost of slightly decreased coverage, as compared to Erdem and Sclaroff's method without considering the necessary overlapped FOVs.

Figure 9.20 illustrates the effect of the case shown in Figure 9.19 in a real-time system. ΔR_a and ΔR_b are 10m and 7m respectively. In this experiment, the threshold S_T is 0.3 to comply with the time needed for executing camera handoff (10 frames) and the maximal moving speed of the objects (0.6 meters per second). In Figure 9.20(a), positions of two omnidirectional cameras are determined by Erdem and Sclaroff's method. We can see that the object is tracked by camera 1 from frame f_{0+10} to f_{0+20}. In frame f_{0+30}, since the object is close to EFOV of camera 1 and its resolution is deteriorating (S_{11} = 0.24), it is under handoff process. However, since the size of overlapped FOV is not large enough, camera 2 cannot track the handoff object with enough resolution even in frame f_{0+40}. As a result, camera handoff fails and the track of the object is lost. Figure 9.20(b) illustrates a similar scenario with a camera placement generated from our method. As expected, camera handoff is successfully carried out from f_{0+30} to f_{0+40}, because the size of overlapped FOV is optimized. The object of interest is tracked continuously across two cameras.

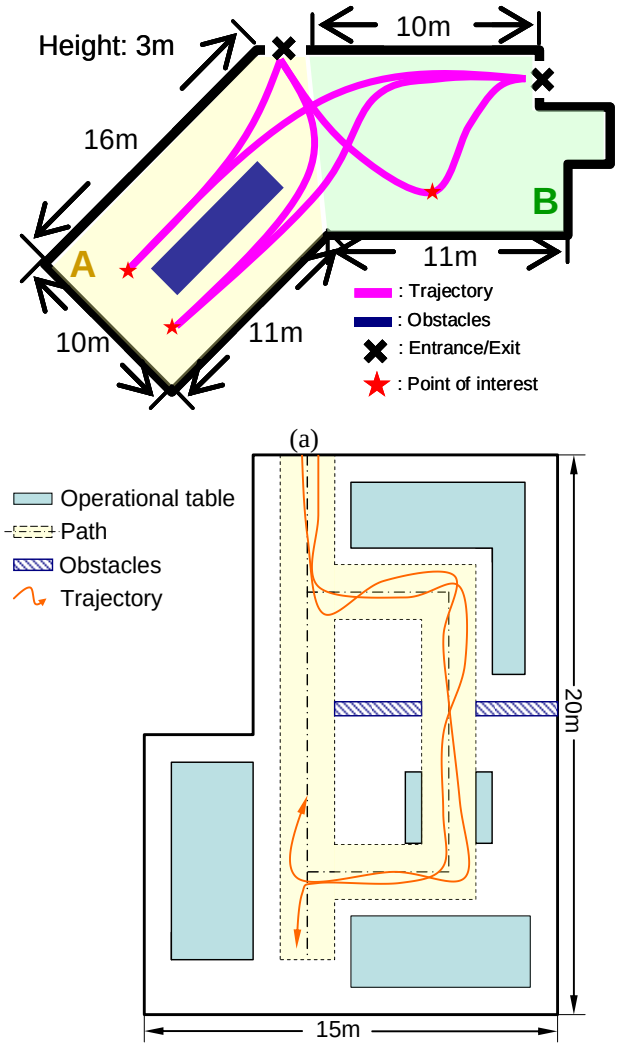
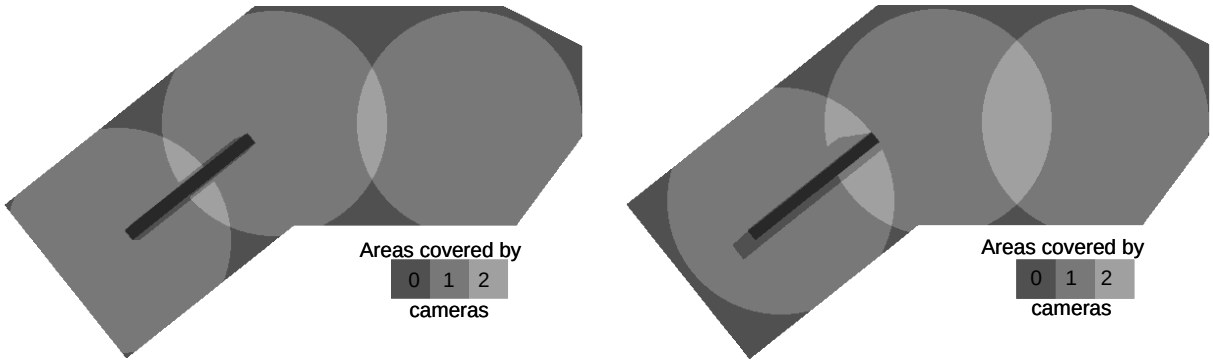
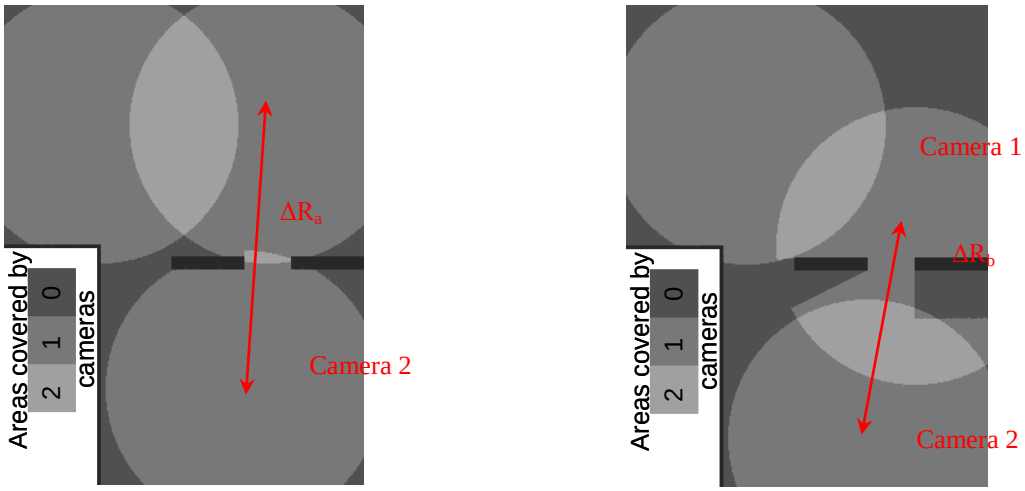


Figure 9.17 Illustration of the two indoor floor plans (a) and (b).



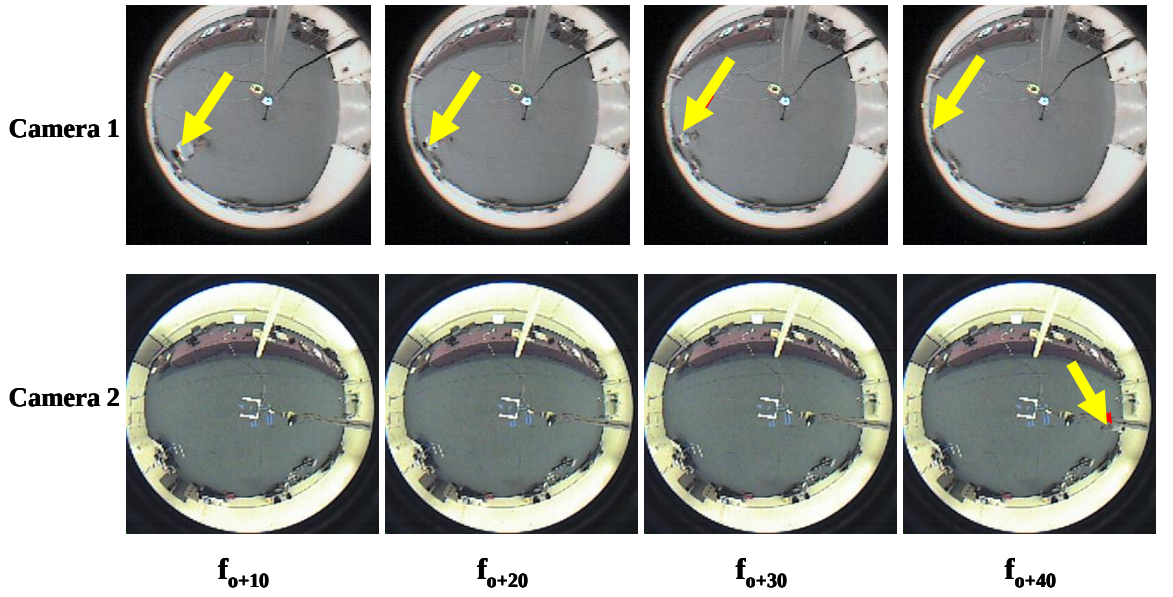
(a) (b)

Figure 9.18 Optimal camera positioning of the first indoor floor plan using omnidirectional cameras (a) Erdem and Sclaroff's method (Coverage: 88.4 % and Handoff Success Rate: 52.8%) and (b) our method (Coverage: 86.0% and Handoff Success Rate: 79.0%).

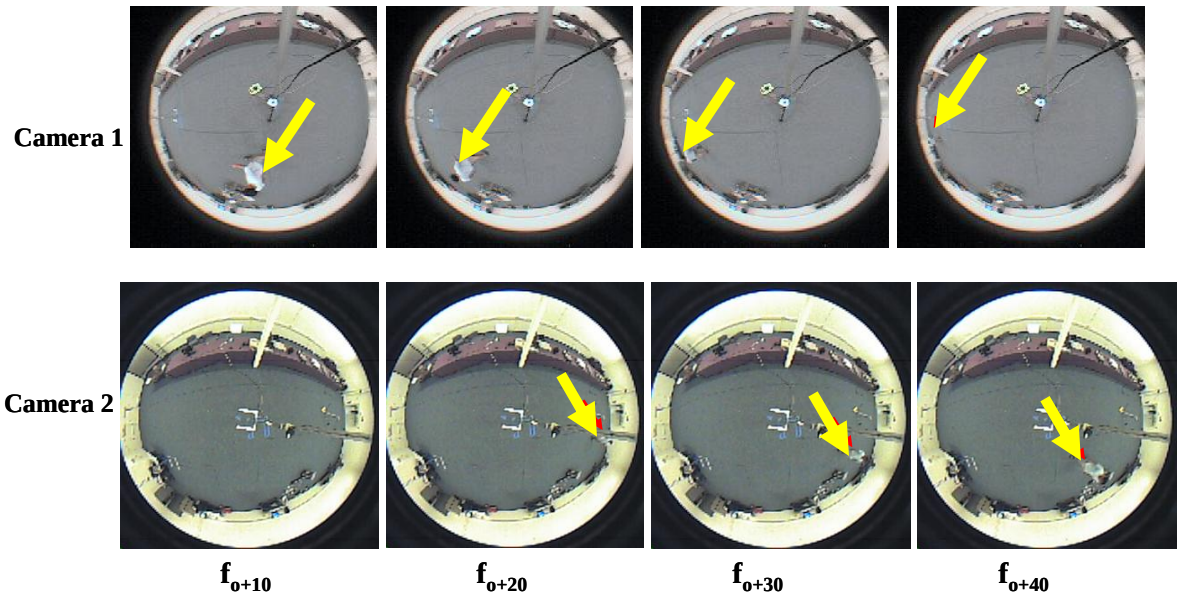


(a) (b)

Figure 9.19 Optimal camera positioning of the second indoor floor plan using omnidirectional cameras (a) Erdem and Sclaroff's method (Coverage: 92.1% and Handoff Success Rate: 50%) and (b) our method (Coverage: 81.5% and Handoff Success Rate: 92.6%).



(a)



(b)

Figure 9.20 Illustration of the effect of two camera placement methods on consistent labeling in a real time system (a) Erdem and Sclaroff's method. In frame f_{0+30}, since the object is close to EFOV of camera 1 and its resolution is deteriorating, it is under handoff process. However, since the size of overlapped FOV is not large enough, camera 2 cannot track the handoff object with enough resolution even in frame f_{0+40}. As a result, camera handoff fails and the track of the object is lost. (b) our method. Camera handoff is successfully carried out from f_{0+30} to f_{0+40}, because the size of overlapped FOV is optimized. The object of interest is tracked continuously across two cameras.

9.5 Adaptive resource management

In this section, we study the individual and combined effects of the three components, M_S, M_D, and M_O, defined in the trackability measure. Afterwards, experiments are conducted to verify the effectiveness of our proposed camera handoff algorithm via video sequences generated by ourselves and dataset S7 in PETS' 2006 [PETS06]. Figure 9.21 shows the floor plan of the experimental environment. The camera placement is optimized using a modified Erdem and Sclaroff's method [Erdem06]. Static perspective cameras with a resolution of 640×480 are placed along the walls at a height of 3m with a tilt angle θ_T of -30°. Two priority levels are assigned to the objects, N_{pr}=2. The maximum number of objects that can be tracked simultaneously is three for all cameras, N_{max} = 3 in our case. The thresholds T_O, T_D, and T_S are 0.2 to comply with the time needed for executing camera handoff (5 seconds average) and the maximal moving speed of the objects (0.6 meters per second). The discrepancy between the maximum number of tracked objects in Figure 7.1(b) and in our experiment is that the surveillance system in our experiment includes behavioral understanding in addition to multiple object tracking algorithm. The behavioral understanding part is necessary for assigning different priorities to tracked objects. As a result, the surveillance system illustrated in our experiment can only sustain at most three tracked objects without deteriorating the system's frame rate. In other words, to generate Figure 7.1(b), the system only includes multi-object tracking. Thus, it can monitor 10 objects without deteriorating the frame rate. This observation also exemplifies the importance of resource management in a real-life scenario. Since the focus of this paper is not developing object tracking and consistent labeling algorithms, we use existing algorithms for multi-object tracking and consistent labeling. Image difference and homography-based approaches are implemented for object tracking and consistent labeling, respectively.

9.5.1 Experiments on trackability measure

From the definition of the trackability measure, we first study the individual effect of M_S, M_D, and M_O based on real-time tracking system where camera 2 indicated in Figure 9.21 is used in this experiment. According to the derivation introduced in Equations (7.4) and (7.5), we notice that the components M_S and M_D mainly describe the variations along and orthogonal to the camera's optical axis, respectively. As expected, in Figures 9.22 and 9.23, M_S increases as the target moves toward the camera along the optical axis and M_D increases as the target moves toward the image center. In Figure 9.24, two targets walk diagonally across the camera's FOV with the same direction at different speeds. As a result, the relative distance between them decreases. This variation is indicated by a decreased M_O, as shown in Figure 9.24.

Figure 9.25 illustrates sampled frames at f_n and f_{n+15} from a real-time tracking sequence 1 with two static perspective cameras. The cameras' positions are specified in Figure 9.21 as camera 1 and 2. Table 9.4 lists $M_{S,ij}$, $M_{D,ij}$, and $M_{O,ij}$ for the i^{th} object observed by the j^{th} camera at frames f_n and f_{n+15}, where i ranges from 1 to 5 and j is either 1 or 2. Figures 9.26, 9.27, 9.28, 9.29, and 9.30 illustrates continuous trackability measures, $M_{S,ij}$, $M_{D,ij}$, and $M_{O,ij}$, of objects 1, 2, 3, 4, and 5 from frame f_n to f_{n+20} in real-time tracking sequence 1. In frame f_n, object 4 is blocked by

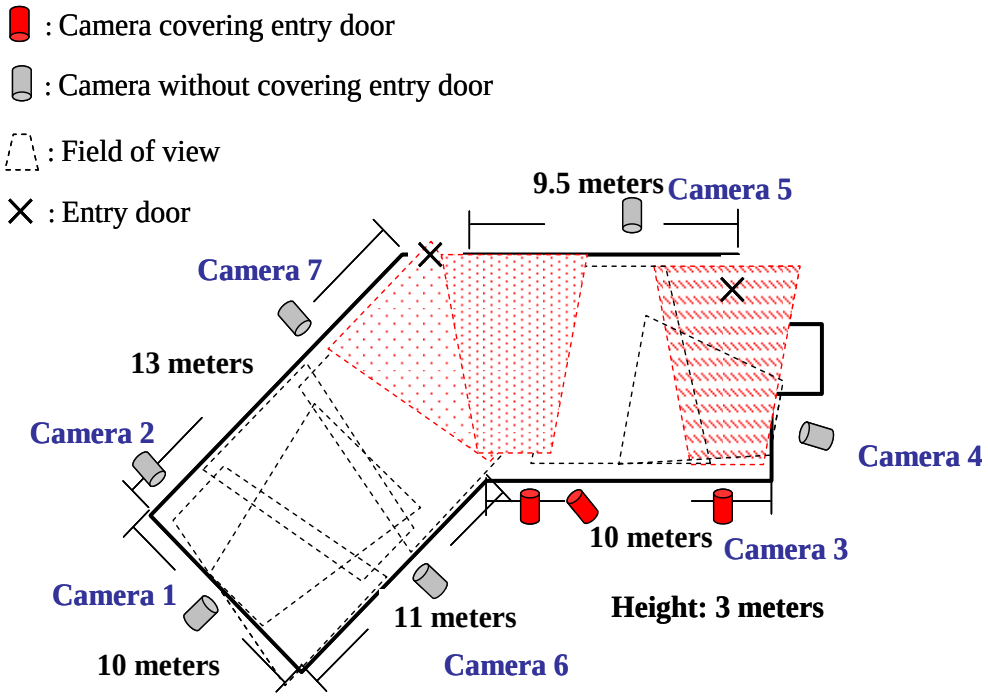
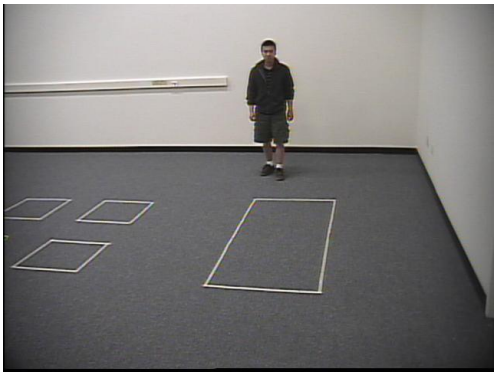
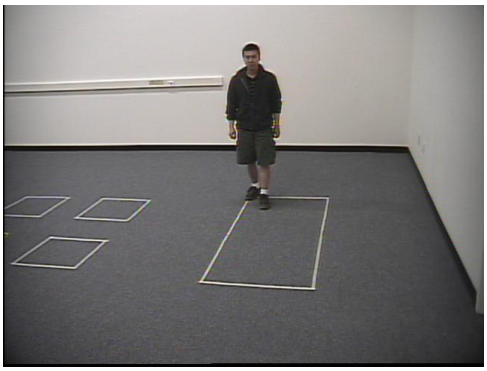


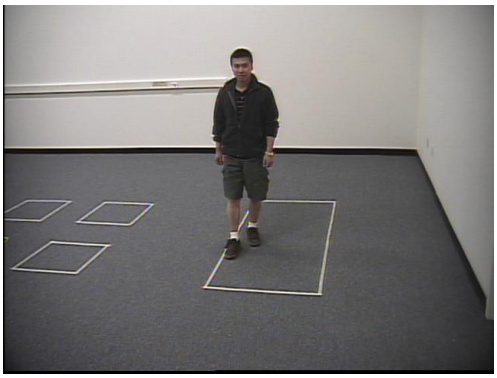
Figure 9.21 Floor plan of the experimental environment.



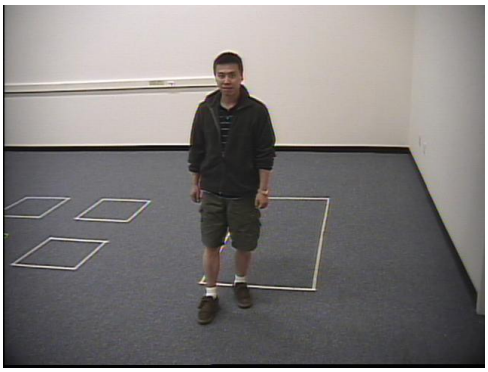
M_S = 0.16



M_S = 0.22

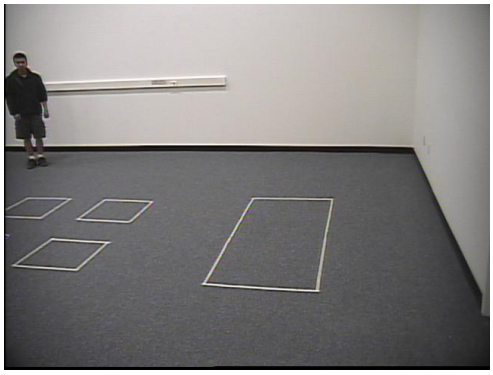


M_S = 0.57

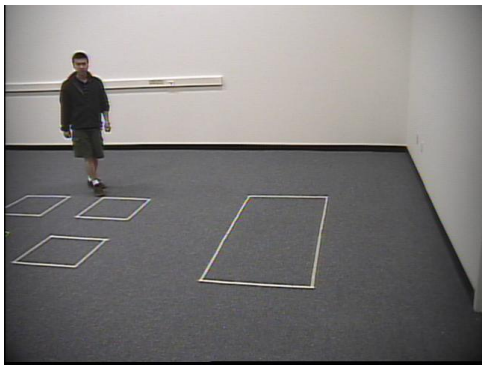


M_S = 0.76

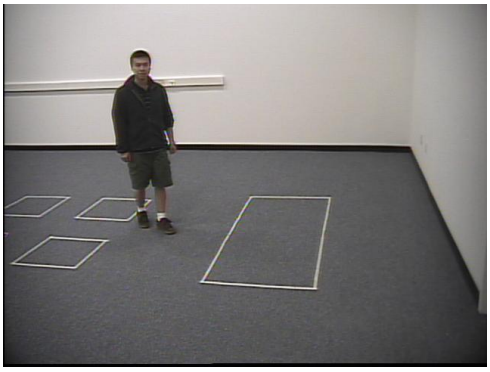
Figure 9.22 The computed resolution component M_S from frames acquired by a real-time tracking system as the object of interest moves toward the camera along the optical axis.



M_D = 0.11



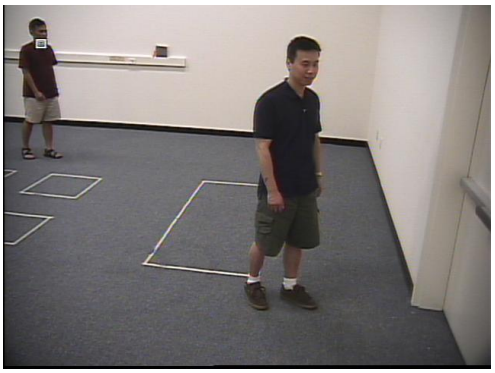
M_D = 0.22



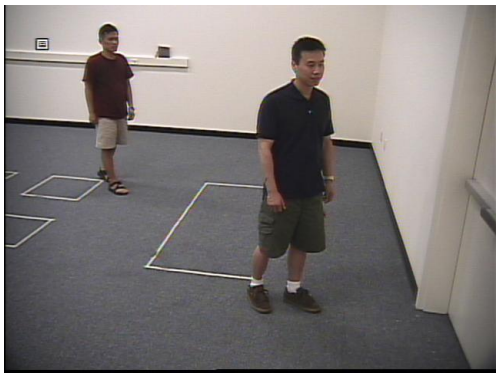
M_D= 0.4

M_D = 0.77

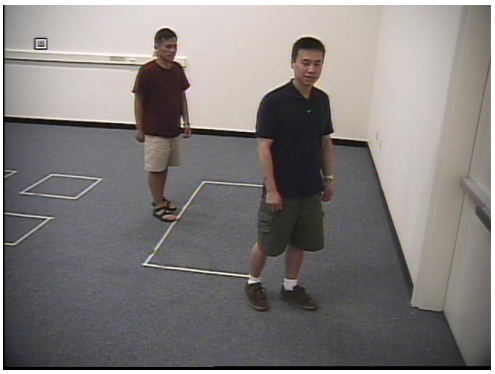
Figure 9.23 The computed distance component M_D from frames acquired by a real-time tracking system as the object of interest moves toward the image center.



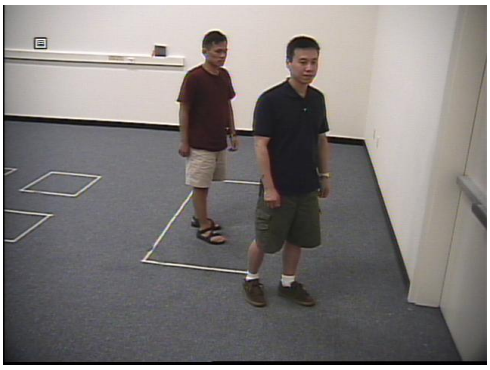
M_O = 0.82



M_O = 0.47



M_O = 0.35



M_O = 0.24

Figure 9.24 The computed occlusion component M_O from frames acquired by a real-time tracking system. Two objects move across the camera's FOV at different speeds, resulting in a decreased relative distance between them.

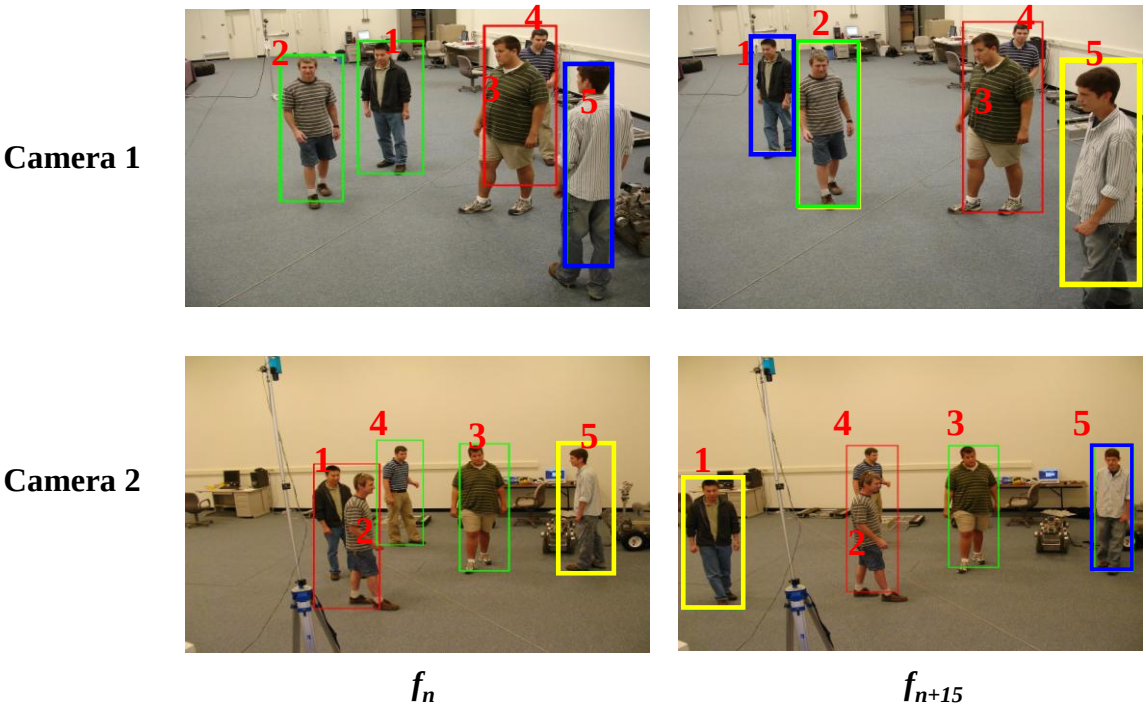


Figure 9.25 Illustration of the effectiveness of our proposed trackability measure in the camera handoff procedure at sampled frames f_n and f_{n+15} in real-time tracking sequence 1.

Table 9.4 The illustration of $M_{O,ij}$, $M_{D,ij}$, and $M_{S,ij}$ shown in Figure 9.25.

		Object 1 (*i*=1)		**Object 2 (*i*=2)**		**Object 3 (*i*=3)**		**Object 4 (*i*=4)**		**Object 5 (*i*=5)**	
		f_n	f_{n+15}	f_n	f_{n+15}	f_n	f_{n+15}	f_n	f_{n+15}	f_n	f_{n+15}
Camera 1 (*j*=1)	$M_{O,ij}$	0.31	0.15	0.41	0.15	0	0	0	0	0.18	0.25
	$M_{D,ij}$	0.6	0.5	0.5	0.4	0.45	0.5	0.43	0.4	0.38	0.14
	$M_{S,ij}$	0.43	0.41	0.41	0.42	0.43	0.42	0.3	0.3	0.45	0.6
Camera 2 (*j*=2)	$M_{O,ij}$	0	0.25	0	0	0.6	0.5	0.24	0	0.5	0.6
	$M_{D,ij}$	0.6	0.15	0.6	0.6	0.9	0.6	0.85	0.56	0.43	0.15
	$M_{S,ij}$	0.42	0.41	0.43	0.41	0.42	0.42	0.41	0.41	0.40	0.41

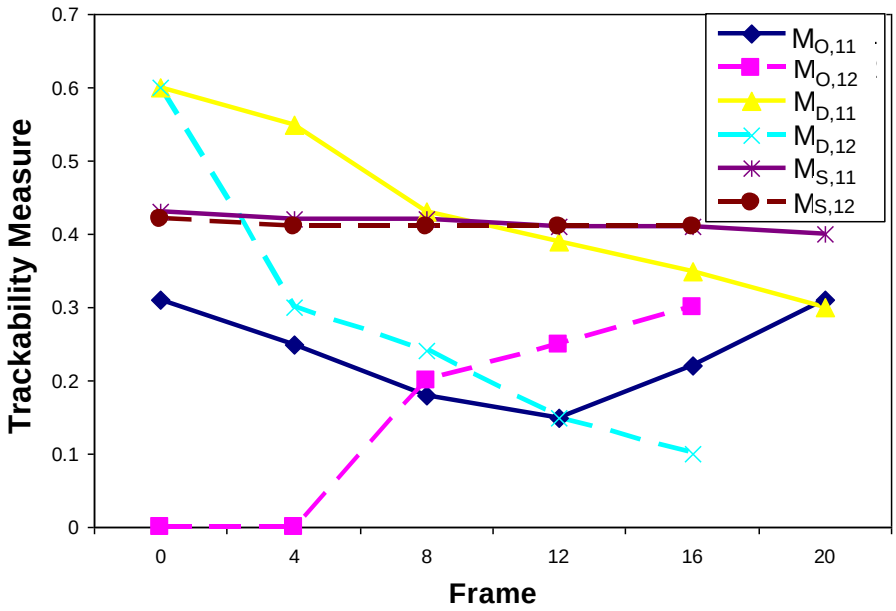


Figure 9.26 Illustration of continuous trackability measures, $M_{S,ij}$, $M_{D,ij}$, and $M_{O,ij}$, of objects 1 from frame f_n to f_{n+20} in real-time tracking sequence 1.

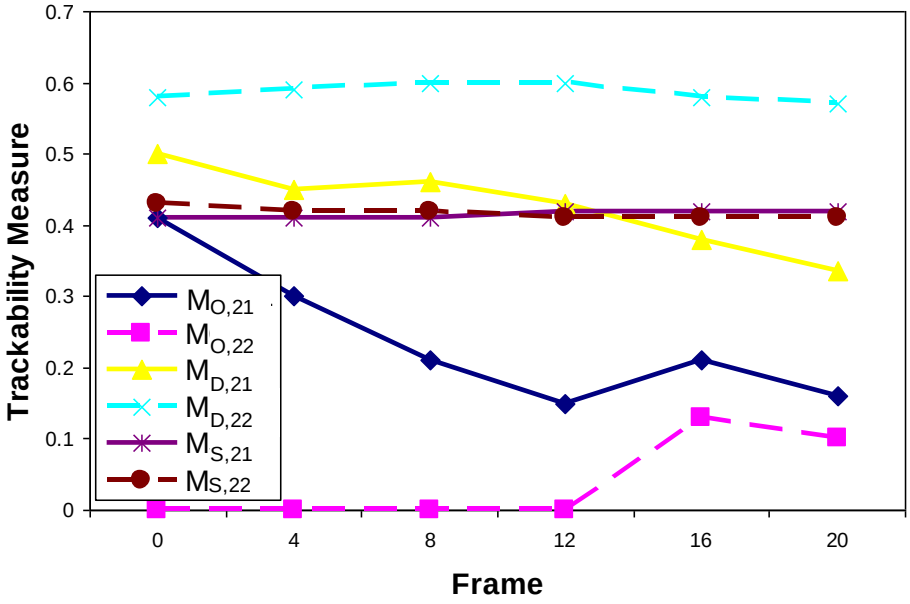


Figure 9.27 Illustration of continuous trackability measures, $M_{S,ij}$, $M_{D,ij}$, and $M_{O,ij}$, of objects 2 from frame f_n to f_{n+20} in real-time tracking sequence 1.

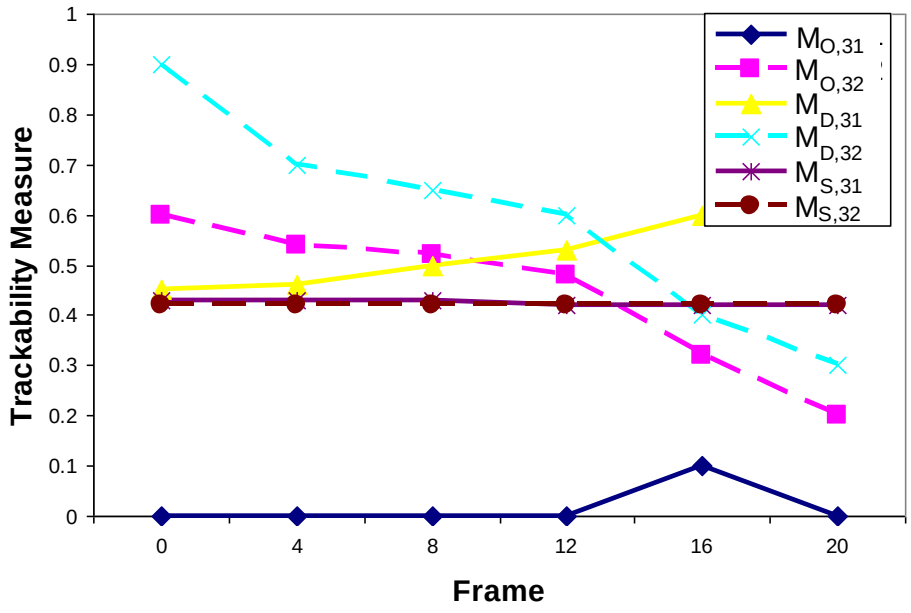


Figure 9.28 Illustration of continuous trackability measures, $M_{S,ij}$, $M_{D,ij}$, and $M_{O,ij}$, of objects 3 from frame f_n to f_{n+20} in real-time tracking sequence 1.

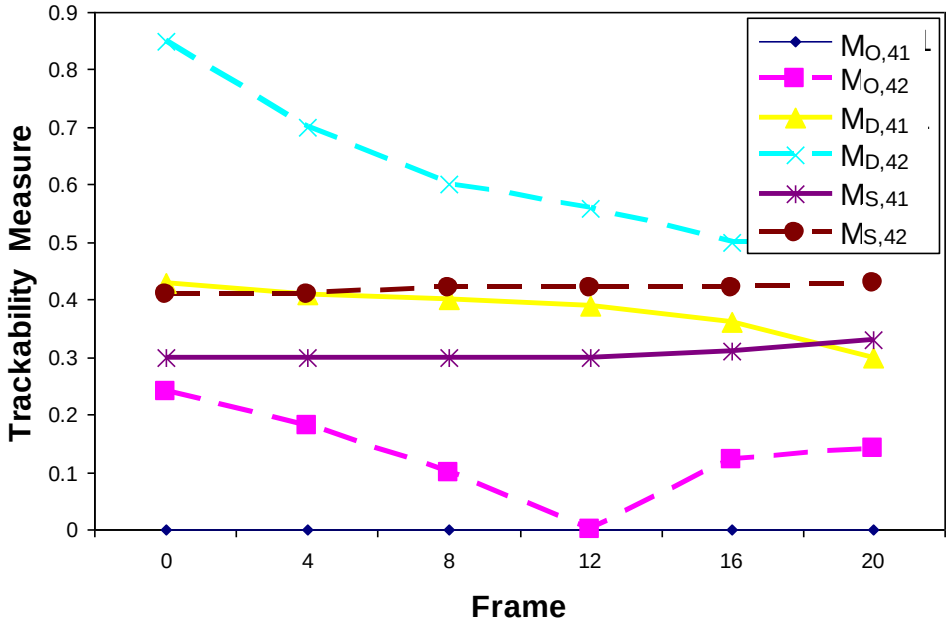


Figure 9.29 Illustration of continuous trackability measures, $M_{S,ij}$, $M_{D,ij}$, and $M_{O,ij}$, of objects 4 from frame f_n to f_{n+20} in real-time tracking sequence 1.

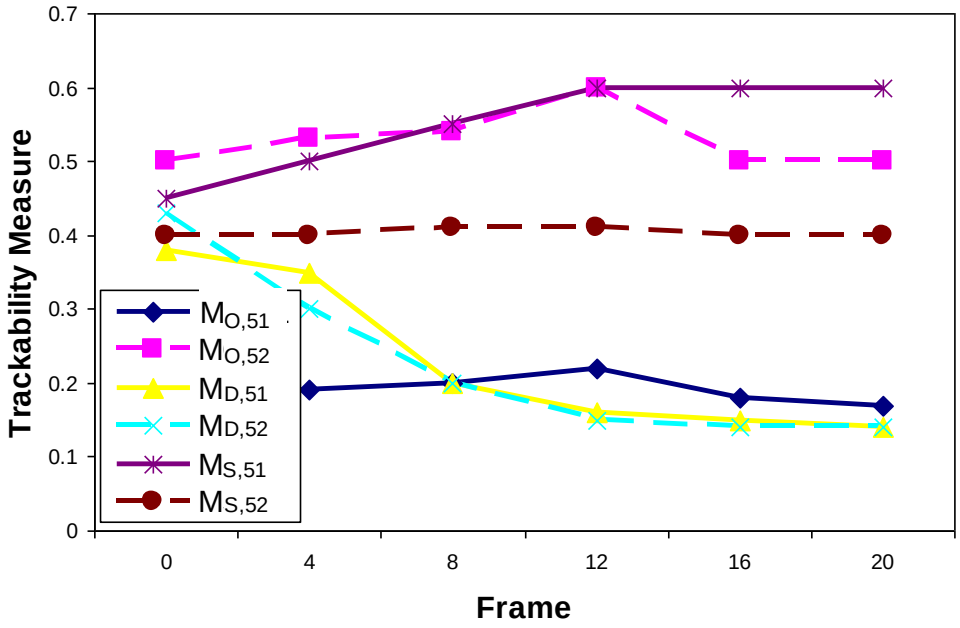


Figure 9.30 Illustration of continuous trackability measures, $M_{S,ij}$, $M_{D,ij}$, and $M_{O,ij}$, of objects 5 from frame f_n to f_{n+20} in real-time tracking sequence 1.

object 3 in the image of camera 1 while object 1 is blocked by object 2 in the image of camera 2. Both objects can be observed without occlusion in the other camera. Thus, objects 4 and 1 are transferred to camera 2 and 1, respectively. Object 5 in the camera 1 is close to object 3 and 4. Its $M_{O,51}$ is 0.18, less than T_O = 0.2. A handoff request is, therefore, triggered for object 5. On the other hand, camera 1 sends out a handoff request to its adjacent camera 2 and receives a positive response. As a result, object 5 in camera 1 will be transferred to camera 2, as marked by a yellow rectangle. Similarly, in frame f_{n+15}, object 5 in camera 2 is close to the edge of the camera's FOV, where its $M_{D,52}$ is 0.15 and less than T_D = 0.2. It requires camera handoff. On the other hand, camera 2 sends out the handoff request to its adjacent camera 1 and the request is granted, which is marked with a yellow rectangle in the camera 1. In general, we can see trackability measure gives a quantified metric to direct the camera handoff successfully and before the tracked object is occluded or falls out of FOV of currently observing camera.

9.5.2 Experiments on adaptive resource management

In order to illustrate the importance of our proposed adaptive resource management in camera handoff, Figure 9.31 illustrates sampled frames at f_n and f_{n+15} from real-time tracking sequence 2 with three static perspective cameras. The cameras' positions are specified in Fig. 7 as camera 3, 4 and 5. To illustrate the effectiveness of adaptive resource management, we focus on object 1. In frame f_n, even though camera 5 can see object 1, it does not track the object. This is because camera 4 tracks object 1 first and does not send out handoff request to adjacent cameras. In f

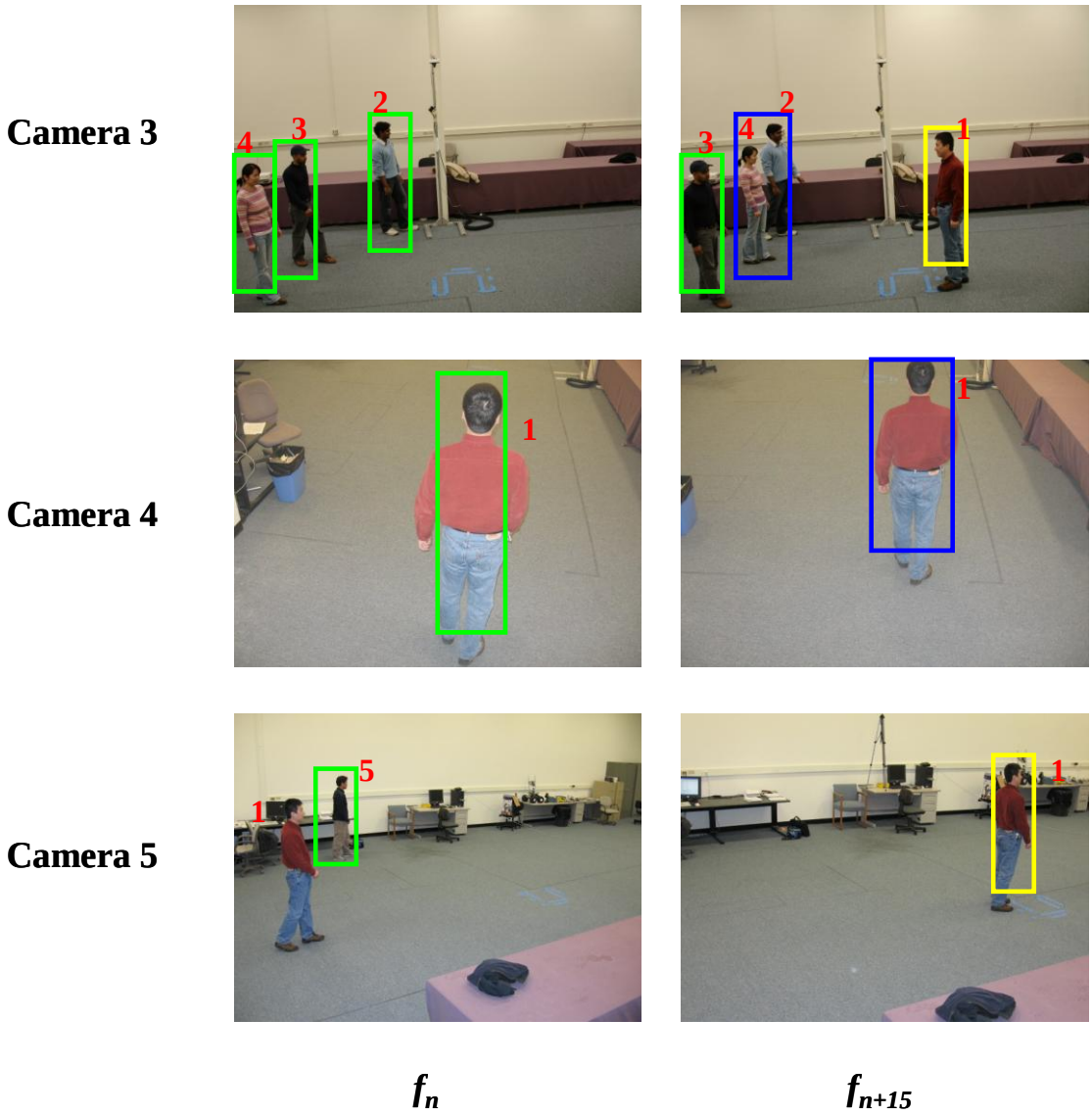


Figure 9.31 Illustration of the effectiveness of our proposed adaptive resource management in the camera handoff procedure at sampled frames f_n and f_{n+15} in real-time tracking sequence 2.

frame f_{n+15}, object 1 is moving out of FOV of camera 4 and camera 4 had send out handoff request to adjacent camera 3 and 5 before frame f_{n+15}. Since camera 3 has reached its maximum system load ($P_{O,13'}$ = 0.9 and $P_{O,15'}$ =0.1) and M_S, M_D, and M_O are not dominant factors in the camera selection process, camera 5 is the next best camera to track object 1. In general, our adaptive resource management is able to guide the camera handoff procedure to choose the least system load.

9.5.3 Experiments on overall performance

In order to examine the overall performance of our proposed camera handoff algorithm including trackability measure and adaptive resource management, the algorithm discussed in [Khan03] is implemented and serves as the comparison reference. The reference algorithm simply triggers a handoff request whenever the object of interest is close to the edge of camera's FOV without regarding the system's load, object priority, and the next best camera to track the object. Note that since there is no direct works corresponding to ours to the best of our knowledge, we choose Khan and Shah' work as a symbolic algorithm to demonstrate problems we face and then overcome in a real-life case. To accommodate Khan and Shah's work to our experiments, we make the following adjustments for their algorithms: (1) we trigger a handoff request when its distance to the edge of the camera's FOV (M_D) is smaller than the predefined threshold, T_D; (2) we choose the next best camera by merely the biggest M_D in adjacent cameras; (3) According to our experiment, average 10 frames is necessary for Khan and Shah's work to carry out a successful consistent labeling in a general situation. The failure of consistent labeling may occur when less than 10 frames are collected before the object is moving out of FOV of currently observing camera. One solution to reduce the possibility of failure of consistent labeling is to increase the overlapped views among adjacent cameras. This leads to the decreased overall coverage, thus, requiring more cameras to cover the area. This may not be practical in many cases. Thus, optimizing the tradeoff between coverage and overlapped views is used in this experiment. As a result, accumulating sufficient number of frames is necessary before objects fall out of FOV of currently observing camera to avoid the failure of handoff process.

In our experiment, we first illustrate how frame rates fluctuate when not considering adaptive resource management scheme in the tracking system. The overall tracking rate, the ratio between the time of objects being tracked by the system and the total time of objects staying in the FOV of the system, is used to describe the system's overall performance. To obtain a statistically valid estimation of the overall tracking rate, simulations are carried out to enable a large amount of tests under various conditions. Several points of interest are generated randomly to form a pedestrian trace. Overall tracking rate is obtained from simulation results of 300 randomly generated traces. In order to understand the behavior of our proposed camera handoff algorithm facing varying arrival rates of the objects with low and high priority, the ratio λ_L/λ_H is set to vary from 0.8 to 1.2. The expected probability of camera overload for objects with low and high priorities is $P_{th,L}$ = 0.2 and $P_{th,H}$ = 0.2. Note that once we lose the track of the object due to failure of camera handoff, we will not recover it until the object moves to another adjacent

camera.

Figure 9.32, 9.33, and 9.34 show performance comparisons of our adaptive resource management method and the reference algorithm [Khan03] with various λ_L/λ_H in term of the handoff success rate. The notation Adaptive-0.8 suggests a system using our proposed resource management method with λ_L/λ_H =0.8 and the notation KS-0.8 means the reference system with λ_L/λ_H =0.8. Figure 9.32 illustrates the system equipped with our adaptive resource management can keep a steady frame rate of 8fps while the frame rate of the system based on the reference algorithm varies between 3fps and 8fps. In addition, in Figure 9.33 and 9.34, regardless of λ_L/λ_H , the overall tracking rate of our adaptive approach is higher than that of the static approach. A considerable improvement in overall tracking rate by 20% is achieved in comparison with the Khan and Shah's work. The observed inferior overall tracking rate of the reference method results from its fluctuating frame rate. When the frame rate is low, less information is acquired for the execution of consistent labeling, hence deteriorating the accuracy of identity matching and then the overall tracking rate. In other word, the continuity of objects being tracked in the system is compromised.

Figure 9.35 illustrates sampled frames from f_n to f_{n+30} from real-time tracking sequence 3 with three static perspective cameras. In this sequence, since objects 1 and 2 are carrying valuable materials, reduced frame rates is not allowed for the sake of security. Thus, in this experiment, objects 1 and 2 represent the high priority rank. Object 3 represents the low priority rank. The cameras' positions are specified in Figure 9.21 as camera 1, 6, and 7. Table 9.5 lists $M_{S,ij}$, $M_{D,ij}$, $M_{O,ij}$, and $P_{O,ij'}$ for the i^{th} object observed by the j^{th} camera at frames from f_n to f_{n+30}, where i ranges from 1 to 3 and j is either 1, 6 or 7. In frame f_n, object 1 and 2 are tracked by camera 7. Object 3 is tracked by camera 6. In frame f_{n+10}, object 2 is occluded by object 1 in camera 7. However, our trackability measure has triggered the camera handoff procedure before the occlusion happens. Even though object 1 can be seen by camera 1 and 6 and represents similar $M_{S,ij}$, $M_{D,ij}$, and $M_{O,ij}$ in both cameras, camera 1 has the lowest computational load as compared with camera 6 ($P_{O,11'}$ = 0.1 and $P_{O,16'}$ =0.3). Thus, object 1 is transferred to camera 1. In frame f_{n+20}, object 2 is under camera handoff procedure since it is moving out of FOV of camera 1 ($M_{D,21}$=0.15). In frame f_{n+30}, object 2 had been successfully handed over to camera 7. In general, we can see that the newly defined trackability measure gives a quantified metric to direct the camera handoff successfully and smoothly before the tracked object is occluded or falls out of FOV of currently observing camera. Also, our adaptive resource management is able to effectively guide camera handoff to choose the camera with the least system load. This can reduce the probability of missing critical events and improve the system's level of threat awareness. The maintained frame rate also stabilizes the performance of consistent labeling and leads to an improved handoff success rate.

9.5.4 Experiments on PETS's video sequence

Figure 9.36 illustrates sampled frames at f_{1147}, f_{1225}, f_{1292}, f_{1348}, and f_{1414} from PETS' 2006 dataset S7 where it contains a single person with a suitcase who loiters before leaving the item of

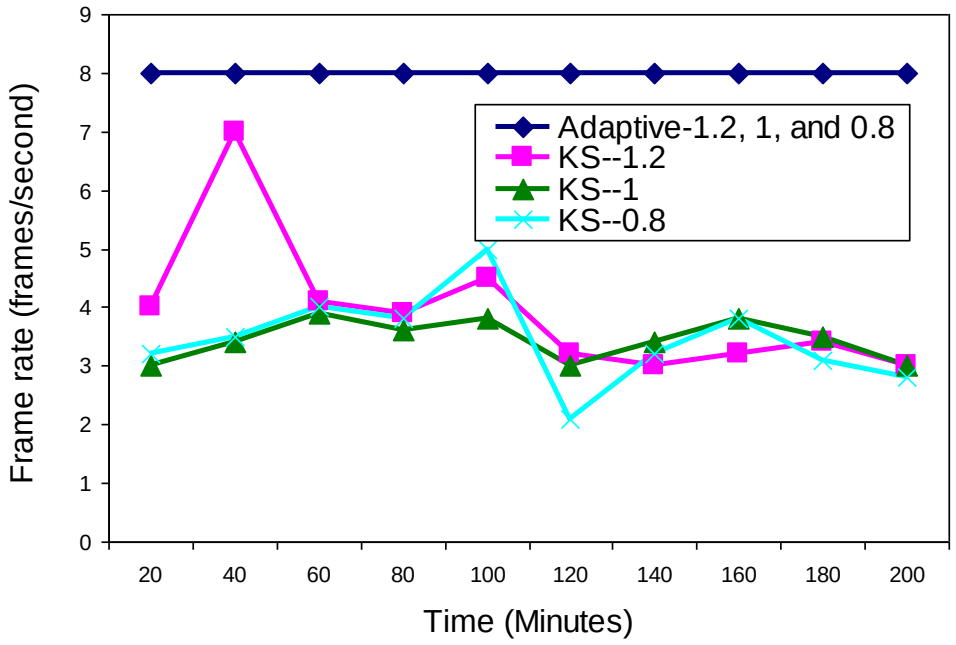


Figure 9.32 Comparisons of camera handoff approaches with our proposed adaptive and Khan & Shah' static resource management methods with various $\frac{\lambda_L}{\lambda_H}$. This illustrates how frame rates fluctuate when not considering the adaptive resource management scheme in the system. Adaptive and KS denote our proposed adaptive and Khan & Shah' static resource management methods respectively.

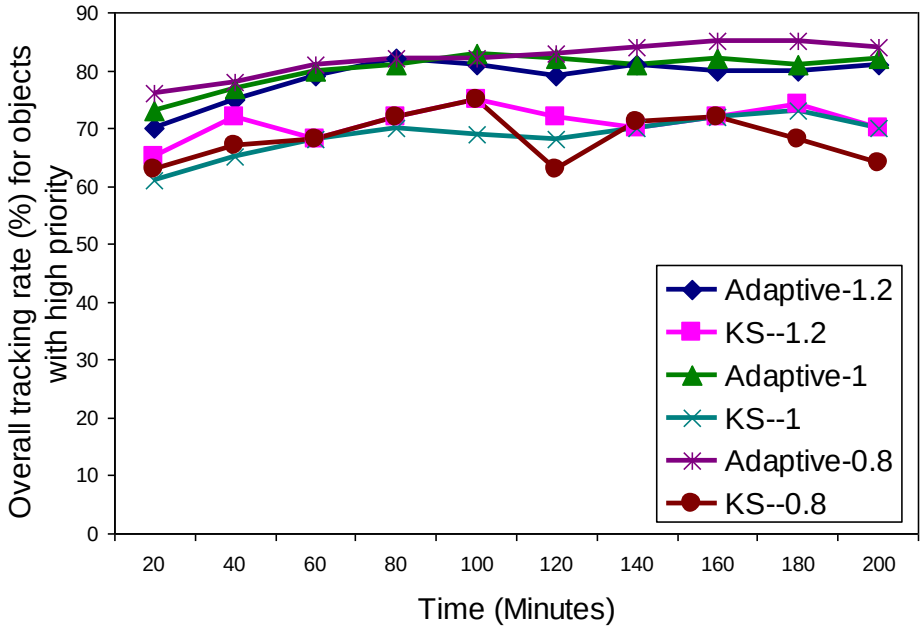


Figure 9.33 Comparisons of camera handoff approaches with our proposed adaptive and Khan & Shah' static resource management methods with various $\frac{\lambda_L}{\lambda_H}$. This illustrates handoff success rate for objects with high priority.

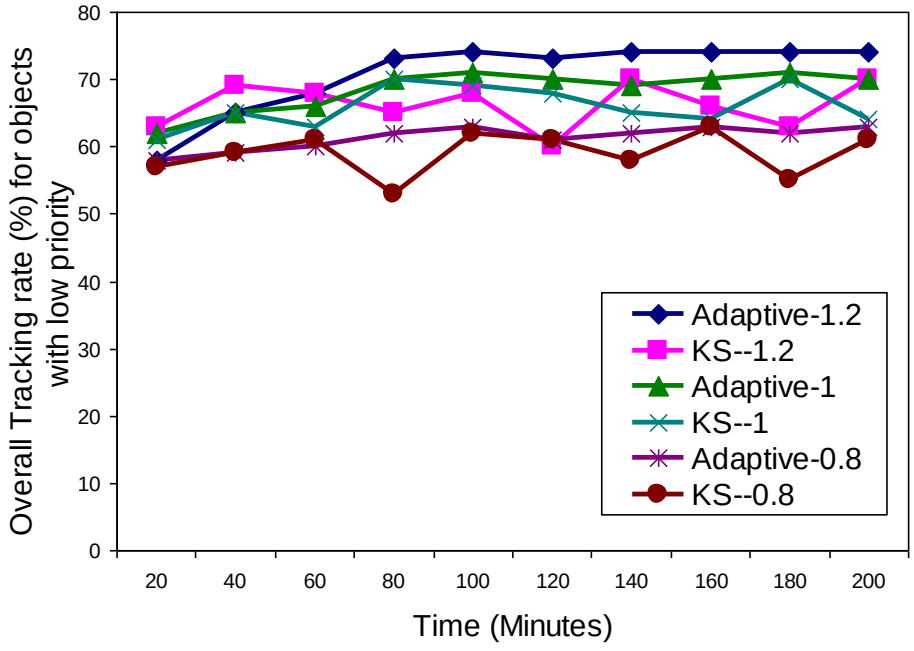


Figure 9.34 Comparisons of camera handoff approaches with our proposed adaptive and Khan & Shah' static resource management methods with various $\frac{\lambda_L}{\lambda_H}$. This illustrates handoff success rate for objects with low priority.

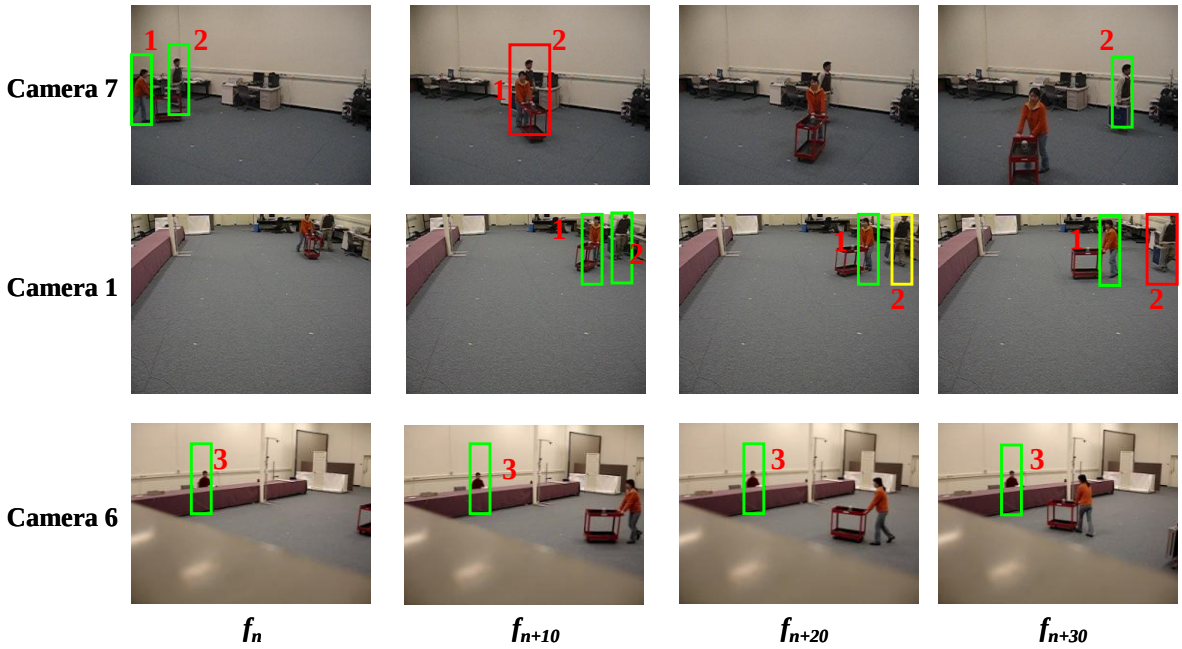


Figure 9.35 Illustration of the effectiveness of our proposed camera handoff procedure including trackability measure and adaptive resource management at sampled frames f_n and f_{n+30} in real-time tracking sequence 3.

Table 9.5 The illustration of $M_{O,ij}$, $M_{D,ij}$,, $M_{S,ij}$, and $P_{O,ij'}$ shown in Figure 9.35

		Object 1 (*i=1*)				Object 2 (*i=2*)				Object 3 (*i=3*)			
		f_n	f_{n+10}	f_{n+20}	f_{n+30}	f_n	f_{n+10}	f_{n+20}	f_{n+30}	f_n	f_{n+10}	f_{n+20}	f_{n+30}
Camera 7 (*j=7*)	$M_{O,ij}$	0.31	0	×	×	0.3 1	0	×	0.7	×	×	×	×
	$M_{D,ij}$	0.16	0.85	×	×	0.35	0.84	×	0.39	×	×	×	×
	$M_{S\,ij}$	0.5	0.49	×	×	0.48	0.48	×	0.8	×	×	×	×
	$P_{O,ij'}$	0.6	0.6	0.1	0.3	0.6	0.6	0.1	0.3	0.6	0.6	0.1	0.3
Camera 1 (*j=1*)	$M_{O,ij}$	×	0.23	0.25	0.3	×	0.23	0.28	0.29	×	×	×	×
	$M_{D,ij}$	×	0.3	0.32	0.34	×	0.2	0.15	0.1	×	×	×	×
	$M_{S\,ij}$	×	0.7	0.7	0.71	×	0.7	0.65	0.6	×	×	×	×
	$P_{O,ij'}$	0.1	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6	0.6
Camera 6 (*j=6*)	$M_{O,ij}$	×	×	×	×	×	×	×	×	0.99	0.89	0.69	0.59
	$M_{D,ij}$	×	×	×	×	×	×	×	×	0.8	0.8	0.8	0.8
	$M_{S\,ij}$	×	×	×	×	×	×	×	×	0.75	0.75	0.75	0.75
	$P_{O,ij'}$	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3

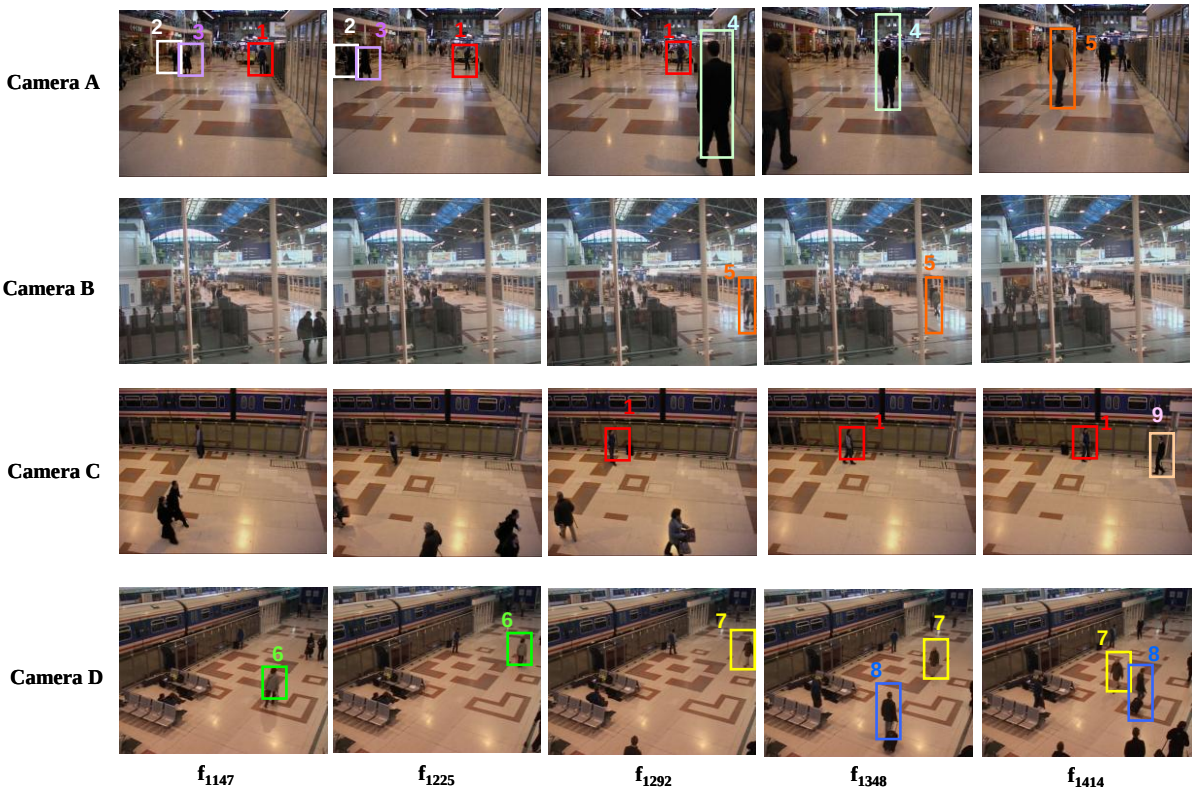


Figure 9.36 Illustration of the effectiveness of our proposed camera handoff procedure including trackability measure and adaptive resource management at sampled frames f_{1147}, f_{1225}, f_{1292}, f_{1348}, and f_{1414} in PETS' 2006 dataset S7.

luggage unattended and four cameras are monitoring the scene. During this event other people move in close proximity to the item of luggage. Two priority levels are assigned to the objects, N_{pr}=2. The maximum number of objects that can be tracked simultaneously is also three for all cameras, N_{max} = 3. The thresholds T_O, T_D, and T_S are 0.2 to comply with the time needed for executing camera handoff (5 seconds average) and the maximal moving speed of the objects (0.6 meters per second). In this sequence, since object 1 is leaving his luggage unattended in the scene, which may post a threat to the area, reduced frame rates are not allowed. To illustrate the effectiveness of our proposed handoff algorithm, we focus on object 1. In the beginning, object 1 is tracked by camera first. In frame f_{1292}, because object 4 is going to occlude object 1 ($M_{o,1A}$=0.18), handoff request from camera A is sent out to adjacent cameras B, C, and D. Since camera C has the lowest system load ($P_{O,1B'}$ = 0.4, $P_{O,1C'}$ = 0.1 and $P_{O,1D'}$ = 0.3), the resolution of object 1 in camera B is too low ($M_{S,1B}$=0.13), and object 1 has similar $M_{S,ij}$, $M_{D,ij}$, and $M_{O,ij}$ in both cameras C and D, object 1 is transferred to camera C. In general, we can see that our defined trackability measure gives a quantified metric to direct the camera handoff successfully and smoothly before the tracked object is occluded by other objects. Also, our adaptive resource management is able to effectively guide camera handoff to choose the camera with the least system load. This can reduce the probability of missing critical events and improve the system's level of threat awareness.

9.6 Mobile tracking platform

Figure 9.37 shows the entire system, including the web camera, range sensor, and robotic platform. Robotic mobility is accomplished through two independent tracks. These two tracks are modularly interchangeable, and each is capable of moving the robot by itself. Motion can be controlled directly by a computer sending motion commands into the track motors via a RS232 signal. Figure 9.38 shows the experimental results.

In Figure 9.38, the robot initially moves backward to avoid the first obstacle which is lying very close to the beginning position of the robot. Without this backward movement, the robot could strike the obstacle (while turning for following the object), potentially losing sight of the tracked object. In positions (1), (3), and (5), since there is no obstacle sensed in the laser scan range, the system does not trigger the obstacle avoidance phase. Experiment results show that the robot can continuously tract the moving object and the dynamic goal potential fields method can guide the robot to move to a new position without colliding with any obstacle while the tracked object is moving with a low speed.

In Figure 9.39, the experimental results represent the system performance with different robot step sizes and object moving speeds. Figure 9.39(a) represents that we set that the object moving speed equals to 2m/s and robot step size is about 5inches. Our experiment result shows that this algorithm could find a very good adjusted path to move towards new object position. Figure 9.39(b) represents we set the object moving speed equals to 2m/s and robot step size is about 20 inches. Our experiment result shows that this algorithm could not adjust its path very well to move towards new object position. This experiment shows that big step size has trouble in dealing with high-speed object. Figure 9.39(c) and (d) represent that we use the dynamic step

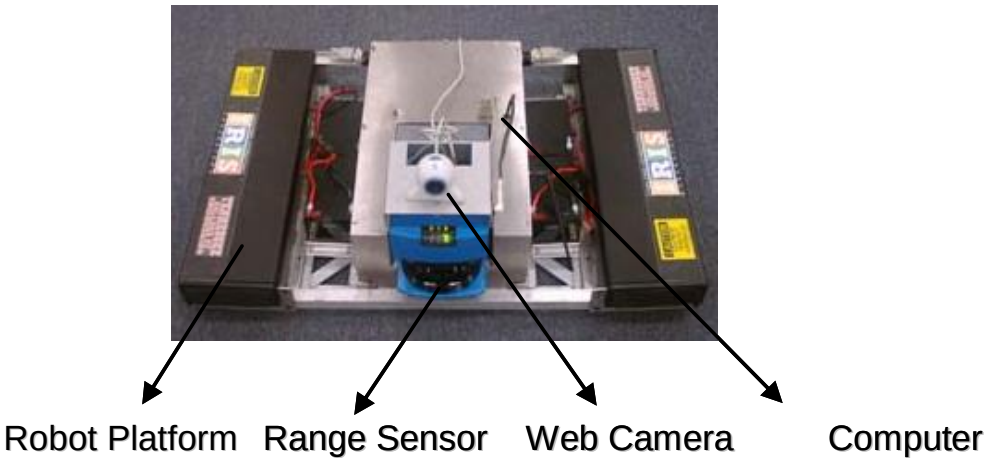
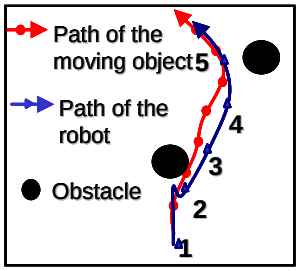
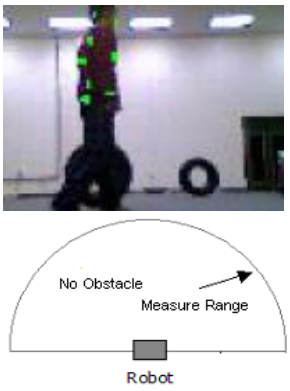


Figure 9.37 This system picture shows the platform components.

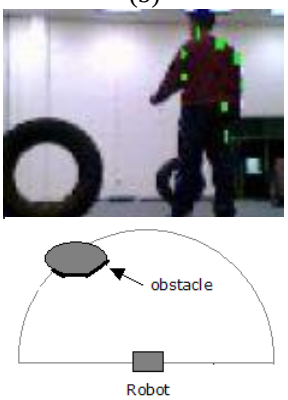
(a)



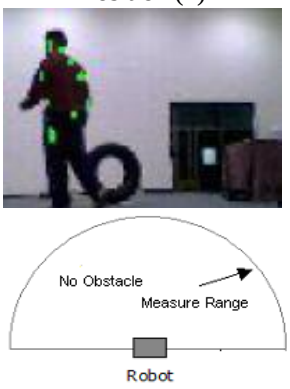
(b)



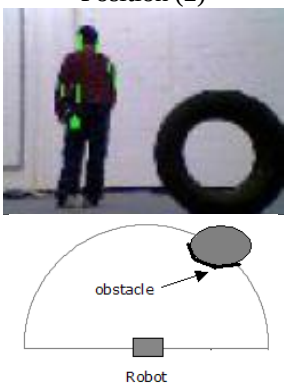
Position (1)



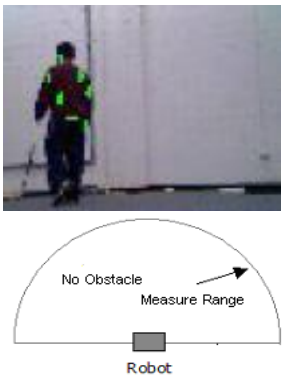
Position (2)



Position (3)



Position (4)



Position (5)

Figure 9.38 Experimental result of the mobile tracking platform. (a) represents the indoor experimental environment. (b) represents the relative path of the moving object and robot.

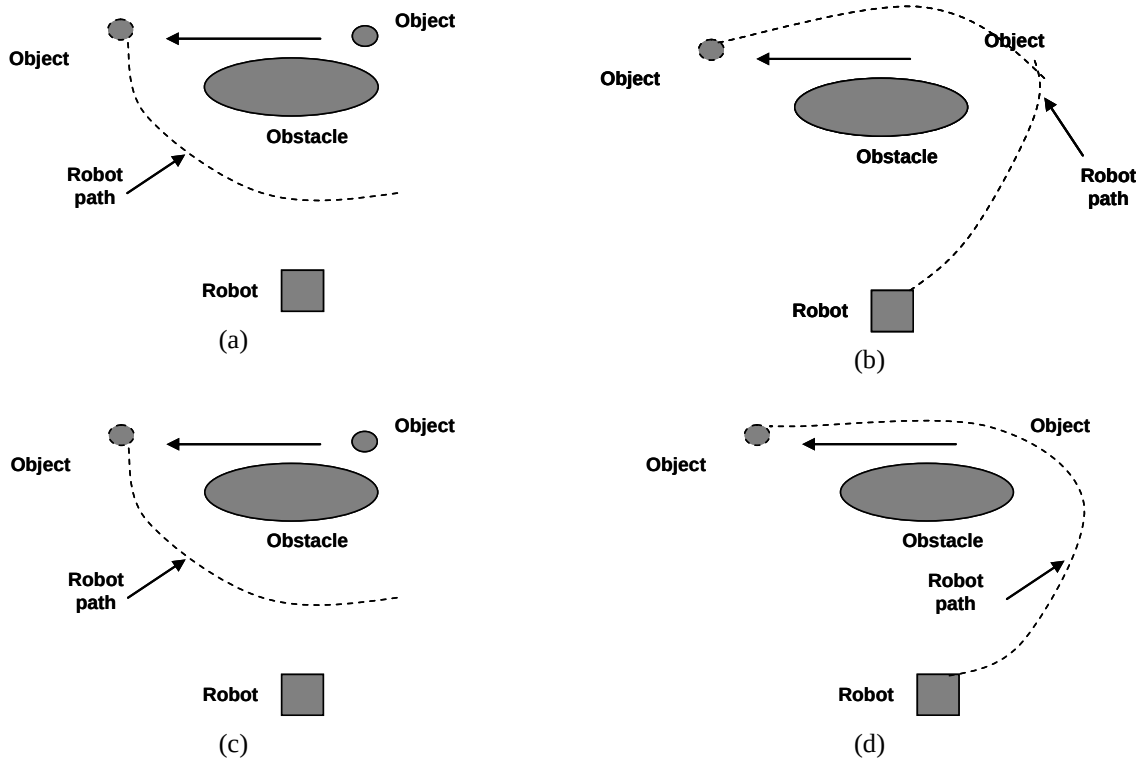


Figure 9.39 Illustration of the system performance with different robot step sizes and object moving speeds. (a) moving speed = 2m/s and robot step size = 5 inches. (b) Object moving speed = 2m/s and robot step size = 20 inches. (c) Object moving speed = 2m/s and dynamic robot step size. (d) moving speed = 0.5m/s and dynamic robot step size.

size strategy, which means that the robot can adjust its step size according to the object's speed. These experiments show that dynamic step size can always find better adjusted-path when the object is moving in relatively slow speed.

10 Conclusion

The overall goal of this dissertation work is to investigate the automated surveillance system with multi-camera and robotic platform. Dual camera systems have been widely used in surveillance because of the ability to explore the wide field of view (FOV) of the omnidirectional camera and the wide zoom range of the PTZ camera. Most existing algorithms require a priori knowledge of the omnidirectional camera's projection model to solve the non-linear spatial correspondences between the two cameras. To overcome this limitation, two methods are proposed: (1) geometry and (2) homography calibration, where polynomials with automated model selection are used to approximate the camera's projection model and spatial mapping, respectively. Our proposed methods shown in Section 3 not only improve the mapping accuracy by reducing the dependence on the knowledge of the projection model but also feature reduced computations and improved flexibility in adjusting to varying system configurations. Nevertheless, a surveillance system with multiple PTZ cameras is more and more popular. In this muli-PTZ camera system, we need to change pan and tilt angles, and zoom degree from time to time to have a better monitoring results. Most existing algorithms [Basu97, Chen07] require fully calibrated PTZ cameras to infer the relative positioning and orientation between two PTZ cameras, which is not only a time-consuming procedure, but lack the flexibility in adjusting to varying system configurations. In addition, how to effectively coordinate those PTZ cameras to obtain the best monitoring result for each object of interest is still in question.

Camera handoff is a crucial step to obtain a continuously tracked and consistently labeled trajectory of the object of interest in automated multi-camera surveillance. Camera handoff should comprise three fundamental components, time to trigger handoff process, the execution of consistent labeling, and the selection of the next optimal camera. In Section 4, we used an observation measure to quantitatively formulate the effectiveness of object tracking so that we can trigger camera handoff timely and select the next camera appropriately before the tracked object falls out of the field of view (FOV) of the currently observing camera. In the meantime, we presented a novel solution to the consistent labeling problem in omnidirectional cameras. Our proposed consistent labeling approach can perform as accurately as the geometry-based approach without tedious calibration processes and outperform Calderara's homography-based approach.

Since most existing camera handoff algorithms including ours need a certain amount of time to successfully carry out the camera handoff procedure, especially for the execution of consistent labeling, we introduced an additional constraint to optimally reserve sufficient cameras' overlapped FOVs for the camera placement. Our proposed camera placement method exhibits a significant increase in the camera handoff success rate at the cost of slightly decreased coverage, as compared to Erdem and Sclaroff's method without considering the necessary overlapped

FOVs. Nevertheless, although most multiple objects tracking algorithms are proven efficient, we find it difficult to maintain a constant frame rate given limited resources. Herewith, resources include (1) CPU capacity for executing object tracking, crowd segmentation, and behavior understanding in a completely automated manner, and (2) network bandwidth for exchanging camera handoff information. The computational complexity of most existing tracking systems is from the running times of order, $O(N_p n)$ to $O(N_p n^3)$, where N_p is the number of tracked objects. There inherently exists an upper bound on the number of objects that can be tracked simultaneously without deteriorating the system's frame rate. On the other hand, the study of camera overload is another important criterion to be incorporated into camera handoff to maintain a required minimum frame rate, which is ignored by most existing handoff algorithms.

However, in the majority of surveillance systems, their cameras are stationary. These stationary systems often require the desired object stay within the surveillance range of the system. Thus, the robotic platform we propose uses a visual camera to sense the movement of the desired object and a range sensor to help the robot detect and then avoid obstacles in real time while continuing to track and follow the desired object. Experiment shows this robotic and intelligent system can fulfill the requirements of tracking an object and avoiding obstacles simultaneously when the object moves in speed of 4 km/hr.

In previous chapters, we have presented a survey of multi-camera surveillance systems, derived our theoretical framework, and demonstrated the effectiveness of the proposed methods via extensive experiments and comparisons with existing leading algorithms. We conclude this dissertation with a brief summary of the contributions and a short discussion of the directions for future research.

10.1 Summary of contributions

- **Heterogeneous mapping of omnidirectional and PTZ cameras**: Two spatial mapping methods, geometry and homography calibration, are proposed. The geometry calibration method can approximate various projection models, features automatic model selection and straightforward implementation for off-the-shelf cameras, and eliminates the requirement of a known projection model. The homography calibration method directly derives a unified polynomial model between the pan, tilt, zoom values of the PTZ camera and the projected point on the image plane of the omnidirectional camera. In comparison with the reference algorithms [Cui98, Scotti05] that require the knowledge of the camera's projection model, our methods select the optimal model according to a statistical metric or test considering both uncertainty in estimation and modeling. Therefore, the proposed methods feature improved mapping accuracy, reduced computational complexity, and ability to accommodate varying camera configurations.

- **Homogenous mapping of PTZ cameras:** One spatial mapping method is proposed to derive the relation of pan, tilt, and zoom values between any pair of PTZ cameras without prior knowledge of their intrinsic parameters and relative positions. In comparison with the reference algorithm [Chen07A], our proposed approach not only reduces the dependence on the knowledge of intrinsic parameters of PTZ camera, but improves the

degree of autonomy and reduces the system's computational complexity at the cost of slightly decreased pixel accuracy. In general, this slightly decreased pixel accuracy does not affect the overall performance for the application of automated surveillance systems, as long as the desired object can be seen within the field of view and can be compensated by consistent labeling approaches [Lowe04] without added cost.

- **Target hopping for the dual camera system:** The next best target (NBT) problem is addressed, which exemplifies a typical problem in multiple object tracking using cameras with different FOVs and resolutions. An adaptive algorithm is designed for a minimized computational and traveling time. The proposed algorithm studies the targets' dynamic distribution in the environment and generates the optimal visiting sequence for the PTZ camera. In comparison with the sequential visiting and nearest neighbor methods, the proposed adaptive algorithm requires less computational and visiting time, which is critical to real-time applications.

- **Camera handoff and determination of size of overlapped view for multiple omnidirectional cameras:** We present a novel solution to the consistent labeling problem in omnidirectional cameras. An automatic spatial mapping procedure considering both the noise inherent to the tracking algorithms used by the system and the lens distortion introduced by omnidirectional cameras is proposed to obtain the correspondences between the trajectories of the same object seen in different omnidirectional cameras without human interventions. For the purpose of automated and persistent object tracking, typical of most surveillance requirements, we propose to use the Wilcoxon Signed-Rank Test for the trajectory association. Our proposed consistent labeling algorithm can perform as accurately as the geometry-based approach without tedious calibration processes and outperform Calderara's homography-based approach [Calderara05]. In the meantime, our proposed camera placement approach that optimally reserves sufficient cameras' overlapped FOVs accomplishes the task of automated and persistent object tracking. As a result, it features a significant increase in the handoff success rate at the cost of slightly decreased coverage as compared to Erdem and Sclaroff's method [Erdem06] without considering the necessary overlapped FOVs.

- **Resource management mechanism:** our handoff algorithm employs an adaptive resource management mechanism to dynamically allocate cameras' resources to multiple objects with different privileges so that the required minimum frame rate is maintained. In other words, the overload probability is one important criterion to evaluate the performance of a multi-camera system fulfilling multiple object tracking. It determines the number of objects that may be dropped due to limited resources. Therefore, in practice, it is desirable to distribute the resources dynamically according to the system's current load and the object's priority rank. Experimental results illustrated that our handoff algorithm outperforms Khan and Shah's method [Khan03] by keeping a higher overall tracking rate and a more stable frame rate. This improves the reliability of the tracking system for continuously tracking multiple objects across multiple cameras

- **Mobile tracking platform:** We describe a robotic application that tracks a moving object by utilizing a mobile robot with multiple sensors. The robotic platform uses a visual camera to sense the movement of the desired object and a range sensor to help the robot detect and then avoid obstacles in real time while continuing to track and follow the desired object. In terms of real-time obstacle avoidance capacity, we also presents a modified potential field algorithm called Dynamic Goal Potential Field algorithm (DGPF) for this robotic application specifically. Experiment shows this robotic and intelligent system can fulfill the requirements of tracking an object and avoiding obstacles simultaneously when the object is moving at about 4 km/hr

10.2 Directions for future work

Automated video surveillance systems have been widely studied in the past decade. Because of its promise that it can monitor an area without human intervention all times, its applications can cover crime prevention [Lei06], pre-emptive interest protection [Collins00], national security [Yilmaz06], etc. However, even though results most of exiting automated video surveillance systems present are efficient in some cases, the evaluation of the robustness of an automated video surveillance system remains difficult given the requirement that a system should operate in all times and under varying conditions such as weathers, number of objects, illuminations, etc. Thus, the issue of evaluating the performance of automated video surveillance systems has become increasingly important.

The conventional approach [Jaynes02, Doermann00] for performance evaluation is to generate ground truth from pre-recorded video sequences. List et al. [List05] pointed out that the manual generation of ground truth is a time-consuming task and, thus, inevitably error prone. Performance evaluation algorithms based on comparison with ground truth can be further classified according to the type of metrics employed [Schlogl04, Erdem04]. In some cases, these metrics are useful to assess the overall segmentation quality on frame-to-frame basis but fail to provide an evaluation of individual object segmentation, because they are restricted to pixel-level discrepancy between the detected foreground and the ground truth [Lazarevic-McManus06]. In addition, the aforementioned methods, deterministic approaches, usually reserve a robustness margin or factor, such as two or three times the expected number of objects or the strength of illumination, in the system design. This often results in overdesign, thus increasing costs, or underdesign, causing frequent system failures from unexpected disturbances.

Reliability is defined as the probability that a component or system can perform a required function for a given period of time, *t*, when used under stated operating conditions [Ebeling97]. In other words, it is the probability of a non-failure over time. In terms of the interpretation of quality, reliability is concerned with how long the system continues to function once it becomes operational. A poor-quality system will likely have poor reliability, and a high-quality system will have a high reliability. To define the reliability in a system, three definitions must be made specific: (1) failures should be defined relative to the function being performed by the system; (2) the unit of time must be identified; (3) the system should be observed under normal performance such as environment, design loads, and operating conditions.

The reliability can be expressed as

$$R(t) = Pr\{T \geq t\}, \tag{10.1}$$

where T represents the time to failure of the system and $T \geq 0$. $R(t) \geq 0$, $R(0)=1$, and $\lim_{t\to\infty} R(t) = 0$. If we define

$$F(t) = 1 - R(t) = Pr\{T < t\}, \tag{10.2}$$

where $F(0)=0$ and $\lim_{t\to\infty} F(t) = 1$, then $F(t)$ is the probability that a failure occurs before time t. We will refer to $R(t)$ as the reliability function and $F(t)$ as the cumulative distribution function (CDF) of failure distribution, and

$$f(t) = \frac{dF(t)}{dt} = -\frac{dR(t)}{dt}, \tag{10.3}$$

as the probability density function (PDF). Thus, the failure rate, $\lambda(t)$, provides an instantaneous rate of failure at time t, which is defined as

$$\lambda(t) = -\frac{dR(t)}{dt}.\frac{1}{R(t)}, \tag{10.4}$$

Based on Equation (10.4), we can derive

$$R(t) = \exp\left[-\int_0^t \lambda(t')dt'\right], \tag{10.5}$$

In practices, most of governmental and private contract specifications require each function being performed by the system must have a 90 percent or better reliability over a designed time. Thus, the objective of our proposed framework is to derive the reliability function, $R(t)$, of the automated video surveillance system based on censored data and defined metrics. On the other hand, given a required environment we can give you a unified probability describing the chance of the system running functionally over a designed time.

In doing so, we perform a statistical test in order to accept or reject the hypothesis that the observed failure times come from a specified distribution. In general, fitting a theoretical distribution is preferred over empirically developing a model, namely nonparametric model, because empirical models do not provide sufficient information beyond the range of the sample data [Ebeling97]. In reliability engineering the tails of the distribution are of most interest. Moreover, often the failure process is a result of some physical phenomena that can be associated with a particular distribution. If the sample is consistent with a theoretical distribution, then much stronger results based on the properties of the theoretical distribution are possible. Figure 10.1 illustrates our proposed reliability assessment for automated video surveillance systems.

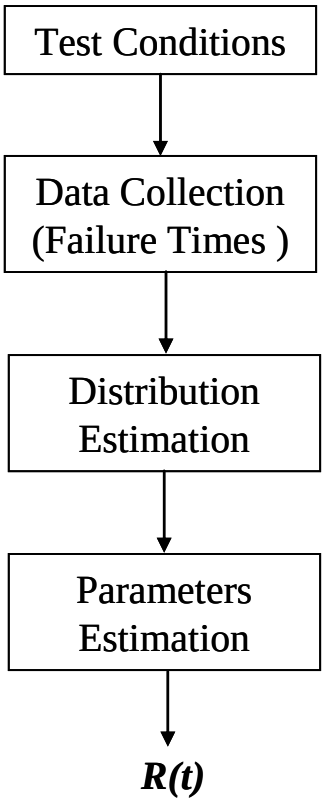


Figure 10.1 Illustration of our proposed framework for assessing the reliability of automated video surveillance systems.

In Figure 10.1, test conditions represent an environment where the system may be designed to operate. This condition can be set to generic terms such as an indoor illumination, randomly or regularly moving multi-object, etc. Once test conditions are established, we can perform iterative experiments to sample failure times, which of each failure time indicate how long the system can function in a predefined measurement. For example, the measurement is set to two randomly walking objects should be tracked simultaneously by the automated video surveillance. If not, record the time period from the beginning of this experiment to the failure.

Afterwards, we use Chi-Square goodness of fit test to determine which theoretical distribution can most appropriately represent the collected failure times because it does not have the restriction that location, scale, and shape parameters cannot be estimated from the data, as compared with Kolmogorov-Smirnov goodness of fit test. Such a test compares a null hypothesis (H_0) with an alternative hypothesis (H_1) having the following form:

H_0: the failure times came from the specified distribution.

H_1: the failure times did not come from the specified distribution.

In this paper, four theoretical distributions, exponential, weibull, normal, and lognormal are performed. In general, if the value of its test statistic is smaller than its critical value, H_0 is accepted. In the meantime, parameters of each distribution are estimated from maximum

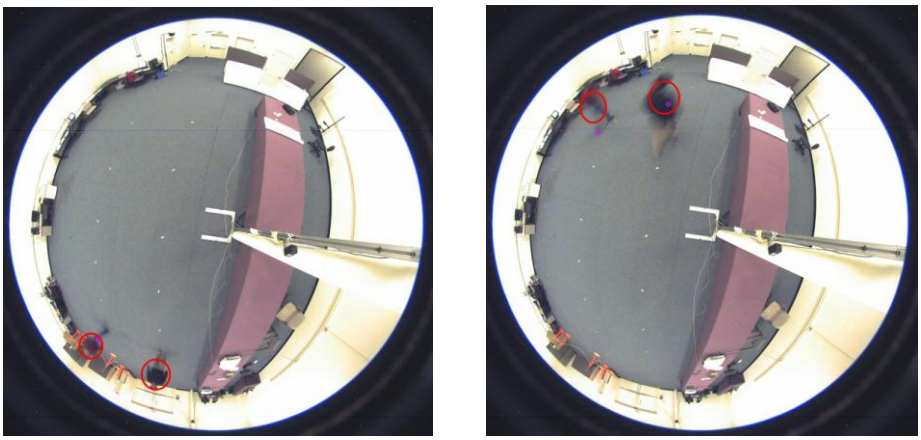
likelihood estimators (MLEs). In particular, since Chi-Square goodness of fit test can only present if this data come from the specified distribution, we have the likeliness that multiple theoretical distributions are accepted [Wackerly02]. Thus, the least-square curve fitting is used to determine which distribution is the best fit in the pool of accepted theoretical distributions. In general, the bigger the value of the least-square curve fitting (index of fit), the best fit it is [Wackerly02]. Finally, the reliability model of an automated video surveillance system under tested conditions is derived, which is based on Equation (10.2).

In our indoor real-time surveillance environment with dimensions of 30m×15m×3m, one omndirectional cameras (IQeve3) is placed in the middle of the environment at a height of 3m and used to fulfill multi-object tracking. Cui's Background differencing and radial profile analysis [Cui98] is used for multi-object tracking. Omnidirectional images, with a resolution of 320×320, are obtained via an intranet connection with maximal 10 frames per second. The strength of illumination is in between 400lux and 650lux in our indoor environment.

Two objects are randomly walking in the environment. Whenever two objects are not being simultaneously tracked by the systems, we stop the experiment and record the failure time, and repeatedly carry out the same experiment. Figure 10.2 illustrates one experimental video sequence. In Figure 10.2, two tracked objects, which are marked with two red circles, are walking back and forth in the monitored environment. At time 115 seconds, the system can only track one person, which only one red circle appears. Thus, the failure time of this experiment is 115 seconds. We repeat it again and collect 40 failure times. Table 10.1 summarizes Chi-Square goodness of fit and the least-square curve fitting for four theoretical distributions when the level of significance is set to 0.05. In Table 10.1, only Weibull distribution can represent the collected data. Since the estimated parameters for Weibull distribution are $\hat{\beta} = \mathbf{1.77}$ and $\hat{\theta} = \mathbf{122}$ in this case, the estimated reliability model [Ebeling97] of Cui's object tracking algorithm under our test environment with two randomly walking objects is

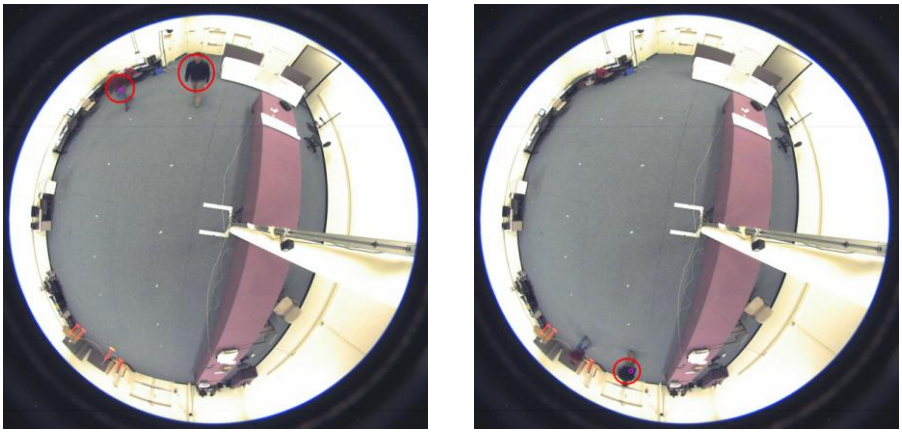
$$R(t) = e^{-(\frac{t}{122})^{1.77}}, \tag{10.6}$$

Based on Equation (10.6), for a desired 0.9 reliability in our test environment with two randomly walking object, we can estimate the design life by $t = 122(-\ln 0.9)^{\frac{1}{1.77}} \cong 34$ (seconds). In conclusion, the benefit of our proposed model is to give us a unified and statistical index to evaluate the performance of automated video surveillance systems. However, since the cause of failure in an automated video surveillance system is not limited to the case of two randomly walking objects, two open topics in this area are: (1) multiple criteria can be added to test the failure of the system such as drastic changes of illumination, distance between objects, the effect of CPU load, etc. Accordingly, multidimensional probability density estimation (eq. the method of multidimensional kernel density estimation) can be used to estimate the reliability model of the system; (2) the lack of completely available failure data and indeterminate nature of future events lead to the uncertainty problem. Thus, the uncertainty analysis (eq. the combination of Maximum-Entropy Principle [Kapur89, Dai07] and Bayesian approach) of the derived reliability model is necessary, which can help us understand the applicability of the reliability model.



t = 0 (seconds) *t = 15 (seconds)*

……



t = 65 (seconds) *t = 115 (seconds)*

Figure 10.2 Illustration of one experimental video sequence for reliability measurement.

Table 10.1 Summary of goodness of fit for four theoretical distributions.

Distribution	**Test Statistic**	**Critical Value**	***H*$_0$**	**Index of Fit**
Exponential	15.3	7.81	Reject	0.6
Weibull	4.8	5.9	**Accept**	0.92
Normal	7.24	5.9	Reject	0.68
Lognomal	10	5.9	Reject	0.65

Bibliography

[Agapito01] L. Agapito, E. Hayman, and I. Reid, "Self-Calibration of Rotating and Zooming Cameras", *International Journal of Computer Vision*, vol. 45, no. 2, pp. 107-127, 2001.

[Agrawal03] M. Agrawal and L. S. Davis, "Camera calibration using spheres: a semi-definite programming approach", *Proceedings of 9th IEEE International Conference Computer Vision*, October, 2003.

[Akaike74] H. Akaike, "A new look at the statistical model identification", *IEEE Transactions on Automatic Control*, vol. 19, no. 6, pp. 16-23, 1974.

[Allen93] P. K. Allen, A. Timcenko, B. Yoshimi, and P. Michelman, "Automated tracking and grasping of a moving object with a robot hand-eye system", *IEEE Transaction on Robotics and Automation*, vol. 9, no. 2, pp. 151-165, 1993.

[Angella07] F. Angella, L. Reithler, and F. Gallesio, "Optimal Deployment of Cameras for Video Surveillance Systems", *IEEE Conference on Advanced Video and Signal Based Surveillance*, Sept. 2007.

[Antonini06] G. Antonini, S. Venegas, M. Bierlaire and J. Thiran, "Behavioral priors for detection and tracking of pedestrians in video sequences", *International Journal of Computer Vision*, vol. 69, no. 2, pp. 159-180, 2006.

[Applegate06] D. L. Applegate, R. E. Bixby, V. Chevtal, and W. J. Cook, *The Traveling Salesman Problem: A Computational Study*, Princeton University Press, 2006.

[Arbel01] T. Arbel and F. P. Ferrie, "Entropy-based gaze planning", *Image Vision Computation*, vol. 19, no. 11, pp. 779-786, 2001.

[Ayer94] S. Ayer, P. Schroeter, and J. Bigun, "Segmentation of moving objects by robust motion parameter estimation over multiple frames", *European Conference on Computer Vision*, 1994.

[Bagdanov06] A. D. Bagdanov, A. D. Bimbo, and W. Nunziati, "Improving Evidential Quality of Surveillance Imagery Through Active Face Tracking", *18th International Conf. on Pattern Recognition*, 2006.

[Baillieul86] A. Baillieul, "Avoiding obstacles and resolving kinematic redundancy", *IEEE International Conference on Robotic Automation*, San Francisco, 1986.

[Bajcsy88] R. Bajcsy, "Active perception", *Proceedings of the IEEE*, 1988.

[Balcells05] M. Balcells, D. DeMenthon, and D. Doermann, "An Appearance-based Approach for Consistent Labeling of Humans and Objects in Video", *Pattern and Application*, pp.373-385, 2005.

[Barraquand91] J. Barraquand, B. Langlois, and J. -C. Latombe, "numerical potential field techniques for robot path planning", *IEEE Transactions on Robotic Automation*, 1991.

[Barreto98] J. P. A. Barreto, P. Peixoto, J. Batista, and H. Aranjo, "Control performance issues in a binocular active vision system", *IEEE International Conference Intelligence Robots and System*, 1998.

[Bar-Shalom88] Y. Bar-Shalom and T. Fortmann, "Tracking and data association", *Mathematics in Science and Engineering*, Academic Press, 1988.

[Basu97] A. Basu and K. Ravi, "Active camera calibration using pan, tilt, and roll", *IEEE Transactions on System, Man, Cybernetic, Part B*, vol. 27, no. 3, pp. 559-566, July, 1997.

[Batista98] J. Batista, P. Peixoto, and H. Araujo, "Real-time active surveillance by integrating peripheral motion detection with foveated tracking", *IEEE Workshop on Visual Surveillance*, 1998.

[Bay06] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: speeded up robust feature*", 9th European Conference on Computer vision*, 2006.

[Bearse98] P. M. Bearse, H. Bozdogan, "Subset selection in vector autoregressive (VAR) models using the genetic algorithm with informational complexity as the fitness function," Systems Analysis Modeling Simulation, pp.61-91, 1998.

[Beauchemin95] S. S. Beauchemin and J. L. Barron, "The computation of optical flow", *ACM Computing Surveys*, vol. 27, no. 3, pp. 437-467, 1995.

[Beleznai06] C. Beleznai, B. Fruhstuck, H. Bischof, "Multiple object tracking using local PCA", *18th Int'l Conf. on Pattern Recognition*, June, 2006.

[Black01] J. Black and T. Ellis, "Multiple camera image tracking", *Proceedings Performance Evaluation of Tracking and Surveillance Conference (PETS 2001), with CVPR 2001*, December, 2001.

[Black98] M. Black and A. Jepson, "Aprobabilistic framework for matching temporal trajectories: condensation-based recognition of gestures and expressions", *European Conference on Computer Vision*, 1998.

[Blanco03] J. Blanco, W. Burgard, R. Sanz, and J. L. Fernandez, "Fast face detection for mobile robots by interating laser range data with vision", *International Conference on Advanced Robotics*, 2003.

[Blostein91] S. Blostein and T. Huang, "Detecting small moving objects in image sequences using sequential hypothesis testing", *IEEE Transactions on Signal Processing*, vol. 39, no. 7, pp. 1611-1629, 1991.

[Bohme03] H.-J. Bohme, T. Wilhelm, J. Key, C. Schauer, C. Schroter, H.-M. Grosz, and T. Hempel, "An approach to multi-modal human-machine interaction for intelligent service robots", *Robotics and Autonomous Systems*, vol. 44, p. 83-96, 2003.

[Borenstein91] J. Borenstein and Y. Koren, "The vector field histogram-fast obstacle avoidance for mobile robots", IEEE Transactions on Robotics and Automation, vol. 7, no. 3, pp. 278-288, 1991.

[Bose97] P. Bose, L. J. Guibas, A. Lubiw, M. H. Overmars, D. L. Souvaine, and J. Urrutia, "The floodlight problem", *International Journal of Computation Geometry Application*, Vol. 7, no. 1, pp. 153-163, 1997.

[Boult99] T. E. Boult, R. Micheals, X. Gao, P. Lewis, C. Power, W. Yin, and A. Erkan, "Fame-rate omnidirectional surveillance and tracking of camouflaged and occluded targets," *IEEE Workshop on Visuall Surveillance*, June 1999.

[Bolt04] T. E. Boult, X. Gao, R. Micheals, and M. Eckmann, "Omnidirectional visual surveillance," *Image and Vision Computing*, vol. 22, pp. 515-534, 2004.

[Bolt03] T. E. Boult, "Geo-spatial active visual surveillance on wireless networks", *Applied Imagery Pattern Recognition Workshop*, Washington, DC, October, 2003.

[Bozdogan00] H. Bozdogan, "Akaike's Information Criterion and Recent Developments in Information Complexity", Journal of Mathematical Psychology, Vol. 44, No. 1, 2000, pp. 62-91.

[Brauer-Burchardt01] C. Brauer-Burchardt and K. Voss, "A new algorithm to correct fisheye and strong wide angle lens distortion from single images", *International Conference on Image Processing*, 2001.

[Broaddus05] C. Broaddus, "Statistical model selection for automatic geometric camera calibration", MS Thesis, The University of Tennessee, Knoxville, 2005.

[Cai99] Q. Cai, and J. K. Aggarwal, “Tracking human motion in structured environments using a distributed-camera system”, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 2, pp. 1241-1247, 1999.

[Calderara05] S. Calderara, A. Prati, R. Vezzani, and R. Cucchiara, “Consistent Labeling for Multi-Camera Object Tracking”, *The 13th International Conference on Image Analysis and Processing*, September, 2005.

[Carlsson91] S. Carlsson, B. J. Nilsson, S. C. Ntafos, “Optimum guard covers and m-watchmen routes for restricted polygons”, *Workshop on Algorithms and Data Structures*, 1991.

[Caspi00] Y. Caspi and M. Irani, “A step towards sequence-to-sequence alignment”, *IEEE Conference on Computer Vision and Pattern Recognition*, June 2000.

[Cen07] Y. Cen, L. Wang, and h. Zhang, “Real-time obstacle avoidance strategy for mobile robot based on improved coordinating potential field with genetic algorithm”, *16th IEEE International Conference on Control Applications, Singapore*, October, 2007.

[Chang01] S. Chang and T. H. Gong, “Tracking multiple people with multi-camera system”, *Proceedings of IEEE Workshop on Multi-Object Tracking*, 2001.

[Chang00] T. Chang, S. Gong, and E. Ong, “Tracking multiple people under occlusion using multiple cameras”, *Proceedings of British Machine Vision Conference*, 2000.

[Chen07A] I.-H. Chen and S.-J. Wang, “An efficient approach for the calibration of multiple PTZ cameras”, *IEEE Transactions on Automation Science and Engineering*, Vol. 4, no. 2, April, 2007.

[Chen07B] I.-H. Chen and S.-J. Wang, “Efficient vision-based calibration for visual surveillance systems with multiple PTZ cameras”, *IEEE International Conference on Computer Vision Systems*, 2006.

[Chen08] C.-H. Chen, Y. Yao, D. Page, B.R. Abidi, A. Koschan, M. Abidi, "Heterogeneous Fusion of Omnidirectional and PTZ Cameras for Multiple Object Tracking", *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 18, No. 8, pp. 1052-1063, August 2008.

[Chvatal75] V. Chvatal, “A combinatorial theorem in plane geometry”, *J. Comb. Theory Series*, vol. 18, pp. 39-41, 1975.

[Collins00] R. T. Collins, A. J. Lipton, and T. Kanade “Introduction to the special section on video surveillance”, *IEEE Transactions on Pattern Machine and Machine Learning*, vol. 22, no. 8, pp. 745-746, August 2000.

[Corke96] P. I. COrke and M. C. Good, “Dynamic effects in visual closed-loop systems”, *IEEE Transaction on Robotics and Automation*, vol. 12, no. 5, pp. 671-683, 1996.

[Cortes02] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, “Coverage control for mobile sensing networks”, *IEEE Conference on Robotics and Automation*, 2002.

[Cox93] I. Cox, “A review of statistical data association techniques for motion correspondence”, *International Journal of Computer Vision*, vol. 1, no. 1, pp. 53-66, 1993.

[Cui98] Y. Cui, S. Samarasekera, Q. Huang, and M. Greiffenhagen, “Indoor monitoring via the collaboration between a peripheral sensor and a foveal sensor”, *IEEE Workshop on Visual Surveillance*, , India, January, 1998.

[Dai07] Y.-S. Dai, M. Xie, Q. Log, and S.-H. Ng, “Uncertainty analysis in software reliability modeling by bayesian approach with maximum-energy principle”, *IEEE Transactions on Software Engineering*, vol. 33, no. 11. pp. 781-795, 2007.

[Das88] H. Das, J. –J. E. Slotine, and T. B. Sheridan, "Inverse Kinematic algorithms for redundant systems", *IEEE Transactions on Robotics and Automation*, 1988.

[Das95] S. Das and N. Ahuja, "Perfrmance analysis of stereo, vegence, and focus as depth cues for active vision", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 12, pp. 1213-1219, 1995.

[Dockstader01] S. Dockstader and A. Tekalp, "Multiple camera tracking of interacting and occluded human motion", *Proceedings of the IEEE,* 2001.

[Doermann00] D. Doermann and D. Mihalcik, "Tools and Techniques for video performance evaluation", *The 15th International Conf. on Pattern Recognition*, Sept. 2000.

[Doubek04] P. Doubek, I. Geys, T. Svoboda, and L. VanGool, "Cinematographic rules applied to a camera network", *Proceedings of the 5th Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras*, 2004.

[Ebeling97] C. E. Ebeling, *An Introduction to Reliability and Maintainability Engineering*, McGraw-Hill, 1997.

[Efrat00] A. Efrat, L. J. Guibas, S. Har-Peled, D. C. Lin, J.S. B. Mitchell, and T. M. Murali, "Sweeping simple polygons with a chain of guards", *Symposium on Discrete Algorithms*, 2000.

[Erdem06] U. M. Erdem and S. Sclaroff, "Automated Camera Layout to Satisfy Task-Specific and Floor Plan-Specific Coverage Requirements", *Computer Vision and Image Understanding*, vol. 103, no. 3, pp. 156-169, Sept. 2006.

[Erdem04] C. Erdem, A. Tehalp, and B. Sankur, "Metrics for performance evaluation of video object segmentation and tracking without ground-truth", *IEEE International Conference. on Image Processing*, Oct. 2004.

[Estivill-Castro95] V. Estivill-Castro, J. O'Rourke, J. Urruita, and D. Xu, "Illumination of polygons with vertex lights", *Information Processing Letter*, vol. 56, no 1, pp. 9-13, 1995.

[Everts07] I. Everts, N. Sebe, and G. A. Jones, "Cooperative object tracking with multiple PTZ cameras", *The 14th International Conference on Image Analysis and Processing*, September 2007.

[Faugeras92] O. Faugeras, T. Luong, and S. Maybank, "Camera self-calibration: theory and experiments", *Proceedings of European Conference on Computer Vision*, 1992.

[Fayman01] J. F. Fayman, O. Sudarsky, E. Rivlin, and M. Rudzsky, "Zoom tracking and its applications," Machine Vision and Applications, vol. 13, pp. 25-37, 2001.

[Fox99] D. Fox, W. Burgard, F. Dellaert, and S. Thrun, "Monte Carlo location: efficient position estimation for mobile robots", *Proceedings of the National Conference on Artificial Intelligence*, 1999.

[Fisk78] S. Fisk, "A short proof of Chvatal's watchman theorem", *Journal of Combination Theory Series*, vol. 24, pp. 374, 1978.

[Fitzgibbon01] A. W. Fitzgibbon, "Simultaneous linear estimation of multiple view geometry and lens distortion", *IEEE International Conference on Computer Vision and Pattern Recognition*, 2001.

[Fluret08] F. Fluret, J. Berclaz, R. Lengagne, and P. Fua, "Multicamera people tracking with a probabilistic occupancy map", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 267-273, 2008

[Gordon93] N. Gordon, "A hybrid bootstrap filter for target tracking in clutter", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, No. 1, pp. 353-358, 1997.

[Gordon93] N. Gordon, D. Salmond, and A. Smith, "A novel approach to nonlinear/non-Gaussian Bayesian state estimation", *IEE Proceedings*, 1993.

[Graham03] J. Graham and K. Shillcutt, "Robot Tracking of Human Subjects in Field Environments", *International Symposium on Artificial Intelligence, Robotics, and Automation in Space*, 2003.

[Guibas99] L. J. Guibas, J. C. Latombe, S. M. LaValle, D. Lin, and R. Motwani, "A visibility-based pursuit-evasion problem", *International Journal of Computer Geometrical Application*, vol. 9, no. 4, pp. 471-493, 1999.

[Guler03] S. Guler, J. M. Griffith and I. A. Pushee, "Tracking and handoff between multiple perspective camera views", *IEEE Proceeding of the 32nd Applied Imagery Pattern Recognition Workshop*, USA, October, 2003.

[Gutmann98] J.-S. Gutmann, W. Burgard, D. Fox, and K. Konolige, "An experimental comparison of localization methods", *IEEE International Conference on Intelligent Robots and Systems*, 1998.

[Haasch04] A. Haasch, S. Hoheneer, S. Huwel, M. Kleinehagenbrock, S. Lang, L. Toptsis, G. A. Fink, J. Fritsch, B. Wrede, and G. Sagerer, "BIRON-the bielefeld robot companion", *International Workshop on Advances in Service Robotics*, 2004.

[Hartley94] R. I. Hartley, "Self-calibration from multiple views with a rotating camera", *IEEE International Conference on Computer Vision and Pattern Recognition*, 1994.

[Hartley99] R. I. Hartley, E. Hayman, L. D. Agapito, and I. Reid, "Camera calibration and the search for infinity", *IEEE International Conference on Computer Vision*,1999.

[Hemayed03] E. E. Hemayed, "A survey of camera self-calibration", *IEEE Conference on Advanced Video and Signal Based Surveillance*, July, 2003.

[Hennessey92] M. P. Hennessey and M. Donath, "The multiplexed joint method for determining the inverse kinematics of a N DOF serial link manipulator", *Advance Robotics*, 1992.

[Heyden97] A. Heyden and K. Astrom, "Euclidean reconstruction from image sequences with varying and unknown focal length and principal point", *IEEE International Conference on Computer Vision and Pattern Recognition*, 1997.

[Heyden99] A. Heyden and K. Astrom, "Fexible calibration: minimal cases for auto-calibration", *IEEE International Conference on Computer Vision*, 1999.

[Hoiem07] D. Hoiem, A. N. Stein, A. A. Efros, and M. Hebert, "Recovering occlusion boundaries from a single image," *IEEE International Conference on Computer Vision*, October 2007.

[Horaud06] R. Horaud, D. Knossow, and M. Michaelis, "Camera cooperation for achieving visual attention", *Machine Vision and Applications*, vol. 16, no. 6, pp. 331-342, February, 2006.

[Hu04] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors", *IEEE Trans. on System, Man, and Cybernetics-Part C: Applications and Reviews*, vol. 34, no. 3, pp. 334-343, Aug. 2004.

[Huang04] L. Huang, S. Kumar, and C. –C. Jay Kuo, "Adaptive resource allocation for multimedia Qos management in wireless networks", *IEEE Transactions on Vehicular Technology*, vol. 53, no 2, march 2004.

[Hutchinson96] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control", *IEEE Transaction on Robotics and Automation*, vol. 12, no. 5, pp. 651-670, 1996.

[Isard96] M. Isard and A. Blake, "Contour tracking by stochastic propagation of conditional density", *European Conference of Computer Vision*, 1996.

[Iwata06] K. Iwata, Y. Satoh, I. Yoda, and K. Sakaue, "Hybrid camera surveillance system by using stereo omnidirectional system and robust human detection", *IEEE Pacific-Rim Symposium on Image and Technology (PSIVT 2006)*, Taiwan, December, 2006.

[Javed05] O. Javed, K. Shafique, and M. Shas, "Appearance Modeling for Tracking in Multiple Non-Overlapping Cameras", *IEEE Conference on Computer Vision and Pattern Recognition*, June 2005.

[Javed03] O. Javed, Z. Rasheed, K. Shafique, and M. Shah, "Tracking across Multiple Cameras with Disjoint Views", *IEEE International Conference on Computer Vision*, Oct. 2003.

[Javed05] O. Javed, K. Shafique, and M. Shas, "Appearance modeling for tracking in multiple non-overlapping cameras", *IEEE Conference on Computer Vision and Pattern Recognition*, June 2005.

[Jaynes02] C. Jaynes, S. Webb, R. M. Steele, and Q. Xiong, "An open development environment for evaluation of video surveillance systems", *Proceedings of the Third International Workshop on Performance Evaluation of Tracking and Surveillance*, June 2002.

[Jing04] X.–J. Jing, Y.-C. Wang, and D.–L. Tan, "Artificial coordination field based real-time coordinating collision-avoidance planning for multiple mobile robots", *Control Theory and Applications*, vol. 21, no. 5, pp. 757-764, 2004.

[Junejo07] I. N. Junejo and H. Foroosh, "Geometrically optimized PTZ camera calibration from two images", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007.

[Kahl00] F. Kahl, B. Triggs, and K. Astrom, "Critical motions for auto-calibration when some intrinsic parameters van vary", *Journal of Mathematical Imaging Visibility*, vol. 13, no. 2, pp. 131-146, 2000.

[Kannala04] J. Kannala and S. Brandt, "A generic camera calibration method for fish-eye lenses", *International Conference on Pattern Recognition*, August, 2004.

[Kanazawa95] K. Knazawa, D. Koller, and S. Russell, "Stochastic simulation algorithms for dynamic probabilistic networks", *The 11th Annual Conference on Uncertainty in Artificial Intelligence*, 1995.

[Kang03] J. Kang, I. Cohen, and G. Medioni, "Continuous tracking within and across camera streams", *IEEE International Conference on Computer Vision and Pattern Recognition*, June 2003.

[Kang05] J. Kang, I. Cohen, and G. Medioni, "Persistent Objects Tracking across Multiple Non-overlapping Cameras", *IEEE Workshop on Motion and Video Computing*, 2005.

[Kant90] K. Kang, "Fast collision avoidance for manipulator arms: a sequential search strategy", *IEEE Transactions on Robotics and Automation*, vol. 6, no. 5, 1990.

[Kapur89] J. Kapur, *Maximum-Entropy Models in Science and Engineering*, John Wiley & Sons, 1989.

[Kelly95] P. Kelly, A. Katkere, D. Kuramura, S. Moezzi, S. Chatterjee, and R. Jain, "An architecture for multiple perspective interactive video", *Proceedings of ACM Multimedia 95*, May 1995.

[Khan03] S. Khan and M. Shah, "Consistent Labeling of Tracked Objects in Multiple Cameras with Overlapping Fields of View," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no.10, pp. 1355-1360, Oct. 2003.

[Khatib86] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots", *International Journal on Robotic Research*, vol. 5, no. 1, pp. 90-98, 1986.

[Kim92] J.–O. Kim and P. K. Khosla, "Real-time obstacle avoidance using harmonic potential functions", *IEEE Transactions on Robotic Automation*, 1992.

[Klennrock75] L. Klennrock, Queuing systems Volume 1: Theory, New York: Wiley, 1975.

[Kovalev96] V. Kovalev and M. Petrou, "Mutidimensional co-occurrence matrices for object recognition and matching", *Graphical Models and Image Processing*, vol. 58, no. 3, pp. 187-197, 1996.

[Koivo91] A. J. Koivo and N. Houshangi, "Real-time vision feedback for servoing robotic manipulator with self-tuning controller", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, No. 1, pp. 134-142, 1991.

[Krumn00] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer, "Multi-camera multi-person tracking for easyliving", *Proceedings of IEEE International Workshop on Visual Surveillance*, 2000.

[Lazarevic-McManus06] N. Lazarevic-McManus, J. Renno, and G. A. Jones, "Performance evaluation in visual surveillance using the F-measure", *The 4th ACM International Workshop on Video Surveillance and Sensor-Networks*, Oct. 2006.

[Lei06] R. Lei and L.-Q. Xu, "Real-time outdoor video surveillance with robust foreground extraction and object tracking via multi-state transition management", *Pattern Recognition Letters*, vol. 27, no. 15, pp. 1816-1825, Nov. 2006.

[Lim07] F. L. Lim, W. Leoputra, and T. Tan, "Non-overlapping Distributed Tracking System Utilizing Particle Filter", *The Journal of VLSI Signal Processing*, vol. 49, no. 3, pp. 343-362, 2007.

[Lee00] L. Lee, R. Romano, and G. Stein, "Monitoring activities from multiple video streams: establishing a common coordinate frame", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 758-767, Aug. 2000.

[Li02] J. Li, C. Chua, and Y. Ho, "Color based multiple people tracking", *Proceedings of IEEE International Conference on Control, Automation, Robotics, and Vision*, 2002.

[Li07] H. Li and C. Shen, "An LMI Approach for Reliable PTZ Camera Self-Calibration", *IEEE Conference on Advanced Video and Signal Based Surveillance*, Nov. 2007.

[Lien06] K.-C. Lien and C.-L. Huang, "Multi-view-based cooperative tracking of multiple human objects in clustered scenes", 18th International Conference on Pattern Recognition, June, 2006

[Lim07] F. L. Lim, W. Leoputra, and T. Tan, "Non-overlapping distributed tracking system utilizing particle filter", *The Journal of VLSI Signal Processing*, vol. 49, no. 3, pp. 343-362, 2007.

[Lipton98] A. Lipton, H. Fujiyoshi, and R. Patil, "Moving target detection and classification from real-time video", *Proceedings of IEEE Workshop on Applications of Computer*, 1998.

[List05] T. List, J. Bins, J. Vazquez, and R. B. Fisher, "Performance evaluating the evaluator", *Proceedings of the Third International Workshop on Performance Evaluation of Tracking and Surveillance*, Oct. 2005

[Lowe04] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints", *International Journal of Computer Vision*, vol. 60, no.2, pp. 91-110, 2004.

[Lozano-Perez81] T. Lozano-Perez, "Automatic planning of manipulator transfer movements", *IEEE Transactions on System, Man, and Cybernetic*, vol. 11, no. 10, 1981.

[Lozano-Perez81] T. Lozano-Perez, "A simple motion-planning algorithm for general robot manipulators", *IEEE Transactions on System, Man, and Cybernetic*, vol. 3, no. 3, 1987.

[Lucas81] B. D. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision", *International Joint Conference on Artificial Intelligence*, pp. 674-679, 1981.

[Luo05] X. Luo and S. M. Bhandarkar, "Multiple object tracking using elastic matching", *IEEE Conference on Advanced Video and Signal Based Surveillance*, Sept. 2005.

[Maciejewski85] A. A. Maciejewski and C. A. Klein, "Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments", *International Journal of Robotic Research*, vol. 4, no3, 1985.

[Maccormick99] J. Maccormick and A. Blake, "A probabilistic exclusion principle for tracking multiple objects", *7th International Conference on Computer Vision*, 1999.

[Matas02] J. Matas, O. Chum, M. Urban, and T. Pajdla, "Robust wide baseline stereo from maximally stable extremal regions", *British Machine Vision Conference*, 2002.

[Maver93] J. Maver and R. Bajcsy, "Occlusions as a guide for planning the next view", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 10, pp. 1016-1030, 1999.

[Mikic00] i. Mikic, Huang, and M. M. Trivedi, "Activity monitoring and summarization for an intelligent meeting room", *Workshop on Human Motion*, 2000.

[Mikolajczyk02] K. Mikolajczyk, Detection of local features invariant to affine transformations, *Ph. D. thesis, Institut National Polytechnique de Grenoble*, France, 2002.

[Mittal04] A. Mittal and L.S. Davis, "Visibility Analysis and Sensor Planning in Dynamic Environments", *European Conference on Computer Vision*, May 2004.

[Mittal01] A. Mittal and L. Davis, "Unified multi-camera detection and tracking using regional matching", *Proceedings of IEEE Workshop on Multi-Object Tracking*, 2001.

[Moons96] T. Moons, L. Gool, M. Proesmans, and E. Pauwels, "Affine reconstruction from perspective image pairs with a relative object-camera translation in between", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 1, pp. 77-83, 1996.

[Morita03] S. Morita, K. Yamazawa, and N. Yokoya, "Networked video surveillance using multiple omnidirectional cameras", *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Japan, July, 2003.

[Nakamura86] Y. Nakamura, "Advanced Robitics: Redundancy and Optimization", *Addison Wesley, Reading*, MA, 1991.

[Nelson98] R. C. Nelson and A. Selinger, "Large-scale tests of a keyed, appearance-based 3D object recognition system", *Vision Research*, vol. 38, no. 15, pp. 2469-2488, 1998.

[Neter04] K. N. Neter, Applied linear regression models, Fourth Edition, McGraw Hill, 2004.

[O'Rouke87] J. O'Rourke, *Art Gallery Theorems and Algorithms*, New York: Oxford, 1987.

[Orekhov07] V. Orekhov, B. Abidi, C. Broaddus, and M. Abidi, "Universal Camera Calibration with Automatic Distortion Model Selection", *IEEE International Conference on Image Processing*, 2007.

[Perez04] L. G. Perez, M. C. G. Alegre, A. Ribeiro, D. Guinea, and J. M. Canas, "Perception and Tracking of Dynamic Objects for Optimization of Avoidance Strategies in Autonomous Piloting of Vehicles", *International Conference Spatial Cognition 2004*, Germany, October, 2004.

[PETS06] PETS: Performance Evaluation of Tracking and Surveillance, http://www.cvg.rdg.ac.uk/slides/pets.html.

[Philomin00] V. Philomin, R. Duraiswami, and L. Davis, "Pedestrian Tracking From a Moving Vehicle", *Proceedings of the IEEE Intelligent Vehicles Symposium 2000 Dearborn (MI)*, USA, October 3-5, 2000.

[Pito99] R. Pito, "A solution to the next best view problem for automated surface acquisition", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 10, pp. 1016-1030, 1999.

[Pitt99] M. Pitt and N. Shephard, "Filtering via simulation: auxiliary particle filters", *Journal of the American Statistical Association*, vol. 94, 1999.

[Pollefeys99] M. Pollefeys, R. Koch, and L. V. Gool, "Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters", *International Journal of Computer Vision*, vol. 32, no. 1, pp. 7-25, 1999.

[Pope00] A. R. Pope and D. G. Lowe, "Probabilistic models of appearance for 3D object recognition", *International Journal of Computer Vision*, vol. 40, no. 2, pp. 149-167, 2000.

[Remagnino02] P. Remagnino and G. A. Jones, "Automated registration of surveillance data for multi-camera fusion", *5th International Conference Information Fusion*, July 2002.

[Rosin91] P. Rosin, and T. Ellis, "Detecting and classifying intruders in image sequences", *Proceedings of British Machine Vision Conference*, 1991.

[Schlogl04] T. Schlogl, C. Beleznai, M. Winter, and H. Bischof, "Performance evaluation metrics for motion detection and tracking", *The 17th International Conference. on Pattern Recognition*, August 2004.

[Schulz01] D. Schulz, W. Burgard, D. Fox, and A. B. Cremers, "Tracking multiple moving objects with a mobile robot", *IEEE International Conference on Computer Vision and Pattern Recognition*, 2001.

[Scotti05] G. Scotti, L. Marcenaro, C. Coelho, F. Selvaggi, and C. S. Regazzoni, "Dual camera intelligent sensor for high definition 360 degrees surveillance", *Vision, Image, and Signal Processing*, vol. 152, no. 2, Apr. 2005, pp. 250-257.

[Sebe05] I .O. Sebe, S. You, and U. Neumann, "Globally optimum multiple object tracking", *SPIE Defense and Security Symposium*, 2005.

[Seraji91] H. Seraji, R. Colbaugh, and K. Glass, "Redundant robot can avoid obstacles", *JPL Invention Report*, NPO-17852/7346, October, 1991.

[Shah03] M. Shah, "Understanding Human behavior from motion imagery", *Machine Vision and Applications*, Vol 14, pp. 210-214, September, 2003

[Shibata02] M. Shibata and T. Honma, "3D object tracking on active stereo vision robot", *IEEE International Workshop on Advanced Motion Control*, 2002.

[Shibata04] M. Shibata and H. Kawasumi, “Solution for stereo correspondences on active stereo vision robot”, *IEEE International Workshop on Advanced Motion Control*, 2004.

[Sturm04] P. Sturm and S. Ramalingam, “A generic concept for camera calibration”, *European Conference on Computer Vision*, 2004.

[Sugimure04] Y. Sugimure and J. Sato, “Camera Calibration and Reconstruction from the Chain Connection of Mutual Camera Projections”, *17th International Conference on Pattern Recognition*, 2004

[Suzuki01] I. Suzuki, Y. Tazoe, M. yamashita, and T. Kameda, “Searching a polygonal region from the boundary”, *International Journal of Computation Geometry Application*, vol. 11, no. 5, pp. 529-553, 2001.

[Svoboda05] T. Svoboda, D. Martinec, and T. Pajdla, “A Convenient Multi-Camera Self-Calibration for Virtual Environments”, *PRESENCE: Teleoperators and Virtual Environments*, vol. 14, pp. 407-422, 2005.

[Tan94] T. N. Tan, G. D. Sullivan, and K. D. Baker, “Recognizing objects on the ground plane”, *Image Vision Computing*, vol. 12, no. 164, pp. 172, 1994.

[Tarabanis95] P. K. A. Tarabanis and R. Y. Tsai, “A survey of sensor planning in computer vision”, *IEEE Transactions on Robotics and Automation*, pp. 86-104, 1995.

[Thirthala05] S. R. Thirthala, S. N. Sinha, and M. Pollefeys, “Calibration of pan-tilt-zoom (PTZ) cameras and omni-directional cameras”, *IEEE Conference on Computer Vision and Pattern Recognition*, San Diego, CA, Jun. 2005.

[Trivedi02] T. E. Boult, A. Prati, and G. Kogut, “Distributed interactive video arrays for event based analysis of incidents”, *International Conference on Intelligent Transportation Systems*, Singapore, September, 2002.

[Tuytelaars04] T. Tuytelaars and L. V. Gool, “Matching Widely Separated Views Based on Affine Invariant Regions,” *International Journal of Computer Vision*, vol. 59, no. 1, pp. 61-85, 2004.

[Ulrich00] I. Ulrich and J. Borenstein, “VFH: reliable obstacle avoidance with look-ahead verification”, *IEEE International Conference on Robotics and Automation*, 2000.

[Wackerly02] D. D. Wackerly, W. Mendenhall III, and R. L. Scheaffer, *Mathematical Statistics with Applications*, Sixth Edition, Duxbury Advanced Series, 2002.

[Wang06] J.-M. Wang, C.-T. Tsai, S. Cherng, and S.-W. Chen, “Omnidirectional camera networks and data fusion for vehicle tracking in an indoor parking lot,” *IEEE International Conference on Video and Signal Based Surveillance*, Sydney, Australia, Nov. 2006.

[Warren89] C. W. Warren, J. C. Danos, and B. W. Mooring, “An approach to manipulator path planning”, *International Journal of Robotic Research*, vol. 8, no. 5, 1989.

[Yao06] Y. Yao, B. Abidi, and M. Abidi, "Fusion of omnidirectional and PTZ cameras for accurate cooperative tracking", *IEEE International Conference on Advanced Video and Signal Based Surveillance*, November. 2006.

[Yao08] Y. Yao, “Long range automated persistent surveillance” *Ph. D. Thesis*, The University of Tennessee, Knoxville, 2008.

[Yilmaz06] A. Yilmaz, O. Javed, and M. Shah, “Object tracking: a survey”, *ACM Computing Surveys*, vol. 38, no. 4, Dec. 2006.

[Yue04] Z. Yue, S. Zhou, and R. Chellappa, "Robust two-camera tracking using homography", *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2004.

[Zhang00] Z. Ahang, "A flexible new technique for camera calibration", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 11, pp. 1330-1334, 2000.

[Zhang04] Z. Zhang, "Camera calibration with one-dimensional objects", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol 26, no. 7, pp. 892-899, July, 2004.

[Zhao05] T. Zhao, M. Aggarwal, R. Kumar, and H. Sawhney, "Real-time wide area multi-camera stereo tracking", *IEEE Conference on Computer Vision and Pattern Recognition*, USA, June, 2005.

[Zhu00] Z. Zhu, K. K. Rajaseka, E. Riseman, and A. Hanson, "Panoramic virtual stereo vision of cooperative mobile robots for localizing 3D moving objects", *IEEE Workshop on Omnidirectional Vision*, Hilton Head Island, SC, June 2000.

Vita

Chung-Hao Chen, the husband of Chiung-Ju Yeh, was born in Taoyuan, Taiwan, on July 29, 1974, the son of Kuei-Lan Hsieh and Eing-Ming Chen. After graduating in 1993 from Wu-Ling Middle School, Taoyuan, Taiwan, he attended Fu-Jen Catholic University of Computer Science and Information Engineering where he received both a Bachelor of Science degree in 1997 and a Master of Science degree in 1999. During the summer of 2004, he joined the Imaging, Robotics, and Intelligent Systems Laboratory as a graduate research assistant where he completed his Doctor of Philosophy degree in 2009.